

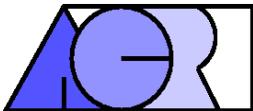
ACRi Software Tools

ANSWER<sup>®</sup> ◆ PORFLOW<sup>®</sup> ◆ TIDAL<sup>®</sup>

## KEYWORD COMMANDS

VERSION 6.42.9  
*Rev: 0*

23 April 2018



©ANALYTIC & COMPUTATIONAL RESEARCH, INC. 1985-2019  
ALL RIGHTS RESERVED

---

---

**COPYRIGHT NOTICE**

---

---

The **ACRi** Software Tools, **ANSWER®**, **PORFLOW®**, **TIDAL®** and **CFDStudio®**, and this documentation, are proprietary products of Analytic & Computational Research, Inc. (**ACRi**) and are protected by the U.S. and international copyright laws. One copy of **ACRi** Software Tools may be made for backup or archival purposes unless multiple copies are permitted under a specific contractual agreement with **ACRi**. Copyright laws prohibit making additional copies of the software for any other reason. Duplication of the written materials accompanying this software is prohibited without the expressed written consent of **ACRi**.

---

---

© ACRi

**IMPORTANT -- PLEASE READ CAREFULLY BEFORE USING THIS SOFTWARE****SOFTWARE LICENSE AGREEMENT**

Please read this license agreement before using this SOFTWARE. This agreement is a legal contract between you (the CUSTOMER) and Analytic & Computational Research, Inc. (ACRi) governing the use of this SOFTWARE. Your first use of this SOFTWARE indicates your acceptance of the terms of this agreement. *Please note that in some instances, the software license is on an annual basis and you must pay an annual fee for continued use of the SOFTWARE. If an annual fee is applicable, this will be conveyed to you at the time of purchase and be indicated on your invoice or in a License Agreement.* If you do not wish to comply with the terms of this agreement, return the SOFTWARE within 30 days and your money will be refunded. If you have any questions concerning this agreement, please contact ACRi, attn. Software Sales, 1931 Stradella road, Bel Air, CA 90077 or call (+1) (310) 471-3023 or send us email at [sales@acriCFD.com](mailto:sales@acriCFD.com).

1. CUSTOMER may use the SOFTWARE on a single-user computer or a single terminal or workstation of a multi-user computer or network. Each installation, terminal or workstation must be separately licensed by ACRi.
2. CUSTOMER may not sublicense, assign, rent, lease or transfer the software or license to another party without the explicit written consent of ACRi. **The software is licensed to the CUSTOMER as the END USER.** CUSTOMER may not use the SOFTWARE for the purposes of another party or use the SOFTWARE for performing simulations for, or consulting with, another party that result in derived income for the CUSTOMER without obtaining explicit written consent of ACRi and without paying a royalty of 15% on such income.
3. CUSTOMER may not incorporate, or allow another party to incorporate the SOFTWARE, in whole or in part, in any other software.
4. CUSTOMER's license to use the SOFTWARE becomes effective only after the CUSTOMER has paid all amounts which are agreed to by the CUSTOMER and the CUSTOMER has received written notification from ACRi that the LICENSE is now active.
5. This AGREEMENT is effective until terminated.
6. This AGREEMENT is subject to immediate termination if CUSTOMER violates any of its provisions.
7. In the event of termination of this AGREEMENT, CUSTOMER shall discontinue all further use of the SOFTWARE and return all copies of the SOFTWARE and documentation supplied by ACRi.
8. Upon termination, customer shall provide written notice that all use of the SOFTWARE has ended.
9. ACRi warrants that the diskettes or electronic files containing the SOFTWARE and accompanying documentation are free of defects in materials and workmanship for a period of 30 days from the date of purchase. In the event of notification of any physical defects in diskettes or documentation, ACRi will replace the defective diskettes and documentation.
10. ACRi disclaims all other warranties, expressed or implied, including but not limited to implied warranties of fitness or usefulness for a particular purpose, either with respect to defects in the diskettes, and documentation, or with respect to any defects of coding and logic in the SOFTWARE, even if such defects are known to ACRi or have been brought to ACRi's attention.
11. ACRi will not be liable to CUSTOMER, or any other party, for consequential loss or damages arising from the use, operation, or modification of the SOFTWARE by CUSTOMER. ACRi will not be responsible for any loss, damage, or claim resulting from reliance by CUSTOMER or any other party on results obtained by use of the SOFTWARE.

If any provision of this AGREEMENT shall be unlawful, void, or for any reason unenforceable, then that provision shall be deemed severable from this AGREEMENT and shall not affect the validity and enforceability of the remaining provisions of this AGREEMENT.

---

---

**WARRANTY**

---

---

No warranty expressed or implied, is provided that this document is complete or accurate in all respects. The information contained in this document, and the software that it describes, are subject to change without notice.

---

---

---

---

**NOTICE**

---

---

The ACRi Software Tools contain multiple options which are described in this document. Which of these options is actually present in your installation of the Software Tool may depend upon the type of contract under which the software was acquired. If you have any questions about the presence or absence of a particular option, please contact ACRi.

---

---

---

**PREFACE**

---

During the past 30 years, ACRi Software Tools **ANSWER®**, **PORFLOW®**, **TIDAL®** and **CFDStudio®**, have evolved from simple mathematical models for flow and heat transfer into comprehensive **Computational Fluid Dynamics Software Tools for analysis of a wide range of engineering and environmental applications** for a wide range of problems. These tools provide for coupled transport of fluid, heat and multiple chemical species in complex geometry. These are able to simulate the transient or steady state behavior of compressible or incompressible fluids in diverse host media and multiple phases with chemical reactions, combustion and other complex physical processes.

**ANSWER®** is a General Purpose CFD package based on the Navier-Stokes Equations. It has been used to analyze problems as diverse as low pressure film deposition, cooling of electronic components, automobile fluid dynamics, ventilation of tunnels, HVAC design optimization, analysis of process tanks, design of ramjets and aircraft engines, and missile launch tube simulations, lubrication of bearings, and performance of high Mach number missiles.

**PORFLOW®** software tool simulates flow in porous and or fractured media based on Darcy Equation. It has been used to analyze problems as diverse as salinity intrusion into fresh water aquifers and remediation of hazardous waste sites. It has been used to evaluate pumping of an aquifer over a period of days, remediation of waste sites over a period of years, corrosion of waste canisters over tens of years, and transport of contaminants from nuclear waste over a time span of hundreds of thousands of years.

**TIDAL®** is based on the Shallow Water Equations and has been used to analyze a wide range of problems including impact of Tsunamis and hurricanes, pollution management of coastal waters, design of marinas, optimization of water treatment outfall locations, transport of oil spills and impact of offshore oil terminals on coastal ecosystem.

**ACRi Software Tools** have evolved with the user's needs to provide a flexible format that is bound neither to a specific algorithm, nor to a particular methodology. Rather, these provide a framework that facilitates experimentation. The user can change numerical schemes, solution method, matrix inversion algorithms, or any of the physical or mathematical features. Two features deserve special mention: generality of applications, and ease of use provided by the conversational **FREEFORM™** command language. These have enabled **ACRi Software Tools** to emerge as leading software in their field of application.

**ACRi Software Tools** are distinguished by the diversity of users. These include commercial, research and educational organizations in over 20 countries. Among our users are AECL (Canada), Aerospatiale (France), Allison Gas Turbine, ANDRA (France), ASCI, S.A. (Spain), BAe-Sema (U.K.), Battelle Pacific Northwest Laboratory, C.N.I.M. (France), Department of Education (Mexico), Fluor Daniel, GERP (FRANCE), General Electric Company, Idaho National Engineering Laboratory, James M. Montgomery, Lam Research Corporation, Lockheed Martin, Marquardt Company, National Aeronautical and Space Administration, Oak Ridge National Laboratory, Renault (France), Rockwell, Savannah River Laboratory, SCK-CEN (Belgium)SNECMA (France), Southwest Research Institute, University of California, US Air Force, US Army, US Navy, U.S. Department of Energy, U.S. Nuclear Regulatory Commission, U.S. Geological Survey, Watkins-Johnson, Westinghouse Corporation, WS Atkins (UK), and a number of other commercial organizations.

Hundreds of publications and project reports on the applications, benchmarking, and verification of **ACRi Software Tools** are currently available. **ACRi Software Tools** has been extensively peer-reviewed and these details are available both on the internet and as well as from our website.

**ACRi Software Tools** rely on the numerical solution of complex mathematical equations. Some familiarity with the strengths and weaknesses of mathematical and numerical algorithms is highly recommended. Every attempt has been made to provide the necessary information for satisfactory use of **ACRi Software Tools** in this manual. As is the case for any software of this type, it is not possible to anticipate all questions and users' requirements. In addition to this manual, we also provide training and support in the use of the software. For additional questions and inquiries, please contact:

**Analytic & Computational Research, Inc. (ACRi).**

1931 Stradella Road, Bel Air, California, 90077

**Email:** [techsupport@ACRiCFD.com](mailto:techsupport@ACRiCFD.com) Phone: +1 310-471-3023

**Web:** [www.ACRiCFD.com](http://www.ACRiCFD.com) & [www.acri.fr](http://www.acri.fr)

*Akshai Runchal  
Bel Air, California*

**ACKNOWLEDGEMENTS**

---

The **ACRi Software Tools** would not have been possible without the support and active contributions of a number of individuals. As can be expected for any scientific application software designed to meet the evolving needs of its users over an extended period of time, many individuals have contributed to its growth. Users have actively participated in the development process by testing the software and contributing suggestions for its improvement.

The contributions of numerous users of the software from all over the world are gratefully acknowledged. Their faith in the value of this software, suggestions for improvements in the user interface and, critique of the user document have proved invaluable. They contributed to the software in numerous ways and enhanced the value of the software immeasurably.

## CONTENTS

	Page
<b>COPYRIGHT NOTICE</b> .....	<b>2</b>
<b>SOFTWARE LICENSE AGREEMENT</b> .....	<b>3</b>
<b>WARRANTY</b> .....	<b>4</b>
<b>NOTICE</b> .....	<b>4</b>
<b>PREFACE</b> .....	<b>5</b>
<b>ACKNOWLEDGEMENTS</b> .....	<b>6</b>
<b>CONTENTS</b> .....	<b>7</b>
<b>Page</b> .....	<b>7</b>
<b>LIST OF TABLES</b> .....	<b>12</b>
<b>LIST OF FIGURES</b> .....	<b>13</b>
<b>NOTATION</b> .....	<b>14</b>
<b>CHAPTER 1</b> .....	<b>18</b>
<b>ACRi FREEFORM™ COMMAND LANGUAGE</b> .....	<b>18</b>
1.1 FREEFORM™ INPUT RECORDS.....	19
1.1.1 <i>Keyword Record</i> .....	19
1.1.2 <i>Continuation Record</i> .....	19
1.1.3 <i>Comment Record</i> .....	19
1.2 ELEMENTS OF INPUT RECORD.....	20
1.2.1 <i>Keyword</i> .....	20
1.2.2 <i>Modifier</i> .....	21
1.2.3 <i>Numeric Field</i> .....	22
1.2.4 <i>Separator Field</i> .....	23
1.2.5 <i>Terminator Character</i> .....	24
1.2.6 <i>Comment Field</i> .....	25
1.2.7 <i>Prompt Character</i> .....	26
<b>CHAPTER 2</b> .....	<b>28</b>
<b>INPUT AND OUTPUT OPTIONS</b> .....	<b>28</b>
2.1 OVERVIEW OF KEYWORD COMMAND LANGUAGE.....	29
2.2 ORDER OF INPUT COMMANDS .....	33
2.3 SPECIFICATION OF GRID AND PROBLEM DIMENSIONALITY.....	34
2.4 INPUT AND OUTPUT FILE UNITS.....	35
2.5 UNITS OF PHYSICAL QUANTITIES .....	37
2.6 DIAGNOSTIC OUTPUT SPECIFICATION .....	38
2.7 DEPENDENT VARIABLES OF ACRi™ SOFTWARE TOOLS .....	39
2.8 TABULAR OUTPUT OF FIELD VARIABLES.....	40
2.9 ARCHIVAL AND POST-PROCESSING OUTPUT.....	45
2.10 RESTART OUTPUT .....	46
2.11 MODIFICATION OF INPUT DATA DURING SIMULATIONS.....	47
<b>CHAPTER 3</b> .....	<b>50</b>
<b>KEYWORD COMMAND NOTATION</b> .....	<b>50</b>
3.1 KEYWORD COMMAND NOTATION.....	51
3.2 Interactive or Run Time Input .....	52
3.3 File Name Specification .....	53
3.4 Sub region Identification on an input command .....	54
3.5 Boundary Identification .....	55
3.6 Specification of Fluid Phase .....	57

3.7	Specification of Frequency for Output .....	58
<b>CHAPTER 4</b>	.....	<b>60</b>
<b>FUNCTIONAL INPUT FOR FREEFROM COMMANDS</b>	.....	<b>60</b>
4.1	General Functional Relationships.....	61
4.1.1	<i>The Tabular Function</i> .....	61
4.1.2	<i>Discrete Point-Wise Function</i> .....	61
4.1.3	<i>Discrete Point-Wise Time-Dependent Function</i> .....	61
4.1.4	<i>Discrete Point-Wise Random Functions</i> .....	62
4.1.5	<i>The Constant Function</i> .....	63
4.1.6	<i>Replacement Function</i> .....	63
4.1.7	<i>Multivariate Polynomial Functions</i> .....	63
4.1.8	<i>The Higher Order Polynomial Functions</i> .....	63
4.1.9	<i>The Power Functions</i> .....	63
4.1.10	<i>The Exponential Functions</i> .....	63
4.1.11	<i>The Logarithmic Functions</i> .....	63
4.1.12	<i>The Trigonometric Functions</i> .....	64
4.1.13	<i>Fourier Series</i> .....	64
4.1.14	<i>Hyperbolic Functions</i> .....	64
4.1.15	<i>Rational Functions</i> .....	64
4.1.16	<i>Power of Ten Function</i> .....	64
4.1.17	<i>Inverse Exponential Function</i> .....	64
4.1.18	<i>The Error Function</i> .....	65
4.1.19	<i>Gate Function</i> .....	65
4.1.20	<i>Sutherland Function</i> .....	66
4.1.21	<i>Multivariate Linear Sum</i> .....	66
4.1.22	<i>Multivariate Square Sum</i> .....	66
4.1.23	<i>Multivariate Square Root of Sum of Squares</i> .....	66
4.2	Command Input of Functional Form .....	67
4.3	Specification of additional operations for functions .....	71
4.4	Examples of Functional Form of Input.....	72
4.5	Node by Node Input Function.....	75
4.6	Node by Node Input as a Function of Time.....	76
4.7	User-Defined Input Options .....	77
<b>CHAPTER 5</b>	.....	<b>82</b>
<b>KEYWORD COMMANDS</b>	.....	<b>82</b>
5.1	ACRI .....	83
5.2	ACTIVE .....	84
5.3	ADAPTIVE .....	85
5.4	ADIABATIC.....	86
5.5	ALLOCATE .....	87
5.6	ARRHENIUS.....	89
5.7	BANNER .....	90
5.8	BLOCK.....	91
5.9	BOREHOLE.....	98
5.10	BOUNDARY .....	100
5.11	BRINKMAN-FORCHHEIMER .....	108
5.12	CAPILLARY .....	110
5.13	CLOSE.....	111
5.14	COMPRESSIBLE .....	112
5.15	CONDUCTIVITY.....	116

---

5.16	CONNECTIVITY .....	133
5.17	CONVERGENCE .....	143
5.18	COORDINATE .....	149
5.19	CORIOLIS .....	159
5.20	CORRELATION .....	160
5.21	CPU .....	163
5.22	DATUM .....	164
5.23	DEBUG .....	165
5.24	DECAY .....	168
5.25	DEFINE .....	170
5.26	DENSITY .....	176
5.27	DIAGNOSTIC .....	181
5.28	DIFFUSION .....	183
5.29	DISABLE .....	184
5.30	DISPERSIVITY .....	186
5.31	DISTRIBUTION COEFFICIENT .....	189
5.32	EBU .....	201
5.33	ELSE .....	203
5.34	END .....	204
5.35	ENDIF .....	205
5.36	FILE .....	206
5.37	FIX .....	208
5.38	FLOW .....	211
5.39	FLUID .....	218
5.40	FLUX .....	223
5.41	FOR .....	229
5.42	FRACTURE .....	230
5.43	FRICTION .....	232
5.44	FUEL .....	234
5.45	GAS .....	235
5.46	GEOMETRY .....	239
5.47	GRAVITY .....	245
5.48	GRID .....	247
5.49	HISTORY .....	251
5.50	HYDRAULIC .....	258
5.51	IF-ELSE-ENDIF .....	259
5.52	INCLUDE .....	261
5.53	INDENT .....	262
5.54	INITIAL .....	263
5.55	INJECTION .....	264
5.56	INLET .....	269
5.57	INTEGRATION .....	270
5.58	LAGRANGIAN TRANSPORT .....	271
5.59	LAMINAR .....	276
5.60	LAND .....	277
5.61	LIMIT .....	278
5.62	LOCATE .....	279
5.63	MATERIAL .....	308
5.64	MATRIX .....	313
5.65	META .....	328
5.66	MOVE .....	331

---

5.67	MULTIPHASE .....	336
5.68	NOZZLE .....	349
5.69	OPEN .....	363
5.70	OPTION .....	366
5.71	OUTLET .....	374
5.72	OUTPUT .....	379
5.73	OXIDIZER .....	388
5.74	PAUSE .....	389
5.75	PERIODIC .....	390
5.76	PERMEABILITY .....	394
5.77	PHASE .....	397
5.78	POROSITY .....	399
5.79	PRANDTL .....	401
5.80	PRECIPITATION .....	402
5.81	PRINT .....	404
5.82	PROBLEM .....	412
5.83	PROPERTY .....	420
5.84	QUIT .....	423
5.85	RADIATION .....	424
5.86	REACTION .....	429
5.87	READ .....	435
5.88	REFERENCE .....	437
5.89	REGENERATION .....	441
5.90	RELAX .....	442
5.91	RENAME .....	443
5.92	RETARDATION .....	444
5.93	SAVE .....	445
5.94	SCALE .....	457
5.95	SCHMIDT NUMBER .....	458
5.96	SCREEN .....	459
5.97	SELECT .....	460
5.98	SET .....	461
5.99	SOLID .....	479
5.100	SOLVE .....	480
5.101	SOOT .....	488
5.102	SOURCE .....	492
5.103	SPECIFIC HEAT .....	507
5.104	SPILL .....	508
5.105	STACK .....	512
5.106	STATISTICS .....	515
5.107	STORAGE COEFFICIENT .....	516
5.108	SWIRL .....	521
5.109	SYMMETRY .....	527
5.110	THERMAL .....	528
5.111	TIDE .....	529
5.112	TIME .....	533
5.113	TITLE .....	534
5.114	TORTUOSITY .....	535
5.115	TRACK .....	536
5.116	TRANSFER .....	539
5.117	TRANSPORT .....	546

5.118	TURBULENCE .....	549
5.119	UPTAKE.....	553
5.120	USER .....	554
5.121	VELOCITY .....	555
5.122	VISCOSITY.....	556
5.123	WALL .....	560
5.124	WRITE .....	568
5.125	ZONE .....	577
	<b>REFERENCES .....</b>	<b>578</b>
	<b>APPENDIX A: FILES PROCESSED DURING EXECUTION.....</b>	<b>583</b>
A.0	INTRODUCTION .....	583
A.1	acrNAMES.ACR .....	584
A.2	acrDFLT.ACR .....	585
A.3	acrINSRT.ACR .....	586
A.4	acrSTOP.ACR .....	587
A.5	acrDONE.ACR.....	588
	<b>APPENDIX B: FILE FORMATS.....</b>	<b>590</b>
B.1	INPUT FILE FORMATS.....	590
	<i>B.1.1 Input file for NODE wise Time-Dependent Function (Section 4.1.3)</i>	<i>590</i>
	<b>APPENDIX C: RADM STOCHASTIC TRANSPORT METHODOLOGY .....</b>	<b>591</b>
C.1	THEORETICAL DESCRIPTION.....	591
	<i>C.1.1 Basic Transport Equation .....</i>	<i>591</i>
	<i>C.1.1 Computational Algorithm .....</i>	<i>591</i>
C.2	COMPUTATION OF CONCENTRATION .....	591
	<i>C.2.1 Species Concentration from Inverse-Distance-Square Algorithm .</i>	<i>591</i>
	<i>C.2.2 Species Concentration from Location of Particle .....</i>	<i>591</i>
	<i>C.2.3 Species Concentration from Particle density .....</i>	<i>592</i>
	<i>C.2.4 Species Concentration from Radius of Influence .....</i>	<i>592</i>

## LIST OF TABLES

TABLE 2.2.1: FUNCTIONAL UNITS OF ACRi™ SOFTWARE TOOLS COMMANDS .....	33
TABLE 2.4.1: I/O FILE UNITS AND THEIR DEFAULT ASSIGNMENT .....	36
TABLE 2.7.1: DEPENDENT VARIABLES OF ACRi™ SOFTWARE TOOLS .....	39
TABLE 2.8.1: STANDARD VARIABLES OF ANSWER™ SOFTWARE TOOL .....	41
TABLE 2.8.2: STANDARD VARIABLES OF PORFLOW™ SOFTWARE TOOL .....	42
TABLE 2.8.3: STANDARD VARIABLES OF TIDAL™ SOFTWARE TOOL .....	43
TABLE 2.8.4: SUPPLEMENTARY VARIABLES OF ACRi™ SOFTWARE TOOLS .....	44
TABLE 2.11.1: ILLUSTRATION OF A 2-SEGMENT SIMULATION SEQUENCE .....	48
TABLE 3.4.1: VALID SUBREGION IDENTIFICATION MODIFIERS .....	54
TABLE 3.5.1 BOUNDARY IDENTIFICATION INDEX .....	55
TABLE 3.6.1: VALID MODIFIERS FOR PHASE DESIGNATION .....	57
TABLE 4.2.1: VALID INPUT FUNCTIONAL FORMS .....	68
TABLE 4.2.1: VALID INPUT FUNCTIONAL FORMS (Continued) .....	69
TABLE 4.2.2: VALID INDEPENDENT VARIABLES .....	70
TABLE 4.2.3: NUMERICAL INPUT FOR THE TABLE SPECIFICATION .....	70
TABLE 4.2.4: NUMERICAL INPUT FOR SIMPLE ANALYTIC FUNCTIONS .....	70
TABLE 4.2.5: NUMERICAL INPUT FOR THE SERIES FUNCTIONS .....	70
TABLE 4.2.6: NUMERICAL INPUT FOR USER SPECIFIED FUNCTIONS .....	70
TABLE 4.3.1: ADDITIONAL OPERATIONS ON FUNCTIONS .....	71
TABLE 4.4.1: EXAMPLES OF SIMPLE ANALYTIC FUNCTIONS .....	72
TABLE 4.4.2: EXAMPLES OF ANALYTIC FUNCTION SERIES .....	73
TABLE 4.4.3: EXAMPLES OF TABULAR FUNCTIONS .....	74
TABLE: VAPOR PRESSURE FOR WATER .....	398

LIST OF FIGURES

FIGURE 3.5.1: BOUNDARY INDEX NOTATION .....56

## NOTATION

Symbol	MEANING	UNITS		
		GENERIC	SI	FPS
a	A constant or exponent	various	various	Various
a <sub>R</sub>	Absorptivity coefficient	L <sup>-1</sup>	1/m	1/ft
b	A constant or exponent	various	various	various
c	A constant or exponent	various	various	various
c	Speed of sound	L t <sup>-1</sup>	m/s	ft/s
C <sub>D</sub>	Drag coefficient	-----	-----	-----
C <sub>P</sub>	Specific heat	L <sup>2</sup> t <sup>2</sup> T <sup>-1</sup>	J/(kg K)	BTU/(lbm°F)
E	Black body radiation energy	M t <sup>-3</sup>	W/m <sup>2</sup>	BTU/ft <sup>2</sup>
f	A frequency	t <sup>-1</sup>	1/s	1/s
F	General transport variable	various	various	various
g	Gravitational acceleration	L t <sup>-2</sup>	m/s <sup>2</sup>	ft/s <sup>2</sup>
h	Enthalpy	L <sup>2</sup> t <sup>-2</sup>	m <sup>2</sup> /s <sup>2</sup>	ft <sup>2</sup> /s <sup>2</sup>
h <sup>o</sup>	Enthalpy of formation of a species	L <sup>2</sup> t <sup>-2</sup>	m <sup>2</sup> /s <sup>2</sup>	ft <sup>2</sup> /s <sup>2</sup>
h <sub>s</sub>	Stagnation enthalpy	L <sup>2</sup> t <sup>-2</sup>	m <sup>2</sup> /s <sup>2</sup>	ft <sup>2</sup> /s <sup>2</sup>
H <sub>R</sub>	Heat of reaction	L <sup>2</sup> t <sup>-2</sup>	m <sup>2</sup> /s <sup>2</sup>	ft <sup>2</sup> /s <sup>2</sup>
I	Radiation intensity	M t <sup>-3</sup>	W/m <sup>2</sup>	BTU/ft <sup>2</sup>
k	Kinetic energy of turbulence	L <sup>2</sup> t <sup>-2</sup>	m <sup>2</sup> /s <sup>2</sup>	ft <sup>2</sup> /s <sup>2</sup>
m	Rate of injection of fluid per unit volume	M L <sup>-3</sup> t <sup>-1</sup>	kg/(m <sup>3</sup> s)	lbm/(ft <sup>3</sup> s)
m	Mass fraction of a species	-----	-----	-----
M	Molecular weight	M mol <sup>-1</sup>	kg/mol	lbm/mol
N	Coordinate normal to a boundary	L	m	ft
N <sub>sn</sub>	Density of soot nuclei	L <sup>-3</sup>	1/m <sup>3</sup>	1/ft <sup>3</sup>
N <sub>ss</sub>	Concentration of soot particles	M L <sup>-3</sup>	kg/m <sup>3</sup>	lbm/ft <sup>3</sup>
P	Thermodynamic pressure	M L <sup>-1</sup> t <sup>-2</sup>	N/m <sup>2</sup>	lbf/ft <sup>2</sup>
P <sub>R</sub>	Reference value of Thermodynamic pressure	M L <sup>-1</sup> t <sup>-2</sup>	N/m <sup>2</sup>	lbf/ft <sup>2</sup>
P <sub>k</sub>	Rate of production of turbulent energy	M L <sup>-1</sup> t <sup>-3</sup>	W/m <sup>3</sup>	BTU/(ft <sup>3</sup> s)
r	Radius of curvature	L	m	ft
R	Radiation flux	M t <sup>-3</sup>	W/m <sup>2</sup>	BTU/ft <sup>2</sup>

... Notation continued

## NOTATION - Continued

Symbol	MEANING	UNITS		
		GENERIC	SI	FPS
R	Gas constant	$L^2 t^{-2} T^{-1}$	J/(kg K)	BTU/(lbm °F)
Re	Reynolds number	-----	-----	-----
R <sub>U</sub>	Universal gas constant	$L^2 t^{-2} T^{-1}$	J/(kg-mol K)	BTU/(lbm-mol R)
s	Reaction or decay rate of a property	$M L^{-3} t^{-1}$	kg/(m <sup>3</sup> s)	lbm/(ft <sup>3</sup> s)
s <sub>R</sub>	Scattering coefficient	$L^{-1}$	1/m	1/ft
S	Source of a fluid property	various	various	various
t	Time	t	s	s
T	Thermodynamic Temperature	T	K	R
T <sub>a</sub>	Convert temperature to absolute units	T	K	R
T <sub>c</sub>	Critical temperature of the fluid	T	K	R
uu	Autocorrelation of U	$L^2 t^{-2}$	m <sup>2</sup> /s <sup>2</sup>	ft <sup>2</sup> /s <sup>2</sup>
U	Fluid velocity in x-direction	$L t^{-1}$	m/s	ft/s
vv	Autocorrelation of V	$L^2 t^{-2}$	m <sup>2</sup> /s <sup>2</sup>	ft <sup>2</sup> /s <sup>2</sup>
V	Fluid velocity in y-direction	$L t^{-1}$	m/s	ft/s
V <sub>i</sub>	Velocity in i <sup>th</sup> direction	$L t^{-1}$	m/s	ft/s
ww	Autocorrelation of W	$L^2 t^{-2}$	m <sup>2</sup> /s <sup>2</sup>	ft <sup>2</sup> /s <sup>2</sup>
W	Fluid velocity in z-direction	$L t^{-1}$	m/s	ft/s
x	x-coordinate	L	m	ft
y	y-coordinate	L	m	ft
z	z-coordinate	L	m	ft

## GREEK SYMBOLS

Symbol	MEANING	UNITS		
		GENERIC	SI	FPS
$\alpha$	A coefficient	various	various	various
$\beta$	A coefficient	various	various	various
$\gamma$	Ratio of specific heats	-----	-----	-----
$\Gamma$	Diffusion coefficient	$M L^{-1} t^{-1}$	kg/(m s)	lbm/(ft s)
$\epsilon$	Rate of dissipation of turbulence energy	$L^2 t^{-3}$	J/(kg s)	BTU/(lbm s)
$\epsilon$	A small quantity	-----	-----	-----
$\theta$	Angular coordinate	radian	radian	radian
$\kappa$	Von Karman Constant	-----	-----	-----
$\lambda$	Decay rate constant	$t^{-1}$	1/s	1/s
$\mu$	Fluid viscosity	$M L^{-1} t^{-1}$	kg/(m s)	lbm/(ft s)
$\rho$	Fluid density	$M L^{-3}$	kg/m <sup>3</sup>	lbm/ft <sup>3</sup>
$\sigma$	Prandtl or Schmidt number	-----	-----	-----
$\sigma$	Stefan-Boltzman constant	$M t^{-3} T^{-4}$	J/(m <sup>2</sup> s K <sup>4</sup> )	Btu/(ft <sup>2</sup> s R <sup>4</sup> )
$\tau$	Shear stress	$M L^{-1} t^{-2}$	N/m <sup>2</sup>	lbf/ft <sup>2</sup>
$\varphi$	Density-related variable	$M L^{-3}$	kg/m <sup>3</sup>	lbm/ft <sup>3</sup>
$\varphi_E$	Equivalence ratio	-----	-----	-----
$X$	Fuel fraction in fine or ambient scale of flow	-----	-----	-----

## SUBSCRIPTS

SYMBOL	PERTAINING TO
e	Effective value of a fluid property
$\Phi$	Fluid property F
i	The $i^{\text{th}}$ coordinate direction or $i^{\text{th}}$ phase
inj	Injected fluid
j	The $j^{\text{th}}$ coordinate direction or $j^{\text{th}}$ phase
k	The $k^{\text{th}}$ coordinate direction or $k^{\text{th}}$ phase
t	Turbulent state of fluid
t	The time coordinate
w	The wall
x	The x direction
y	The y or r direction
z	The z or $\theta$ direction

## SUPERSCRIPTS

SYMBOL	PERTAINING TO
$\Phi$	Pertaining to property $\Phi$
j	Pertaining to the $j^{\text{th}}$ coordinate direction or $j^{\text{th}}$ phase
t	Pertaining to the time coordinate
*	Approximate or reference value of a variable
'	Variation in the value of a variable
n	Pertaining to the $n^{\text{th}}$ fluid or the $n^{\text{th}}$ chemical species
k	Pertaining to the $k^{\text{th}}$ time step

## CHAPTER 1

## ACRi FREEFORM™ COMMAND LANGUAGE

ACRi **FREEFORM™** command language, developed by Analytic & Computational Research, Inc. (**ACRi**), provides a simple and easy to use user interface for complex software packages through a set of conversational, English-like commands. These commands are free of any requirements of format or hierarchy except those naturally arising from the nature of the input. These provide for interactive input or emulate the interactive input in batch mode. Since the input commands are in conversational English-like language, **FREEFORM™** provides a very powerful tool to perform Quality Assurance (QA) and ensure that the input correctly reflects the intended problem specification.

All **ACRi Software Tools** including **ANSWER®**, **PORFLOW®** and **TIDAL®** employ the **FREEFORM™** command language. In addition the **CFDStudio®** provides a flexible, simple to use, versatile pre-and post processor Graphical User Interface (**GUI**). The **CFDStudio®** allows the user to build grids painlessly and display the computed results in a variety of graphical images. The software is written in a mix of **FORTRAN**, **C**, **C++**, **JAVA** and **HTML** languages and is essentially independent of any specific platform.

## 1.1 FREEFORM™ INPUT RECORDS

The **FREEFORM™** Input is specified through three types of records: KEYWORD, CONTINUATION and COMMENT records. These are described below.

### 1.1.1 *Keyword Record*

**Function** To specify the numeric and character data.

#### Structure

- ◆ A keyword record must begin with a keyword.
- ◆ Only one keyword per record is allowed.
- ◆ Modifiers and numerical fields may follow the keyword.
- ◆ Comment, separator, or terminator fields must separate all the keyword, the modifiers and the numerical fields from each other.
- ◆ Any character or numeric data on a keyword record after the first occurrence of a terminator are ignored.

### 1.1.2 *Continuation Record*

**Function** To continue numeric and character input started by a previous keyword record.

#### Structure

- ◆ A continuation record must begin with either a separator or a numeric character as the first character of the record. It must not begin with an alphabetic ('A' through 'Z' or 'a' through 'z') character as the first character of a record.
- ◆ A continuation record may only occur after a keyword record for that group.
- ◆ A continuation record must consist only of a combination of modifiers and numerical fields separated from each other by separators.
- ◆ Any character or numeric data on a continuation record after the first occurrence of a terminator is ignored.
- ◆ Any number of continuation records may follow a keyword record.

### 1.1.3 *Comment Record*

**Function** To enhance the clarity and readability of the input.

#### Structure

- ◆ A comment record must begin with a back-slash (/), asterisk (\*), dollar (\$) or exclamation (!) character in the first column of a record. Any combination of characters may follow the first character.
- ◆ A comment record is not processed. No numerical or character data are extracted; the record is merely written to the output file.
- ◆ A comment record cannot be extended by a continuation record.
- ◆ A comment record can be inserted anywhere in the input.

## 1.2 ELEMENTS OF INPUT RECORD

One or more of the following *seven* basic components comprise an input record: KEYWORD, MODIFIER, NUMERIC, SEPARATOR, TERMINATOR, COMMENT and PROMPT fields. These are described below.

### 1.2.1 *Keyword*

**Function** The keyword identifies the input group.

#### **Structure**

- ◆ The keyword may consist of any characters except separator (Section 1.2.4) or terminator (Section 1.2.5) characters. However, the first character of a keyword must be alphabetic ('A' through 'Z' or 'a' through 'z'). To this extent, the concept of a keyword is similar to that of a variable name in FORTRAN.
- ◆ The keyword may be in upper or lower case.
- ◆ A keyword must begin in the first column of a record unless a command with INDENT keyword occurred before the current keyword. In this case, leading blanks may precede the keyword. The INDENT option is deactivated if a command with INDENT OFF is encountered
- ◆ The keyword is terminated with the first occurrence of a valid separator or terminator character.
- ◆ The keyword may consist of 1 to 256 characters. In general, if there are more than four characters, only the first four are considered meaningful and machine-identifiable. However specific exemptions may exist where more than 4 characters are meaningful and these are identified as needed.

#### **EXAMPLES**

---

ABCD, A123, A&B+, A&B. are all valid examples of a keyword. The keyword specifications of ABCD, abcd, ABCDEFGH, AbCd123, ABCDxxxxxx (where x stands for any character) are all equivalent because only the first four characters are significant and the input is case-insensitive.

1ABC, 567, (abc, 'abc, .abc are all invalid keywords since the 1<sup>st</sup> character in all of these is not alphabetic.

Note that a specification of ABC) or ABC', ABC\$, although valid, is equivalent to that of ABC because the last character in all of these examples is either a separator (Section 1.2.4) or a terminator (Section 1.2.5).

### 1.2.2 *Modifier*

Any character information on an input record following a keyword, except that embedded in a numeric or comment field (see Sections 1.2.3 and 1.2.6), is treated as modifier(s).

**Function** To specify character data that helps in interpretation of the rest of the input data

#### **Structure**

- ◆ A modifier in any input group, if present, must follow the keyword.
- ◆ The modifier is identical to the keyword in its structure. It may consist of any characters, except separator and terminator characters, of which the first character must be alphabetic.
- ◆ A modifier must not start in the first column of a record. It can be from 1 to 32 characters in length; In general, if there are more than four characters, only the first four are considered meaningful and machine-identifiable. However specific exemptions may exist where more than 4 characters are meaningful and these are identified as needed.
- ◆ The modifier must be separated from the keyword, other modifiers and numeric data by a valid separator, terminator or comment field.

#### **EXAMPLES**

---

The structure of a modifier is identical to that of a keyword except that it must not start in the first column of a record. Examples are given in Section 1.2.1

### 1.2.3 Numeric Field

Any numeric characters on a keyword or continuation record following a keyword, except those embedded in a keyword, modifier or comment field (see Sections 1.2.1, 1.2.2 and 1.2.6), are treated as numeric data.

**Function** A numeric field contains numeric data for input variables.

#### Structure

- ◆ A numeric field is a continuous string of characters that must begin with the numeric character set. In this context, the numeric character set consists of the numerals (0-9), the decimal point (.), and the plus (+) and minus (-) operators.
- ◆ A numeric field must consist only of the numeric character set defined above, the asterisk (\*), and the exponent in lower (e, d) or upper (E,D) case. It must not contain any other character.
- ◆ The plus (+) or minus (-) sign, if present, **must immediately precede the numerical value without any intervening blank or other characters.**
- ◆ The asterisk (\*) or the exponent (E, D, e or d), if present, must be embedded; the numeric field must not begin or end with one of these characters.
- ◆ A numeric field must be separated from the keyword, modifiers and other numeric fields by a valid separator, terminator or comment field.
- ◆ Numeric field may be located anywhere on a keyword or continuation record.
- ◆ The numeric values may be specified in any of the following formats:
  - Integer, (e.g., 999),
  - Real (e.g., 999.0, 999.)
  - Exponent (e.g., 9.99E2, 1.2349E1, 9.99D+01 or 1.2345d-01)
- ◆ Successive, repetitive, identical numeric values may be specified by the asterisk (\*) option. Thus, (30., 30., 30.) may be represented as (3\*30. or 3\*3.0E+1); embedded separators or non-numeric characters must not appear in such specification.

#### EXAMPLES

---

The input character strings, 1, 0.1234, .567, +123., -1.0005, 1.2e00, 1.35E0, and 3\*1.2 are all valid examples of a numerical field. Input specifications of 123, 123., 1.23e02, +0.123E+3, 1.23D2, 1\*123, 1\*1.23E02 are all equivalent.

The strings 1ABC, 11X11, 1+2, 11., 1+1.E1 are all invalid numeric specifications. In the first three, non-numeric characters follow a leading numeric character, and in the last three, a valid numeric character occurs in an invalid, embedded location.

Note that a specification of 1.2)2. or 1.2=2, although valid, will be equivalent to a specification of two numeric fields, 1.2 and 2 because of the embedded separator (Section 1.2.4) in both cases. A specification of 1.2\$2 is equivalent to a specification of 1.2 because the 2 following the \$ will be ignored (Section 1.2.5).

### 1.2.4 Separator Field

**Function** To separate the keyword, the modifiers and the numeric fields of an input record.

#### Structure

- ◆ Any string of characters on an input record, which consists only of the characters from the separator character set, is treated as a separator field. The set of separator characters may vary from one installation to another. The common recognized separator characters for all ACRi™ Software are:

**Table 1.2.4.1: Valid Separator Characters In FREEFORM™ Language**

Number	Character	Description	ASCII Sequence #
1	' '	Space or Blank	32
2	(	Left parenthesis	40
3	)	Right parenthesis	41
4	,	Comma	44
5	:	Colon	58
6	;	Semicolon	59
7	=	Equal sign	61
8		Vertical Line	124
9	Tab	Horizontal Tab Character	9

#### EXAMPLES

---

The sequence of characters ';;: ), =====, =', and ; are all valid separator fields. However, (a) or (1) are not valid separator fields. In the first case, the character "a" will be processed by FREEFORM™ as a modifier; in the second, the character "1" will be processed as a numeric field.

### 1.2.5 Terminator Character

**Function** To terminate all input on a keyword or continuation record and to provide a vehicle for the user to insert comments on these records.

#### Structure

- ◆ The dollar (\$) and exclamation mark (!) characters are the only valid terminators.
- ◆ The terminator terminates the input for the keyword or continuation record on which it occurs; input associated with that particular keyword may continue on a continuation record which follows.
- ◆ The terminator may appear anywhere in a record.
- ◆ Any characters following the terminator on that input record are not processed but are treated as user comments and are merely written to the output file.

#### EXAMPLES

---

The character sequences:

```
XYZ    $comments now
! any comments here
123.456 !789.123
```

are all examples of sequences with embedded terminators. In the first sequence, XYZ will be treated as valid character data (either keyword or modifier, depending on its starting position on the input record) whereas the characters following \$ will be ignored. In the second example, the complete sequence will be treated as comments. In the third example, the numerical field 789.123 will be ignored, whereas 123.456 will be treated as numeric data.

### 1.2.6 Comment Field

**Function** To provide a vehicle for the user to insert comments in input to enhance the clarity and readability of the input.

#### Structure

- ◆ A comment field may be in the form of an embedded comment or a comment record.
- ◆ An embedded comment field is one which occurs on a keyword or continuation record. It must begin with a terminator (\$ or !) character. Any combination of characters may follow the terminator. The comment field is terminated at the end of the 256<sup>th</sup> character in that record (Section 1.2.5).
- ◆ A comment field on a comment record may consist of any combination of characters. In this case, the comment field begins with the back slash (/), asterisk (\*), or a terminator (\$ or !) character as the first character and terminates with the 256<sup>th</sup> character.

#### EXAMPLES

---

In the input record:

```
ARRAY = 1., 2., 3., 4., 5.          $ EXAMPLE 1
```

the character string "\$ EXAMPLE 1" is an example of the embedded comment on a keyword record. Input processing stops with the \$ character; all characters on that record following, and including, the \$ character are ignored.

As examples of comment records, the character strings:

```
/ARRAY = 1., 2., 3., 4., 5.          $ EXAMPLE 1
* ARRAY = 1., 2., 3., 4., 5.        - another EXAMPLE
!****///// ARRAY = 1., 2., 3., 4., 5.  $ another example
```

will all be treated as comment records and no processing will be done because one of the comment record identifier characters appears as the first character of the input record.

### 1.2.7 Prompt Character

**Function** To allow for interactive or runtime input of a partial or entire input record.

**Structure**

- ◆ The user may insert a '?' in place of either an entire input record or numerical or character string in the input itself. The command interpreter will pause and prompt the user for input at this stage.
- ◆ The input obtained by the prompt may be comprised of one or more of the elements of an input record described in Sections 1.2.1 through 1.2.6.
- ◆ Any input specified by the prompt is inserted in the input command string starting with the location of the question mark symbol.
- ◆ All input in the original command string following the prompt in that record will be disregarded.
- ◆ The input for the prompt may consist of 1 to 256 characters depending on where the question mark symbol is located. The prompt will pause and display a message telling the user the maximum number of characters allowed.
- ◆ It is possible to avoid the message, referred to above, from being displayed by placing two question mark symbols in succession (eg. '??'). The command interpreter will still pause and wait for input, however it will not display a message.

### EXAMPLES

---

In the input record:

```
SOLVe ?                ! Prompt user for input
```

the character string '?' has the effect of pausing the command interpreter during run-time and displaying a message that asks the user for up to 74 characters of input, since the '?' appears in the 7<sup>th</sup> location of the input record. The comment '! Prompt user for input' will be overwritten by the input provided by the user at run-time or will be replaced by blank characters if the user input is less than 74 characters long.

Alternatively the user may choose to write:

```
SOLVe ??              ! Prompt user for input without a message
```

This input record has the same effect as above except no message is displayed.

*(This page left intentionally blank.)*

## CHAPTER 2

## INPUT AND OUTPUT OPTIONS

ACRi™ Software Tools consist of flexible and user-oriented software packages. The software employs CFDStudio™ GUI for user interface. This GUI provides a very powerful, convenient and versatile pre- and post-processor. Underlying the user interface is the FREEFORM™ command language to provide an adaptable, simple to use and format-free commands to communicate with the software. The structure and syntax of the command language are fully described in Chapter 1. The FREEFORM™ command language is a "keyword"-oriented language. A complete discussion of the input and output options, and the keyword commands is given in chapter 7. The user controls the input and much of the operation of the program. Hence, the solution process can be modified during progression of the calculations by convenient user-specified options. The output is largely controlled by the user, both in terms of its extent and its frequency. This chapter provides an introduction to the important input and output features of ACRi™ Software Tools.

## 2.1 OVERVIEW OF KEYWORD COMMAND LANGUAGE

The user input is divided into several groups. Each group is identified by a "keyword" command and may consist of one or more input records. Each input record must be no more than 256 characters long. Each input-record group, in addition to the keyword that identifies the group, may contain "numeric data", "modifiers" and "comments". Separator, terminator or comment fields must separate the keyword, numeric data and modifiers from each other. Comments may be embedded within input records to enhance the clarity and readability of the input. A detailed description of the **FREEFORM™** command language is given in Chapter 1.

The commands that constitute the I/O interface are listed in Table 2.1.1. These commands, and the modifiers and numeric data that are associated with them, are described in alphabetical order in Chapter 3. Except for the geometry and grid of a problem, all other commands may be used repeatedly to accommodate problem requirements that change with time

TABLE 6.1.1: KEYWORDS OF ACRi™ SOFTWARE TOOLS AND THEIR FUNCTIONS

NO.	KEYWORD	INPUT FUNCTION
1	ACRI	Select the <b>ACRi Software Tool</b> for Simulations
2	ACTIVE	Select an active default sub-domain for input specification
3	ADAPTIVE	Refine the specified grid by to Adapt to the computational solution
4	ADIABATIC	Specify adiabatic external walls
5	ALLOCATE	Allocate table space for storage of user input
6	ARRHENIUS	Specify the Arrhenius reaction rate constants
7	BANNER	Print user & program identification to output file
8	BLOCKAGE	Specify blockage or solid objects in flow field
9	BOREHOLE	Specify location and particulars of a borehole
10	BOUNDARY	Override built-in boundary conditions
11	BRINK	Include Brinkman-Forchheimer terms in governing equations
12	CAPILLARY	Effect of temperature and chemical species on capillary pressure
13	CLOSE	Close specified output device immediately
14	COMPRESSIBLE	Select compressible flow option
15	CONDUCTIVITY	Specify conduction or diffusion constants, coefficients and options
16	CONNECTIVITY	To specify element and vertex connectivity for unstructured grid
17	CONVERGENCE	Specify convergence criterion
18	COORDINATE	Specify grid coordinates
19	CORIOLIS	Specify Coriolis force constant
20	CORRELATION	Computation of correlations between variables and locations
21	CPU	Specify number of Central Processing Units for parallel processing
22	DATUM	Specify reference datum level for hydraulic head
23	DEBUG	Specify debug options
24	DECAY	Specify decay constants for chemical species
25	DEFINE	Specify value of a symbolic variable
26	DENSITY	Select fluid mass density options
27	DIAGNOSTIC	Diagnostic output options
28	DIFFUSION	Specify conduction or diffusion constants, coefficients and options
29	DISABLE	Disable certain built-in default options
30	DISPERSIVITY	Specify dispersivity coefficients for the host matrix
31	DISTRIBUTION	Specify distribution coefficient and options
32	EBU	Select Eddy-break-up limiting options for chemical reactions
33	ELSE	Define the "ELSE" part of the IF-ELSE-ENDIF construct
34	END	End of a problem
35	ENDIF	Terminate the IF-ELSE-ENDIF construct
36	FFT	Perform Fast Fourier Transform operations
37	FILE	Open or close selected I/O device
38	FIX	Fixed pressure, temperature or concentration nodes
39	FLOW	Specify mass flux entering at boundary
40	FLUID	Specify thermal and transport properties of the fluid
41	FLUX	Compute flux of fluid, heat or chemical species crossing a sub-domain
42	FOR	To select material type for a porous matrix
43	FRACTURE	Specify location and particulars of an embedded fracture
44	FRICTION	Specify bottom and wind stress friction coefficients

..... Table 6.1.1 continued

TABLE 6.1.1: KEYWORDS OF ACRi™ SOFTWARE TOOLS AND THEIR FUNCTIONS(continued)

NO.	KEYWORD	INPUT FUNCTION
45	FUEL	Specify fuel composition and heat of combustion
46	GAS	Specify gas constant and parameters
47	GEOMETRY	Specify or modify problem geometry
48	GAS	Specify gas constant and parameters
49	GEOMETRY	Specify or modify problem geometry
50	GRAVITY	Specify constants of gravitational acceleration
51	GRID	Number of grid nodes in the x, y and z directions
52	HISTORY	Provide time history output at selected nodes
53	HYDRAULIC	Hydraulic properties of porous matrix
54	IF	To initiate the IF-ELSE-ENDIF construct
55	INCLUDE	To include a file at run time in the input data stream
56	INDENT	To allow command input to be indented
57	INITIAL	Initial conditions for state variables
58	INJECTION	Specification for injection of Liquid Films and Sprays
59	INLET	To specify an inflow boundary for the domain of computation
60	INTEGRATION	Index for selection of integration profile
61	LAGRANGIAN	Specify lagrangian release and tracking of particles
62	LAMINAR	Specify laminar flow
63	LAND	Define Land Boundary
64	LIMIT	Set upper and lower limits for dependent variables
65	LOCATE	Specify location of sub-regions or boundaries
66	MATERIAL	Specify material types and properties
67	MATRIX	Specify Options for solution of matrix of equations
68	META	Specify a meta command file to generate multiple simulations
69	MOVE	Specify moving surface within flow domain
70	MULTIPHASE	Specify multi-phase or multi-fluid hydraulic characteristics
71	NOZZLE	Specify liquid fuel nozzle injection parameters
72	OPEN	Specify an open boundary through which fluid may enter or leave
73	OPTION	Modify built-in default options
74	OUTLET	Specify an outflow boundary for the domain of computation
75	OUTPUT	Frequency and extent of tabular output
76	OXIDIZER	Oxygen to nitrogen ratio in the oxidizer stream
77	PAUSE	Cause a temporary pause in processing
78	PERIODIC	Specify periodic boundaries for the computational domain
79	PERMEABILITY	Specify permeability of the porous medium
80	PHASE	Specify phase change mode and constants
81	POROSITY	Specify material porosity for porous matrix
82	PRANDTL	Specify Prandtl numbers
83	PRECIPITATE	Specify precipitation reaction from fluid to solid phase
84	PRINT	Print flow rate, sources and statistical measures of variables
85	PROBLEM	Specify general nature and type of problem
86	PROPERTY	Option for mode of property specification
87	QUIT	Terminate solution process
88	RADIATION	Activate thermal radiation calculation

..... Table 6.1.1 continued

TABLE 6.1.1: KEYWORDS OF ACRi™ SOFTWARE TOOLS AND THEIR FUNCTIONS(continued)

NO.	KEYWORD	INPUT FUNCTION
89	REACTION	Specify reaction rate constants
90	READ	Read initial conditions from archive file
91	REFERENCE	Reference values for key variables
92	REGENERATION	Specify regeneration rate for chemical decay chain
93	RELAX	Relaxation factors for governing variables
94	RENAME	Allows renaming of output variables listed in Table 6.8.1
95	RETARDATION	Specify retardation coefficient for a transport variable
96	SAVE	Frequency of output to archive file
97	SCALE	Internal scaling of specified input
98	SCHMIDT	Specify Schmidt numbers
99	SCREEN	Echo some of the diagnostic output to screen
100	SELECT	Specify location of sub-regions or boundaries
101	SET	Set value of a variable as a function of space and time
102	SOLID	Specify conjugate heat transfer and properties
103	SOLVE	Start of solution of equations
104	SOOT	Activate soot computations and specify constants
105	SOURCE	Specify source, injection or withdrawal terms
106	SPECIFIC HEAT	Select specific heat options and constants for the fluid
107	SPILL	Specify oil spill and transport at surface of a water body
108	STACK	To specify stack transformation operations
109	STATISTICS	Obtain output of statistical parameters of variables
110	STORAGE	Specify storage coefficient for a governing transport equation
111	SWIRL	To specify characteristics of swirl imposed on flow
112	SYMMETRY	To identify an external boundary as a symmetry plane or axis
113	THERMAL	Thermal properties of solid or porous matrix
114	TIDE	Specify history of tide at a boundary
115	TIME	Set initial time for simulations
116	TITLE	Problem title specification
117	TORTUOSITY	Specify tortuosity of the porous matrix
118	TRACK	Compute Particle tracks and corresponding elapsed time in flow field
119	TRANSFER	Transfer mass and property variables within the computational domain
120	TRANSPORT	Transport properties of porous matrix
121	TURBULENT	Select turbulent model and specify constants
122	UPTAKE	Specify uptake of a species between different phases
123	USER	User identification for input and output files
124	VELOCITY	Select method of computation of Darcy velocity
125	VISCOSITY	Select fluid viscosity options
126	WALL	Specify internal walls within domain of interest
127	WIND	Specify Wind Parameters for TIDAL® Software
128	WRITE	Generate output of variables
129	ZONE	Specify host media zones

## 2.2 ORDER OF INPUT COMMANDS

The **FREEFORM™** command language allows an order-free input format except for the constraints imposed by common sense. For example, the command that initiates the solution of the equations must follow complete specification of the geometry and physics of the problem. Similarly, the command that signifies the end of calculations must be the last input record for a given problem. Table 2.2.1 lists the relationship of the keyword commands to various functional aspects of problem specification. Although it is not necessary, it may be convenient to follow this functional order. The **SOLVE<sup>1</sup>** command initiates solution of the governing equations and must; therefore, follow complete specification of the problem. The **END** command terminates the execution of a solution and, therefore, must be the last command of an input sequence for a problem.

Certain other constraints may be inherent to the nature of a specific problem. For example, any input in terms of sub-regions (e.g., the **SOURCE** command) must obviously follow the subregion specification (**LOCATE** command). Similarly, if the subregion is specified in terms of grid coordinates, then the **LOCATE** command must come after the **COORDINATE** commands that specify the grid coordinates.

**TABLE 2.2.1: FUNCTIONAL UNITS OF ACRi™ SOFTWARE TOOLS COMMANDS**

ORDER	FUNCTION	RELATED KEYWORD COMMAND
1	Identification	<b>BANNER, TITLE, USER</b>
2	Grid specification	<b>ADAPTIVE, CONNECTIVITY, COORDINATE, GEOMETRY, GRID, SCALE</b>
3	Problem definition	<b>ACRI, ACTIVE, BLOCKAGE, BOREHOLE, BRINK, CAPILLARY, DATUM, FRACTURE, GRAVITY, INCLUDE, INJECTION, INLET, LAGRANGIAN, LAND, LOCATE, MATERIAL, MOVING, NOZZLE, OPEN, OPTION, OUTLET, PHASE, PROBLEM, RENAME, SELECT, SYMMETRY, WALL, ZONE</b>
4	Initial and boundary conditions	<b>ADIABATIC, BOUNDARY, FIX, FLOW, FRICTION, INITIAL, INLET, OPEN, OUTLET, PERIODIC, READ, SET, SWIRL, SYMMETRY, TIDE, TIME, WIND</b>
5	Fluid and Host Matrix Properties and constants	<b>CONDUCTION, DENSITY, DIFFUSION, DISPERSIVITY, DISTRIBUTION, FLUID, FOR, FUEL, HYDRAULIC, GAS, LATENT, MULTIPHASE, OXIDIZER, PERMEABILITY, POROSITY, PRANDTL, PROPERTY, REFERENCE, RETARDATION, SCHMIDT, SPECIFIC, STORAGE, THERMAL, TORTUOSITY, TRANSPORT, VISCOSITY</b>
6	Nature of Flow	<b>COMPRESSIBLE, LAMINAR, RADIATION, REACTION, SOLID, SOOT, TURBULENT</b>
7	Source and sink specifications	<b>ARRHENIUS, CORIOLIS, EBU, DECAY, PRECIPITATE, REACTION, REGENERATION, SPILL, SOURCE, UPTAKE</b>
8	Solution options	<b>DISABLE, INTEGRATION, LIMIT, MATRIX, RELAX</b>
9	Output control	<b>CLOSE, CORRELATION, DEBUG, DIAGNOSTIC, FFT, FLUX, HISTORY, PRINT, OUTPUT, SAVE, SCREEN, STACK, STATISTICS, TRACK, VELOCITY, WRITE</b>
10	Operational control	<b>ALLOCATE, CONVERGENCE, DEFINE, ELSE, END, ENDF, FILE, IF, INDENT, META, PAUSE, QUIT, SOLVE</b>

<sup>1</sup> The keywords are denoted by bold uppercase letters.

### 2.3 SPECIFICATION OF GRID AND PROBLEM DIMENSIONALITY

The computations may be in three-dimensional (3D) or two-dimensional (2D) mode. One or the other mode is selected by the **GRID** command. In the 3D mode, the Cartesian geometry is specified in terms of the (x, y, z) and the cylindrical geometry in terms of (x, r,  $\theta$ ) or (x, y, z) coordinates. In the 2D mode, the geometry is specified in terms of (x, y) for cartesian and (x, r) or (x, y) cylindrical geometry.

Each grid consists of a number of elements. A node is nominally placed in each element. By default this node is located at the geometric center of the element. However it can be located anywhere in the element. Each element and the node contained within it is denoted by a number which varies from 1 to NELEM irrespective of the dimensionality of the problem. In addition a boundary node (equivalent to a linear or planar element in Finite Element Terminology) is automatically inserted at each exterior boundary segment of the computational domain. These boundary nodes greatly facilitate the imposition of complex boundary conditions. The boundary nodes are number sequentially from NELEM+1 to NMAX. In this case, the total number of nodes, NMAX, is the sum of number of elements and the number of exterior boundary segments. The sequence in which the element and boundary nodes are numbered depends on the nature of the grid. If the grid is unstructured, then the nodes are numbered exactly in the same order in which the elements are specified on the geometry input. The boundary nodes are also numbered in the order in which the element with exterior boundary is encountered. For a structured grid the element are numbered, in order, along the x, y and then z direction grid lines respectively. These are followed by the boundary nodes which are defined in order of the boundaries along the x, y and z planes.

For the structured grids, a grid index is defined for each grid node. This index varies from (1,1,1) to (IMAX, JMAX, KMAX) for the 3D and from (1,1) to (IMAX, JMAX) for the 2D geometry. IMAX, JMAX and KMAX are, respectively, the maximum number of grid lines in the x, y (or r) and z (or  $\theta$ ) directions. For 2D computations, KMAX is automatically set to 1. One-dimensional computations are performed in a pseudo-2D mode where the gradients of all variables in the y direction are set to zero. The total number of nodes, including the boundaries is NMAX where  $NMAX = IMAX * JMAX$  in 2D and  $NMAX = IMAX * JMAX * KMAX$  in 3D.

**IMPORTANT NOTE:** For **TIDAL®** Software the velocity field is computed only in the 2D mode since only the velocity components are depth-averaged. The thermal and mass transport equations can be solved in the 3D mode but in this case, the appropriate velocity components must be specified.

## 2.4 INPUT AND OUTPUT FILE UNITS

**ACRi™ SOFTWARE TOOLS** employ a number of I/O file units. The function and default file names for these units are defined in Table 2.4.1. The default file type is FORMATTED for all these files. The user is given an opportunity to attach these units to alternate files (or devices) and change the file type for some of these. The files attached to units 15 and 16, being the standard input and output devices, may be redirected only by the operating system commands. The user may change file names for most of these files. Please see the corresponding user commands for the available options. Further operations can be performed on any of the I/O units by the **FILE** and **CLOSE** commands. The term “auto” for unit number means that the unit number is automatically selected from among the available units.

A number of temporary units may be opened in response to user commands. Many of these are named with the file extension **”TMP”** unless the user explicitly specifies a file name on the command.

TABLE 2.4.1: I/O FILE UNITS AND THEIR DEFAULT ASSIGNMENT

UNIT #	DEFAULT FILE NAME	FUNCTION OF THE FILE
15	Console	Read user input commands from standard input device.
16	xxx.out	Write to Standard Output Device. The string xxx in this file name is replaced by the file name (without extension) of the input command file specified by the user
81	ACRINIT.ACR	ACRi Initialization file
82	ACRIMSG.ACR	ACRi file for diagnostic and error messages, and formats
Auto	zzz_LCNS.ACR	ACRi user License file. The string zzz is replaced by a set of characters that is unique to each user.
Auto	xxx_DEBUG.TMP	User specific diagnostic and debug output ( <b>DEBUG</b> ).
Auto	xxx_GRID_QUALITY.TMP	Grid quality information ( <b>DEBUG GEOMETRY FILE</b> ).
Auto	acr_MTRXxxx.TMP	Solution Matrix Coefficients ( <b>DEBUG MATRIX</b> ). The string xxx is replaced by an identifier for the variable for which the output is generated
Auto	xxx_FLUX.TMP***	Write data for fluxes crossing a sub-region boundary ( <b>FLUX</b> ).
Auto	xxx_DIAGNOSTIC.TMP***	Internally generated file with step-by-step diagnostics of convergence history for each variable for which a transport equation is solved ( <b>DIAGNOSTIC</b> ).
Auto	xxx_HISTORY.TMP	Time-history data file ( <b>HISTORY</b> ).
Auto	xxx_HISTORY_XYZ.TMP	Time-history at specified locations ( <b>HISTORY COORDINATE</b> ).
Auto	xxx_SOURCE.TMP	Time-history of source of variables ( <b>HISTORY SOURCE</b> ).
Auto	xxx_STORAGE.TMP	Time-history of storage of variables ( <b>HISTORY STORAGE</b> ).
Auto	acr_NOZZLE.TMP	Nozzle spray trajectory information ( <b>NOZZLE TRAJECTORY</b> ).
Auto	xxx_PRINT_FORCE.TMP	Time-history of forces or moments on a surface ( <b>PRINT FORCE</b> ).
11	xxx.SAV***	Write data file for restart, archiving and post-processing ( <b>SAVE</b> ).
Auto	xxx_RESTART.TMP***	ACRi restart file ( <b>SAVE RESTART, READ RESTART</b> ).
12	xxx_TABLE.SAV***	Write archive data file in tabular form ( <b>SAVE TABLE</b> ).
Auto	xxx_STATS.TMP	Statistics information pertaining to variables ( <b>STATISTICS</b> ).
Auto	xxx_TRACK.TMP***	Particle Track information for the flow field ( <b>TRACK</b> ).
Auto	acr_XYZ_CRNR.TMP	Corner coordinates for each element ( <b>WRITE CORNERS</b> ).
Auto	xxx_VRTX_MAP.TMP	Vertex Mapping information automatically generated if needed.

\*\*\* The string xxx is replaced by the name of the file (without extension) that is connected to the Standard Output Device (Unit 16).

## 2.5 UNITS OF PHYSICAL QUANTITIES

Any consistent set of units may be employed for input. However, all built-in default values for dimensional physical properties (such as the density of water) are in SI units. If other than SI units are used, then these properties must be specified in appropriate units as part of the input data.

## 2.6 DIAGNOSTIC OUTPUT SPECIFICATION

Four keyword commands control the diagnostic output. These are the **DEBUG**, **DIAGNOSTIC**, **FLUX**, and **OUTPUT** commands. The **DEBUG** command provides a printout of numerical error parameters, which are useful in evaluating the accuracy of numerical solution. The **DIAGNOSTIC** command provides a means to monitor the time-history of the dependent variables at a particular node and the residuals of the governing equations from one iterative step to the next. The **FLUX** command provides output of convective and diffusive flux terms, and the residuals and errors in mass and energy balance in the numerical solution of the equations. The **OUTPUT** command allows the field variables to be written in a tabular format to the standard output device (unit 16).

## 2.7 DEPENDENT VARIABLES OF ACRi™ SOFTWARE TOOLS

ACRi™ SOFTWARE TOOLS provide for the numerical solution of an arbitrary number of coupled transport equations. The user may elect to solve any subset of these equations. Table 2.7.1 lists the variables of the governing equations in the order in which they are solved.

TABLE 2.7.1: DEPENDENT VARIABLES OF ACRi™ SOFTWARE TOOLS

ACRi™ Software	Software Symbol	Mathematical Symbol	Description of the Variable
ANSWER® TIDAL®	U	U	Velocity component in x-direction
ANSWER® TIDAL®	V	V	Velocity component in y or r-direction
ANSWER®	W	W	Velocity component in z or $\theta$ -direction
ANSWER®	PP	$\varphi$	Density and Pressure correction variable
ALL	HT	h or $h_s$	Enthalpy or Stagnation enthalpy (See Note below)
ALL	T	T	Temperature
ANSWER®	K	K	Kinetic energy of turbulence
ANSWER®	E	$\epsilon$	Dissipation of turbulence energy (See Note below)
ANSWER®	FF	$m_c$	Mass fraction of total carbon
ANSWER®	FH	$m_H$	Mass fraction of total hydrogen
ANSWER®	FO	$m_o$	Mass fraction of total oxygen
ANSWER®	FU	$m_{CnHm}$	Mass fraction of fuel species
ANSWER®	CH	$m_{CnHm-2}$	Mass fraction of H <sub>2</sub> -stripped fuel
ANSWER®	CO	$m_{CO}$	Mass fraction of carbon monoxide
ANSWER®	H2	$m_{H2}$	Mass fraction of hydrogen
ANSWER®	RA	$(R_x+R_y+R_z)/3$	Net incoming thermal radiation flux.
ANSWER®	SN	$S_N$	Concentration of soot nuclei
ANSWER®	S1	$S_1$	Concentration of small soot particles
ANSWER®	S2	$S_2$	Concentration of large soot particles
PORFLOW®	P	$P^1$	Pressure head for 1 <sup>st</sup> (primary) fluid phase
PORFLOW®	P2	$P^2$	Pressure head for 2 <sup>nd</sup> fluid phase
PORFLOW®	P3	$P^3$	Pressure head for 3 <sup>rd</sup> fluid phase
TIDAL®	ETA	$\eta$	Fluid elevation above datum
PORFLOW® & TIDAL®	C, C2,C3, C4	$C^k$	Mass concentration of 1 <sup>st</sup> through 4 <sup>th</sup> species in fluid
ALL	User defined	none	Variables defined by the user with the <b>ALLOCATE</b> command or one of the above variables renamed by the user with the <b>RENAME</b> command.

**NOTE:** Many of the input commands refer to the features of the transport equation rather than the variable itself. For these commands the specification of the symbol "T" is taken to be equivalent to that for the enthalpy variable, "H" and that of "L" for the dissipation energy, "E". This for example, is the case for the **SOLVE** command that refers to the solution of the equation or the **SOURCE** command that refers to the source for the equation. This is not true for commands, such as the **SET** command, that specify the value of the variable itself.

## 2.8 TABULAR OUTPUT OF FIELD VARIABLES

The user may obtain tabular output of a number of dependent, independent and supplementary variables. These include the phase space variables listed in Table 2.7.1 plus a number of supplementary variables. Table 2.8.1-3 and 2.8.4 list these variables. ACRi Software Tools define 5 types of variables:

**Node based Variable:** This type of variable is defined at every node of the domain. One value is defined for each element and additionally one value at each boundary location. For a 2D 12 by 10 structured grid (IMAX, JMAX on **GRID** command) there will be a total of 120 nodes. This grid will consist of 80 elements (IMAX-2 times JMAX-2) and 40 boundary nodes. For an unstructured grid the number of total nodes will be the number of elements specified on the **GRID** command plus one node for each external element surface. An external surface is defined as one that is connected only to one element and therefore is not an interface between two elements. All phase space variables listed in Table 2.7.1 fall under this category.

**Element based Variable:** One value is defined for each element. For 12 by 10 structured grid, there will be 80 (10 by 8) values. For an unstructured grid the number of values will be equal to the number of elements specified on the **GRID** command.

**Corner or vertex based Variable:** One value is defined at every vertex of the computational domain. For a 12 by 10 structured grid there will be a total of 99 vertices or corners (IMAX-1 times JMAX-1). For an unstructured grid the vertex coordinate file (**CONNECTIVITY** command) determines the number of vertices.

**Face or Surface based Variable:** One value is defined at every element surface. The total number of faces a function of the grid dimensionality and the manner of arrangement of elements. It is internally computed from the grid specification and connectivity.

**List based Variable:** This type of variables is in the form of a freeform list the length of which is dictated by software and/or user specific needs.

A number of **FREEFORM™** commands are provided to obtain output for these variables in different formats. The primary output of the variables can be obtained by the **OUTPUT**, **SAVE** and **WRITE** commands.

**Output for Node Based Variable:** Table 2.8.1-3 list the node-based variables for which output can be obtained. These are termed “standard” output variables. The output can be written to the Standard Output device (Unit 16) or to a user specified file. The extent and frequency of this output is controlled by the **OUTPUT** and **SAVE** commands.

**Output for Supplementary Variables:** Table 2.8.4 lists some supplementary variables for which output can be obtained. The output for these can be directed to the Standard Output Device or a user-specified file. This output is controlled by the **WRITE** command.

**Output for Derived Flow Variables:** Output for some special variables that are derived from the flow field may be obtained on demand. These variables include, the gradients of the velocity field, the vorticity, stresses, strains, and some special variables. Please see the **OUTPUT** and **SAVE** commands for a detailed description of this output.

**Output for Coefficients and Components of Transport Equation:** Output for different coefficients and components of each transport equations can be obtained during the solution process. Please see the **OUTPUT** and **SAVE** commands for a detailed description of this output.

TABLE 2.8.1: STANDARD VARIABLES OF ANSWER® SOFTWARE TOOL

Software Symbol	Mathematical Symbol	Description of the Variable
U	U	Velocity component in x-direction
V	V	Velocity component in y or r-direction
W	W	Velocity component in z or $\theta$ -direction
P	P	Fluid pressure
HT	h or $h_s$	Enthalpy or Stagnation enthalpy
T	T	Fluid Temperature
K	k	Kinetic energy of turbulence
E	$\epsilon$	Dissipation of turbulence energy
L	$l$	Length scale of turbulence
FF	$m_C$	Mass fraction of total carbon
FH	$m_H$	Mass fraction of total hydrogen
FO	$m_O$	Mass fraction of total oxygen
FU	$m_{CnHm}$	Mass fraction of fuel species
CH	$m_{CnHm-2}$	Mass fraction of H <sub>2</sub> -stripped fuel
CO	$m_{CO}$	Mass fraction of carbon monoxide
H2	$m_{H2}$	Mass fraction of hydrogen
O2	$m_{O2}$	Mass fraction of oxygen
H2O	$m_{H2O}$	Mass fraction of water
CO2	$m_{CO2}$	Mass fraction of carbon dioxide
NO	$M_{NO}$	Mass fraction of Nitrogen Oxides
NO2	$M_{NO2}$	Mass fraction of Nitrogen Di-oxides
FG	G	The property g for the fuel fluctuation equation
FM	$f_m$	The mean-mixture fraction for for the fuel fluctuation equation
PV	$f_v$	The mixture variance for the fuel fluctuation equation
SN	$S_N$	Concentration of soot nuclei
LF		Liquid mass fraction from the nozzle injections
AD		Average droplet diameter for nozzle injections
S1	$S_1$	Concentration of small soot particles
S2	$S_2$	Concentration of large soot particles
RA	$(R_x+R_y+R_z)/3$	Net incoming thermal radiation flux.
RGAS	R	The specific gas constant for the gaseous mixture
MU	$\mu$	Effective viscosity of fluid
RHO	$\rho$	Mass density of fluid
POR	$\Theta_E$	Effective porosity of a grid element
PHYD		Hydraulic head of the fluid
EBYK	$\epsilon/k$	Turbulence time constant
CP	$C_p$	Specific Heat of fluid mixture
SPEED	V	Modulus of Velocity = $\text{Sqrt}(u^2+v^2+w^2)$
EDYN	$0.5 V^2$	Dynamic energy of fluid
PDYN	$0.5 \rho V^2$	Dynamic pressure of fluid
PTOTAL	$P+0.5 V^2$	Total pressure of fluid
MACH	$c/a$	Mach Number for the fluid = $V/\text{sqrt}[(\gamma-1)C_pT]$
VOL	$\delta V$	Volume of the Element
X, Y, Z	x, y, z	X, Y and Z Coordinates of the Node Location
User-Specific	---	User-specific variables defined in the <a href="#">acrNAMES.ACR</a> file
Dynamically Allocated	---	Dynamically allocated variables with <b>ALLOCATE</b> or <b>RENAME</b> commands

TABLE 2.8.2: STANDARD VARIABLES OF PORFLOW® SOFTWARE TOOL

Software Symbol	Mathematical Symbol	Description of the Variable
U	U	Velocity component in x-direction
V	V	Velocity component in y or r-direction
W	W	Velocity component in z or $\theta$ -direction
P	P	Fluid pressure
HT	h or $h_s$	Enthalpy or Stagnation enthalpy
T	T	Fluid Temperature
POR	$\Theta_E$	Effective porosity of a grid element
PORT	$\Theta_T$	Total porosity of a grid element
PORD	$\Theta_D$	Diffusional porosity of a grid element
PHYD		Hydraulic head of the fluid
CP	$C_p$	Specific Heat of fluid mixture
SPEED	V	Modulus of Velocity = $\text{Sqrt}(u^2+v^2+w^2)$
EDYN	$0.5 V^2$	Dynamic energy of fluid
PDYN	$0.5 \rho V^2$	Dynamic pressure of fluid
PTOTAL	$P+0.5 V^2$	Total pressure of fluid
MACH	$c/a$	Mach Number for the fluid = $V/\text{sqrt}[(\gamma-1)C_pT]$
C, C2,C3, C4	$C^k$	Mass concentration of 1 <sup>st</sup> through 4 <sup>th</sup> species in fluid
S, S2, S3	$S^1, S^2, S^3$	Saturation fraction for the 1 <sup>st</sup> , 2 <sup>nd</sup> and 3 <sup>rd</sup> fluid phase
RHO, RHO2,	$R^1, R^2$	Normalized Density of the 1 <sup>st</sup> and 2 <sup>nd</sup> fluid phase
RHOG	$R^3$	Normalized Density of the 3 <sup>rd</sup> (gas) fluid phase
RHOS	$\rho_s$	Mass density of solid matrix
H, H2, H3	$H^1, H^2, H^3$	Total hydraulic head for the 1 <sup>st</sup> , 2 <sup>nd</sup> and 3 <sup>rd</sup> fluid phase
MOIS	$\Phi_E$	Effective moisture in the first fluid phase
MU	$\mu/\mu^*$	Viscosity ratio for the primary Fluid
QX, QY, QZ		X, Y and Z components of element-average Darcy Velocity
VOL	$\delta V$	Volume of the Element
X, Y, Z	x, y, z	X, Y and Z Coordinates of the Node Location
User defined	---	User-defined variables with <b>ALLOCATE</b> or <b>RENAME</b> commands

**TABLE 2.8.3: STANDARD VARIABLES OF TIDAL® SOFTWARE TOOL**

Software Symbol	Mathematical Symbol	Description of the Variable
U	U	Velocity component in x-direction
V	V	Velocity component in y or r-direction
W	W	Velocity component in z or $\theta$ -direction
HT	h or $h_s$	Enthalpy or Stagnation enthalpy
T	T	Fluid Temperature
RHO	$\rho$	Mass density of fluid
CP	$C_p$	Specific Heat of fluid mixture
SPEED	V	Modulus of Velocity = $\text{Sqrt}(u^2+v^2+w^2)$
EDYN	$0.5 V^2$	Dynamic energy of fluid
PDYN	$0.5 \rho V^2$	Dynamic pressure of fluid
PTOTAL	$P+0.5 V^2$	Total pressure of fluid
C, C2,C3, C4	$C^k$	Mass concentration of 1 <sup>st</sup> through 4 <sup>th</sup> species in fluid
ETA	$\eta$	Fluid elevation above datum
H	H	Total fluid depth = $h + \eta$
DEPT	h	Bathymetric depth below datum
PA	$P_a$	Atmospheric pressure
WIND	w	Wind speed
ANGL	$\Theta$	Angle that the wind vector makes with the positive direction of x
VOL	$\delta V$	Volume of the Element
X, Y, Z	x, y, z	X, Y and Z Coordinates of the Node Location
User defined	---	User-defined variables with <b>ALLOCATE</b> or <b>RENAME</b> commands

**TABLE 2.8.4: SUPPLEMENTARY VARIABLES OF ACRi™ SOFTWARE TOOLS**

Software Symbol	Variable Basis	Variable Type	Description of the Variable
XC, YC, ZC	Vertex	Real	x, y and z coordinates for element vertices
VOLF	Face	Real	Volume associated with an element interface. This consists of the sum of the part volumes of the two adjoining elements. For example, in Figure 4.1.5, for the “e” face, it consists of the part volumes of the two elements contained between P and E.
VOLR	Face	Real	Non-dimensional fraction of face volume associated with the “-ve” side of the interface to VOLF. The “-ve” side is defined with reference to the area vector at the face. This definition of direction is arbitrary and depends on the manner in which the elements are processed.
AFX, AFY, AFZ	Face	Real	The x, y and z components of the area vector for an element face.
FC	Face	Real	Convective mass flux across an element interface.
FD	Face	Real	Coefficient for diffusive flux across an interface. For an orthogonal grid, the diffusive flux is defined as the product of this coefficient and the difference of the variables values between the two nodes associated with the interface (such as P and E for the “e” face in Figure 4.1.5)
NCRN	Vertex	Integer	Element to vertex connectivity
NBRS	List	Integer	Element to node connectivity. For each element, the list contains the node numbers to which the element is connected. The number of members for each element is equal to the number of surfaces for the element.
NFACE	List	Integer	Element to surface number connectivity. For each element, the list contains the surface numbers that comprise the element boundary.
LINK	List	Integer	Face to node number mapping matrix. The list is arranged by surface number. For each surface the list contains 4 members: the two nodes on the “-ve” and “+ve” sides of the surface, and the surface numbers for two neighborhood surfaces unconnected to the surface.
MTYP	Node	Integer	Material Type index

## 2.9 ARCHIVAL AND POST-PROCESSING OUTPUT

Files containing archival and post-processing outputs may be obtained in three distinct modes. These files are generated by the **SAVE** command in the **SAVE BLOCK**, **SAVE TABLE** and **SAVE RESTART** modes. The output from these may be used to restart a problem or for post-processing functions such as to produce contour, raster, surface or vector plots on a console screen, printer or plotter. The files can be generated at any required frequency.

Whenever output from these commands is written to a file, diagnostic messages appear on the Standard Output Device that identifies the information being transferred to the file. The file is self-documenting. In addition to the values of the field variables, it contains the file identification data, the user identification, the problem title, the time and date of creation, the basic grid information, and the names of variables stored on the archive file.

The **SAVE BLOCK** command provides flexibility in the selection of variables, the output frequency and the nature of data records (formatted or unformatted). In the **SAVE BLOCK** mode, the variables are listed individually and separately for whole of the domain of computation in a block format. The values of the grid coordinates and field variables (see Tables 2.8.1 to 2.8.3) follow the basic problem and data identification. An alphanumeric record precedes each variable field and lists the name of the variable, the time and step number of archival, the data type and the number of values. The archive information is written in either unformatted or formatted records, depending on user specification. Please see the **SAVE** command for further details.

In the **SAVE TABLE** mode of the command generates output that is arranged by rows and columns for the whole or part of the domain of computation. The basic problem and data identification fields are followed by variables listed in a tabular manner for each node individually. Only formatted data records are allowed. Each row of data pertains to a single element and each column contains the value of a variable for that element. The file can be used for archiving or post-processing purposes. Please see the **SAVE** command for further details.

The **SAVE RESTART** mode is strictly meant for restart of a given problem and is generated only in **UNFORMATTED** mode at the specified frequency.

## 2.10 RESTART OUTPUT

A special file for restart of a problem can be generated by the **SAVE** command with the **RESTART** modifier. This file is in an unformatted format and is machine specific. This file can only be read with the **READ** command with a **RESTART** modifier. If the simulations are restarted from this file, then the problem specification must be identical to that for the original simulation.

## 2.11 MODIFICATION OF INPUT DATA DURING SIMULATIONS

The simulation of a problem is initiated by the **SOLVE** command. The total span of simulations, however, may be divided into segments, and a **SOLVE** command used for each segment. Any time-dependent or sequential aspects of the input or output requirements may be changed between the calculation segments. For any problem, all specification relating to the geometry is considered independent of time; the rest of the problem specification, including operational control and output requirements, can all change during the course of simulations.

An example of a two-segment calculation sequence is given in Table 2.11.1. In this example, the output requirements for both the archive file and the tabular output are changed after 5 time units of simulation.

**TABLE 2.11.1: ILLUSTRATION OF A 2-SEGMENT SIMULATION SEQUENCE**

```
TITLE EXAMPLE OF A TWO SEGMENT CALCULATION WITH OUTPUT CHANGES
////////////////////////////////////
GRID 11 BY 12
COORDinate X: RANGE = 100, grid spacing increment ratio = 1.1
COORDinate Y: RANGE = 40, grid spacing increment ratio = 1.
/
INLEt at X- boundary
OUTLEt at X+ boundary
/
SET U = 1. everywhere
SET T = 500 everywhere      !! except at inlet as modified below
INITial T = 300 from (1,1) to (1,11)  !! INLEt
/
OUTPut for P (pressure) at this stage  !! Print initial values of P
/
//////////////////////////////////// Comment. Start of first segment of calculations
SOLVE for 10 secs in time step of 0.1 sec
/
OUTPut for variables P and T
SAVE variables U, V, P
/
//////////////////////////////////// Comment. Start of second segment of calculations
SOLVe for 10 secs in time step of 0.5 sec
/
OUTPut for variables U, V, W, P and T
SAVE variables U, V, W, P and T
/
END
```

(This page left intentionally blank)

## CHAPTER 3

## KEYWORD COMMAND NOTATION

This chapter describes the notation used for the keyword commands that comprise the user interface for all ACRi Software Tools including ANSWER® PORFLOW® and TIDAL® software package. This interface is based on the ACRi format-free command language FREEFORM™. Knowledge of the structure and syntax of this language is essential for understanding the descriptions of keyword commands in this chapter. The command language is fully described in Chapter 1 of this document. Descriptive notation for the keyword commands is explained in this chapter.

### 3.1 KEYWORD COMMAND NOTATION

The notation for input commands is described below. A detailed description of the user input follows. The **FREEFORM™** command language used for this input is described in Chapter 1.

<b>BOLD</b>	Uppercase characters in bold typeface denote The <b>FREEFORM™ keyword commands</b> . The user may specify the keyword character string in upper or lower case. Bold typeface is used in this manual for emphasis only; it must not be used as part of user input.
<b>CAPS</b>	Upper case characters in normal or bold typeface denote <b>modifiers</b> of <b>FREEFORM™</b> that are <i>significant</i> for interpretation of user input. The user may specify the character string in upper or lower case.
char	Lower-case characters denote information in commands which is <i>not significant</i> for interpreting user input but improves the clarity or readability of the input. The character strings shown may or may not be specified by the user, or other character strings may replace them.
	Vertical bar indicates a choice; only one of the items separated by the bar (and enclosed in braces or square brackets) may be specified.
{ }	Braces indicate that the enclosed item (or one of the enclosed items separated from others by vertical bars) is required and must be specified.
[ ]	Square brackets indicate that the enclosed item is optional.
....	Ellipses (in horizontal or vertical format) indicate that other, similar items may follow those shown.
<b>Nn</b>	The n <sup>th</sup> numeric field in an input command denoted by a keyword.
<b>fname</b>	The name of a file or device; see Section 3.3.
<b>subrgn</b>	The sub region for the applicable input; see Section 3.4.
<b>idsub</b>	A unique identity for a sub region of the domain of computation; see Section 3.4.
<b>func</b>	An analytic or tabular function for input of a numerical quantity; see Section 4.1.
<b>phase</b>	Fluid phase to which the input applies; see Section 3.6.
<b>V<sub>frq</sub></b>	Frequency of output; see Section 3.7.

### 3.2 INTERACTIVE OR RUN TIME INPUT

The **FREEFORM™** command language allows interactive or run time data input for any keyword command. Occurrence of a question mark (?) in the input is a prompt for interactive input by the user. The user may therefore insert a '?' for a numerical or character string in the input. When a '?' is read, the command interpreter pauses and prompts the user for interactive input. Any input so specified is then appended to the original input command string, starting with the location of the question-mark symbol. Additional information about this input feature is given in Chapter 1. The **FREEFORM™** command interpreter is based on a 256-character input string for each input record (see Appendix B). Therefore, the number of characters that can be inserted after the question mark depends on the location of the question mark.

### 3.3 FILE NAME SPECIFICATION

Many input commands allow part or all of the input data to be read from a user-specified file (or device) or output to be directed to an output file. The name of the file, which is denoted by 'fname' in this manual, is specified as a character expression that **must be enclosed in single or double quotes**. The file or device name may be up to 256 characters long, consisting of any characters accepted by the operating system as a valid I/O file name. In the input mode, the data is read from the file in a format-free mode according to the rules of FORTRAN 90.

**Some operating systems do not distinguish between the lower and upper case file names.** On such systems, for example, a file named 'tmp.fil' is equivalent to a file named 'TMP.FIL' (or some other combination of upper and lower case characters). On such systems, an error may occur if file names are specified that are identical except for the case. **In general, it is recommended that the user should consistently use either upper or lower case characters for file names.**

By default, a file is assumed to be in **"FORMATTED"** mode. Some commands also allow an **"UNFORMATTED"** mode of input or output. If allowed, the **"UNFORMATTED"** modifier on the command selects this mode.

**If a file name is specified on a command, then the file is immediately opened and is available for I/O operations.**

In the output mode, a unique file name must be associated with each type of command that generates an output file. The user may assign a name by explicit specification on a command. Once assigned, the new file name becomes the default name until changed by a subsequent command of the same type. If the file name is different than a previously specified name, then the old file associated with that command is closed and the subsequent output is written to a new file with the specified name. If no file name is specified, then a default name is assigned to each output file (see Section 2.4).

**Currently the total number of files that can be opened at any time is limited to 256. The actual number of user definable files is in the neighborhood of 240 since some of the files are used for internal operations. Some operating systems provide even a lower limit on the number of open files.**

### 3.4 SUB REGION IDENTIFICATION ON AN INPUT COMMAND

Some input commands may specify values for a sub region of the domain. For these commands, the desired sub region must be previously specified by a **LOCATE** or **SELECT** command (in this document “**LOCATE**” is used to identify both these commands). Multiple modes of **LOCATE** command are available to accommodate a diversity of input and output requirements. Each **LOCATE** command identifies a subregion with a unique identifier specified as **ID=idsub**, where ‘**ID**’ in the command must immediately precede the **idsub** identifier. The **idsub** identifier may consist of any combination of **up to 32 unique alphanumeric characters** without any embedded spaces or graphic characters. The valid characters are the same as those for a modifier (see Section 1.2.2). This **ID=idsub** identifier is then used subsequently on a keyword command to selectively provide input for that subregion.

A **LOCATE** command for a subregion may also be specified without a unique identifier, then this subregion may be identified on a subsequent command by specifying **LOCATE** or **SELECT** as a modifier till the occurrence of the .next **LOCATE** command. *For some commands, a unique subregion identity may be required for proper implementation of the input features; this is discussed in the description of each command.*

If a subregion specification is omitted by the user, but is required by the keyword command, then the input is assumed to be for the total domain of interest, which is relevant for that command. The choices available for the '**subrgn**' modifier are summarized in Table 3.4.1.

**TABLE 3.4.1: VALID SUBREGION IDENTIFICATION MODIFIERS**

<b>subrgn</b>	<b>Interpretation</b>
<b>SELE</b>	The input applies to the most recently defined subregion of the <b>LOCATE</b> or <b>SELECT</b> command. Same as <b>LOCATE</b> modifier.
<b>LOCA</b>	The input applies to the most recently defined subregion of the <b>LOCATE</b> or <b>SELECT</b> command. Same as <b>SELECT</b> modifier.
<b>ID=idsub</b>	The input applies to the subregion that was assigned the identity 'idsub' in a previous <b>LOCATE</b> or <b>SELECT</b> command.

*The **LOCATE** commands are of two types:* (1) those that specify a sub-region within the computational domain and (2) those that specify exclusively a boundary of the domain.

If a **LOCATE** command specifies only a boundary, then only the nodes falling at the boundary are included in the subregion.

However, if a **LOCATE** command specifies an interior subregion, then **by default both the interior field nodes as well as any exterior boundary nodes that are immediately adjacent to the region are included in the definition.** If a **FIELD** modifier is present on a **LOCATE** command, then the exterior boundary nodes are omitted.

Two additional modifiers are available with the subregion identification on an input command (say a **SET** command) to further select a subset of the region specified by the **LOCATE** command. These are the **FIELD** and the “**dir**” modifiers. If a **FIELD** modifier is specified alongside an **ID=idsub** modifier on an input command (say a **SET** command), then the command applies only to the field nodes and not to the exterior boundary nodes included in the **subrgn**. For example, if this modifier is present on a **SET** command, then the values are set at all nodes in the interior of the subregion but not at the boundary. *The **FIELD** modifier is ignored if the **LOCATE** command was specified for a structured grid with the grid index coordinates. In that case the values are set as defined by the (I,J,K) grid indices.* For structured grids, it is also possible to specify a **dir** modifier along with the **ID=idsub** specification on an input command. In this case, only the nodes at the subregion boundary where the outward normal matches the specified **dir** modifier are selected. See the Section 3.5 below for available choices for the **dir** specification.

**The only exception to the above is an input command that specifies a node-by-node input for a subdomain (see, Sections 4.5 and 4.6) for a subregion defined by a **LOCATE** command that specifies a rectangular window based on (I,J,K) grid indices. In this case, the nodes selected are exactly as specified by the (I,J,K) grid indices. For this node-by-node function a **FIELD** modifier either on the corresponding **LOCATE** command or on the input command is ignored. See Sections 4.5 and 4.6 for further details.**

### 3.5 BOUNDARY IDENTIFICATION

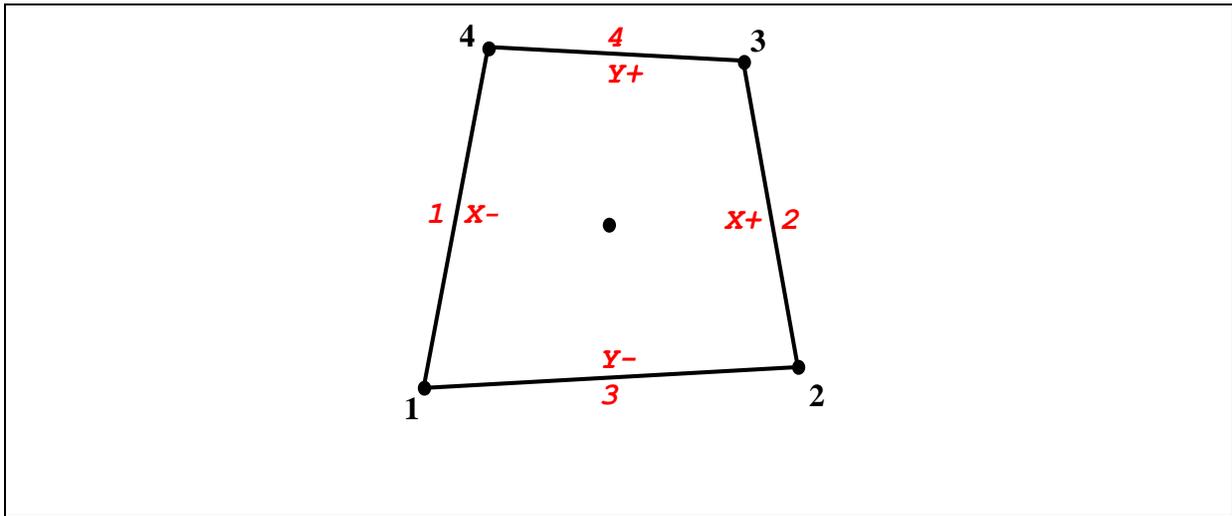
Some of the commands refer to input for the boundaries of the problem. These boundaries may be the external boundaries, which enclose the domain of computation, or these may be internal boundaries that are located within the domain of computation. For these commands the boundary orientation is identified either explicitly on the command by a direction modifier or it is specified as an integral part of the sub-domain specification by the **LOCATE** command with a **PAIRED** list of element and surface numbers.

The identification of the boundary orientation by a direction modifier (**dir**) is illustrated in Table 3.5.1. The modifier consists of a character followed by a sign that represents the direction of the outward normal at the boundary. It should be noted that this direction is a nominal direction with respect to the local orientation of the grid element ( $\xi, \eta, \zeta$ ) for a curvilinear grid). This orientation may or may not be related to the global framework in which the (x, y, z) coordinates of the element vertices are specified. The local orientation for an element is uniquely determined by the manner in which its vertices are numbered. The notation is illustrated in Figure 3.5.1 for both 2D and 3D elements. The vertices are numbered in an anti-clockwise manner in the xy-plane. Briefly, the local x ( $\xi$ ) vector points from vertex 1 to vertex 2, the local y ( $\eta$ ) vector from vertex 1 to vertex 4 and, for 3D elements, the local z ( $\zeta$ ) vector from vertex 1 to vertex 5. For a structured grid the local and global framework are consistent with each other. However, for an unstructured grid, the local framework may change from element to element since the vertex numbering may be completely arbitrary. In such a case a more general description is used which consists of the specification of a paired list of element and surface numbers (see **LOCATE** command with **PAIR** modifier). As illustrated in Figure 3.5.1, the surface numbers for 2D elements vary from 1 to 4 and those for 3D elements from 1 to 6.

**TABLE 3.5.1 BOUNDARY IDENTIFICATION INDEX**

Orientation	Interpretation
<b>X-</b>	The outward normal is in the negative direction of the local x or $\xi$ , coordinate. It is equivalent to the specification of surface number 1 of Figure 3.5.1.
<b>X+</b>	The outward normal is in the positive direction of the local x or $\xi$ , coordinate. It is equivalent to the specification of surface number 2 of Figure 3.5.1
<b>Y-</b>	The outward normal is in the negative direction of the local y or $\eta$ coordinate. It is equivalent to the specification of surface number 3 of Figure 3.5.1
<b>Y+</b>	The outward normal is in the positive direction of the local y or $\eta$ coordinate. It is equivalent to the specification of surface number 4 of Figure 3.5.1
<b>Z-</b>	The outward normal is in the negative direction of the local z or $\zeta$ coordinate. It is equivalent to the specification of surface number 5 of Figure 3.5.1.
<b>Z+</b>	The outward normal is in the positive direction of the local z or $\zeta$ coordinate. It is equivalent to the specification of surface number 6 of Figure 3.5.1.

Vertex and Face Number Nomenclature for a Quadrilateral Element



Vertex and Face number Nomenclature for a Hexahedral Element

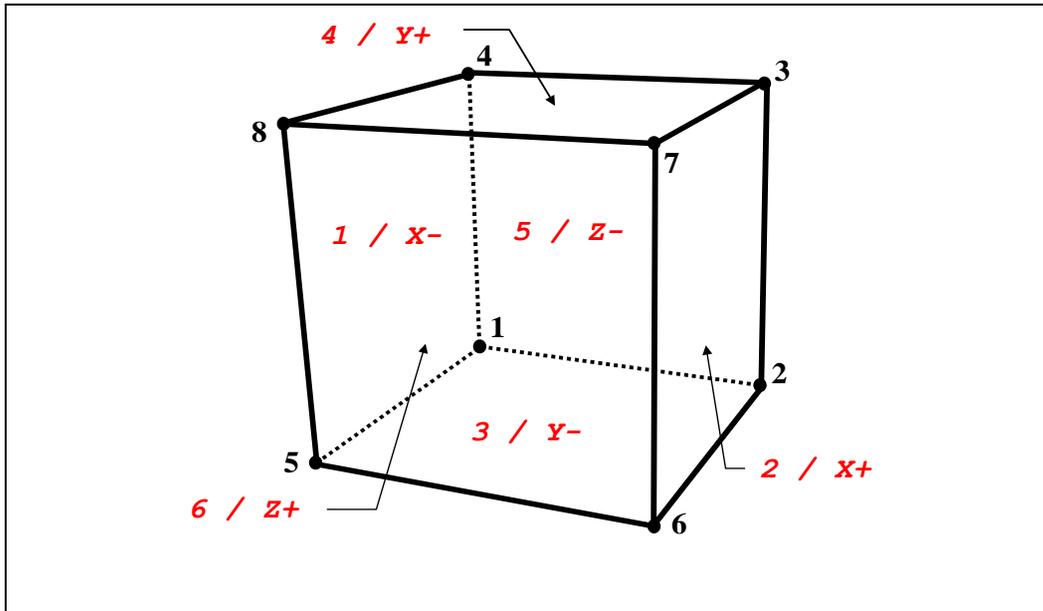


FIGURE 3.5.1: BOUNDARY INDEX NOTATION

### 3.6 SPECIFICATION OF FLUID PHASE

Some of the ACRi software tools can simulate multi-phase flow while other tools are restricted to a single-phase flow. For example, **PORFLOW®** can simulate up to three phases for the fluid. Some of the input commands are used to provide input for one or more of these phases. The modifier that specifies the phase in the command itself is symbolically denoted by 'phase' in this manual. The choices available for this modifier are summarized in Table 3.6.1. [The phase modifier is currently only available with the multi-phase versions of the ANSWER® and PORFLOW® Software Tools; it is not available with TIDAL® series of ACRi software.](#)

**TABLE 3.6.1: VALID MODIFIERS FOR PHASE DESIGNATION**

User Specification of phase	Interpretation For The Corresponding Command
<b>FIRSt</b>	The input is specified for the first phase of the fluid. This is always the default option; if no modifier is specified, then the input is assumed to be for the first phase of the fluid.
<b>SECONd</b>	The input is for the second phase of the fluid.
<b>THIRd</b>	The input is for the third phase of the fluid. <a href="#">This modifier is available only for the PORFLOW® Software Tool.</a>
<b>GAS</b>	The input is for the <b>GAS</b> Phase of the fluid. The gas phase is always the last of the active fluid phases. Thus this modifier is equivalent to the <b>FIRSt</b> modified for a single-phase gas, the <b>SECONd</b> modifier for a liquid-gas, and the <b>THIRd</b> modifier for a liquid-liquid-gas simulation.
<b>VAPO</b>	Same as <b>GAS</b> Modifier

### 3.7 SPECIFICATION OF FREQUENCY FOR OUTPUT

Many of the output commands provide for an option for the user to specify the frequency of output. This input is denoted in this manual by the symbol  $V_{\text{freq}}$  and may be specified in terms of the number of steps or the time interval. The step interval mode is the default mode and in this case, for example, a specification of 10 will result in output at step number 10, 20, 30, etc. The time interval mode is activated if the modifier **TIME** is present on the command. In this case, whether or not the output is obtained exactly at the specified interval, depends on the time step specified by the **SOLVE** command. If an exact time at which the output is due is not simulated, then the output is obtained at the first time step after the due time.

(This section left intentionally blank)

## CHAPTER 4

## FUNCTIONAL INPUT FOR FREEFORM COMMANDS

This chapter describes the built-in and user-specified functions that can be used to specify key components of the FREEFORM input as functions of other built-in and user specified phase space variables. These functions can be applied to initial and boundary conditions, sources and fluid and host matrix properties. Additionally these functions can be used for extensive user specific pre-and post-processing of input data and results obtained from the **ACRi Software Tools**.

#### 4.1 GENERAL FUNCTIONAL RELATIONSHIPS

A number of commands allow the use of a functional form for the required input. The general form of the functional relationship is:

$$\Phi = \Phi(\xi) \quad (4.1.1)$$

here  $\Phi$  represents a field variable, or a source or boundary value for a dependent variable, and represents time, a spatial coordinate or a field variable. A library of built-in tabular and analytic functions for  $\Phi$  is provided. In addition, it is possible to specify user-defined functions as discussed in Section 4.7. The user should note that the library of built-in functions is constantly being updated and additional options may have been incorporated subsequent to the date of publication of this manual. Please contact ACRi for the latest information. In the description of all these functions the symbols  $A_n$ ,  $A$ ,  $B$ ,  $C$ ,  $D$ ,  $E$  and  $F$  represent constants.

##### 4.1.1 The Tabular Function

The most powerful and general option available for this purpose is that of a tabular function in which  $\Phi$  is specified as an arbitrary function of  $\xi$ . The desired value is obtained as a step-wise or piece-wise linear, quadratic or cubic approximation over the values of  $\Phi$  in the neighborhood of the given value of  $\xi$ . This can be expressed as:

$$\text{Given: } \Phi(\xi) = \Phi_n(\xi_n); \quad 1 < n < N$$

$$\Phi(\xi_n + \delta\xi) = \Phi_n + \alpha \left( \frac{\partial\Phi}{\partial\xi} \delta\xi \right)_n + \beta \left( \frac{\partial^2\Phi}{\partial\xi^2} \delta\xi^2 \right)_n + \gamma \left( \frac{\partial^3\Phi}{\partial\xi^3} \delta\xi^3 \right)_n; \quad \xi_n < \xi < \xi_{n+1} \quad (4.1.2)$$

Where  $\alpha=\beta=\gamma=0$  for step wise,  $\alpha=1; \beta=\gamma=0$  for linear,  $\alpha=1; \beta=1/2; \gamma=0$  for quadratic and  $\alpha=1; \beta=1/2; \gamma=1/6$  for cubic interpolation. For the higher order polynomials, the gradients near the boundaries are computed by one-sided Taylor-Series expansion.

For periodic implementation of the tabular function, the value of  $\xi$  used in the above interpolation relation is computed as:

$$\xi_{\text{used}} = \text{mod}[\xi, (\xi_N - \xi_1)], \quad (4.1.3)$$

If extrapolation of the tabular function is specified; then the values are extrapolated by:

$$\Phi(\xi_1 + \delta\xi) = \Phi_1 + \left( \frac{\partial\Phi}{\partial\xi} \delta\xi \right)_1; \quad \xi < \xi_1$$

$$\Phi(\xi_N + \delta\xi) = \Phi_N + \left( \frac{\partial\Phi}{\partial\xi} \delta\xi \right)_N; \quad \xi > \xi_N \quad (4.1.4)$$

##### 4.1.2 Discrete Point-Wise Function

This function allows the specification of a table of values for a subregion such that for each node of the subregion:

$$\Phi_n = A_n; \quad 1 < n \leq N, \quad (4.1.5)$$

where  $A_n$  are arbitrary constants and  $N$  is the total number of nodes in the subregion.

##### 4.1.3 Discrete Point-Wise Time-Dependent Function

This function allows the specification of a table of values for a subregion such that for each node of the subregion the values are a function of time. The general form of the given input is:

$$\Phi_n(t) = A_n; \quad 1 < n \leq N, \quad (4.1.6)$$

where  $A_n$  are arbitrary constants and  $N$  is the total number of nodes in the subregion. The individual values at any given time are computed from sets of values specified at discrete time intervals at  $t=t^1, t^2, \dots, t^m$ . from:

$$\Phi_n^t = \Phi_n^m + \alpha \frac{t - t^m}{t^{m+1} - t^m} (\Phi_n^{m+1} - \Phi_n^m); \quad t^m \leq t < t^{m+1}. \quad (4.1.7)$$

Where the coefficient  $\alpha$  is set to 0 for step-wise and 1 for linear interpolation:

#### 4.1.4 Discrete Point-Wise Random Functions

The Random functions allow the assignment of values to each node of the subregion such that the overall distribution is random without any recognizable patterns. The well-known examples of random numbers are the results of the roll of an ideal die and the sequence of digits that appear in the number  $\pi$ . The set of random numbers generated by the built-in generators is a sequence of Pseudo-Random Numbers (PRN) and should be sufficient for most engineering purposes. If “true” randomness is required then the user is advised to generate such a sequence and specify it through a file with the option defined by Equation 4.1.5 above.

Three distinct PRN options are currently available: these are the Uniform, Normal (Gaussian) and Log-Normal set of random numbers. For set of N values, all the three distributions are all given by::

$$\Phi_n = x_n; \quad 1 < n \leq N, \quad (4.1.8)$$

The Uniform Random number generator uses the “Minimal” Pseudo Random Number Generator of Park and Miller (1988) with Bays-Durham Shuffle. This generator belongs to the group of Linear Congruential Generators (LCG). The cycle repeats after 2,147,483,647 numbers. The uniform random number sequence is defined by a mean value ( $\mu$ ) and a range (R). For a sub-domain with a total of N nodes, the PNR, denoted by  $x_n$ , are then generated by the algorithm and assigned in order of the occurrence of the nodes in the sub-domain according to: The probability density function (pdf) and the cumulative distribution function (cdf) for this distribution are:

$$\text{pdf}(x) = \frac{1}{R}, \quad \mu - \frac{R}{2} \leq x \leq \mu + \frac{R}{2}; \quad 0 \text{ for all other } x \quad (4.1.9)$$

$$\text{cdf}(x) = \frac{x - R/2}{R}, \quad \mu - \frac{R}{2} \leq x \leq \mu + \frac{R}{2}; \quad 0 \text{ if } x < \mu - \frac{R}{2}; \quad 1 \text{ if } x > \mu + \frac{R}{2} \quad (4.1.10)$$

The mean, the median and the variance of the uniform distribution are given by:

$$E(x) = \mu; \quad m(x) = \mu; \quad \text{Var}(x) = \frac{R^2}{12} \quad (4.1.11)$$

The Normal or Gaussian probability density function (pdf) is defined by a mean,  $\mu$ , and a standard deviation,  $\sigma$ . The sequence of normally distributed PNR,  $x_n$ , is generated from the uniformly distributed PNR (see Equations 4.1.9) by the Box-Muller Transformation. With  $x$  as the random variable the pdf and its cdf are given by:

$$\text{pdf}(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right], \quad (4.1.12)$$

$$\text{cdf}(x) = \frac{1}{2} + \frac{1}{2} \text{erf} \frac{x-\mu}{\sqrt{2\sigma^2}}, \quad (4.1.13)$$

The mean, the median and the variance of the normal distribution are given by:

$$E(x) = \mu; \quad m(x) = \mu; \quad \text{Var}(x) = \sigma^2 \quad (4.1.14)$$

:

The Log-normal distribution is the single-tailed probability distribution of a random variable such that the logarithm of the random variable is normally distributed. It is based on the built-in generator for normally distributed PNR described above and uses the exponential of the random numbers to generate a log-normal distribution. The pdf and cdf of the distribution are:

$$\text{pdf}(x) = \frac{1}{x\sqrt{2\pi\sigma^2}} \exp\left[-\frac{\{\ln(x)-\mu\}^2}{2\sigma^2}\right], \quad (4.1.15)$$

$$\text{cdf}(x) = \frac{1}{2} + \frac{1}{2} \operatorname{erf} \frac{\ln(x) - \mu}{\sigma \sqrt{2}}, \quad (4.1.16)$$

The mean,  $E(x)$ , the median,  $m(x)$ , and the variance of the Log-Normal distribution are given by:

$$E(x) = e^{\mu + \sigma^2/2}; \quad m = e^{\mu}; \quad \text{Var}(x) = (e^{\sigma^2} - 1) e^{2\mu + \sigma^2} \quad (4.1.17)$$

The log-normal distribution is also assigned to a sub-domain according to Equation 4.1.12 above.

#### 4.1.5 The Constant Function

$$\Phi = \Phi_0 \quad (4.1.18)$$

#### 4.1.6 Replacement Function

$$\Phi = \xi \quad (4.1.19)$$

#### 4.1.7 Multivariate Functions based on Basic Operations

$$\Phi = \sum_{n=1}^N \xi_n + \Phi_0 \quad (\text{Add Function}) \quad (4.1.20)$$

$$\Phi = \xi_1 - \sum_{n=2}^N \xi_n + \Phi_0 \quad (\text{Subtract Function}) \quad (4.1.21)$$

$$\Phi = C \prod_{n=1}^N \xi_n \quad (\text{Multiply Function}) \quad (4.1.22)$$

$$\Phi = C \xi_1 / \prod_{n=2}^N \xi_n \quad (\text{Divide Function}) \quad (4.1.23)$$

$$\Phi = \Phi_0 - \sum_{n=1}^N \xi_n \quad (\text{Complementary Function}) \quad (4.1.24)$$

#### 4.1.8 Multivariate Polynomial Functions

$$\Phi = A_0 + \sum_{n=1}^N A_n \xi_n; \quad 1 \leq N \leq 7 \quad (4.1.25)$$

$$\Phi = A_0 + \sum_{n=1}^N A_n \xi_n + B_n \xi_n^2; \quad 1 \leq N \leq 7 \quad (4.1.26)$$

$$\Phi = A_0 + \sum_{n=1}^N A_n \xi_n + B_n \xi_n^2 + C_n \xi_n^3; \quad 1 \leq N \leq 7 \quad (4.1.27)$$

#### 4.1.9 The Higher Order Polynomial Functions

$$\Phi = A_0 + A_1 \xi + A_2 \xi^2 + A_3 \xi^3 + A_4 \xi^4 \quad (4.1.28)$$

$$\Phi = A_0 + A_1 \xi + A_2 \xi^2 + A_3 \xi^3 + A_4 \xi^4 + A_5 \xi^5 \quad (4.1.29)$$

$$\Phi = A_0 + A_1 \xi + A_2 \xi^2 + A_3 \xi^3 + A_4 \xi^4 + A_5 \xi^5 + A_6 \xi^6 \quad (4.1.30)$$

$$\Phi = A_0 + A_1 \xi + A_2 \xi^2 + A_3 \xi^3 + A_4 \xi^4 + A_5 \xi^5 + A_6 \xi^6 + A_7 \xi^7 \quad (4.1.31)$$

#### 4.1.10 The Power Functions

$$\Phi = A (\xi + B)^C + D \quad (4.1.32)$$

$$\Phi = A |\xi + B|^C + D \quad (4.1.33)$$

#### 4.1.11 The Exponential Functions

$$\Phi = A \exp(B\xi + C) + D \quad (4.1.34)$$

$$\Phi = \sum_{n=1}^N A_n \exp(B_n \xi + C_n) + D \quad (4.1.35)$$

#### 4.1.12 The Logarithmic Functions

$$\Phi = A \ln(B\xi + C) + D \quad (4.1.36)$$

$$\Phi = \sum_{n=1}^N A_n \ln(B_n \xi + C_n) + D \quad (4.1.37)$$

**4.1.13 The Trigonometric Functions**

$$\Phi = A \text{Sin} (B\xi + C) + D \quad (4.1.38)$$

$$\Phi = \sum_1^N A_n \text{Sin} \left( 2\pi \frac{\xi+C_n}{B_n} \right) + D \quad (4.1.39)$$

$$\Phi = A \text{Cos} (B\xi + C) + D \quad (4.1.40)$$

$$\Phi = \sum_1^N A_n \text{Cos} \left( 2\pi \frac{\xi+C_n}{B_n} \right) + D \quad (4.1.41)$$

$$\Phi = A \text{Tan} (B\xi + C) + D \quad (4.1.42)$$

$$\Phi = \sum_1^N A_n \text{Tan} \left( 2\pi \frac{\xi+C_n}{B_n} \right) + D \quad (4.1.43)$$

$$\Phi = A \text{arcSin} (B\xi + C) + D \quad (4.1.44)$$

$$\Phi = \sum_1^N A_n \text{arcSin} \left( 2\pi \frac{\xi+C_n}{B_n} \right) + D \quad (4.1.45)$$

$$\Phi = A \text{arcCos} (B\xi + C) + D \quad (4.1.46)$$

$$\Phi = \sum_1^N A_n \text{arcCos} \left( 2\pi \frac{\xi+C_n}{B_n} \right) + D \quad (4.1.47)$$

$$\Phi = A \text{arcTan} (B\xi + C) + D \quad (4.1.48)$$

$$\Phi = \sum_1^N A_n \text{arcTan} \left( 2\pi \frac{\xi+C_n}{B_n} \right) + D \quad (4.1.49)$$

**4.1.14 Fourier Series**

$$\Phi = \frac{a_0}{2} + \sum_1^N \left[ a_n \text{Cos} \frac{n\pi(\xi-\xi_0)}{L} + b_n \text{Sin} \frac{n\pi(\xi-\xi_0)}{L} \right] \quad (4.1.50)$$

**4.1.15 Hyperbolic Functions**

$$\Phi = A \text{Sinh}[B\xi + C] + D \quad (4.1.51)$$

$$\Phi = A \text{Cosh}[B\xi + C] + D \quad (4.1.52)$$

$$\Phi = A \text{Tanh}[B\xi + C] + D \quad (4.1.53)$$

**4.1.16 Rational Functions**

$$\Phi = \frac{A+B\xi}{C+D\xi} \quad (4.1.54)$$

$$\Phi = \frac{(A+B\xi)^E}{(C+D\xi)^F} \quad (4.1.55)$$

**4.1.17 Power of Ten Function**

$$\Phi = A 10^{\frac{B}{\xi+C}} + D \quad (4.1.56)$$

**4.1.18 Inverse Exponential Function**

$$\Phi = A \exp \left( \frac{B}{\xi+C} \right) + D \quad (4.1.57)$$

**4.1.19 The Error Function**

The error function is defined as:

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt \quad (4.1.58)$$

The complimentary and inverse error functions are defined as:

$$\operatorname{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-t^2} dt = 1 - \operatorname{erf}(x) \quad (4.1.59)$$

$$\operatorname{erf}[\operatorname{erf}^{-1}(x)] = x \quad (4.1.60)$$

The error function is an odd function, so that:

$$\operatorname{erf}(-x) = -\operatorname{erf}(x) \quad (4.1.61)$$

Two options are provided to approximate this function. The first one is that given by Abramowitz and Stegan (*Handbook of Mathematical Functions, 9th printing, Section 7.1.28, p 299*):

$$\operatorname{erf}(x) \approx 1 - \frac{1}{1 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4 + a_5 x^5 + a_6 x^6}; \quad x \geq 0 \quad (4.1.62)$$

With  $a_1=0.0705230784$ ,  $a_2=0.0422820123$ ,  $a_3=0.0092705272$ ,  $a_4=0.0001520143$ ,  $a_5=0.0002765672$  and  $a_6=0.0000430638$ , the maximum error of this approximation is  $3 \cdot 10^{-7}$ . The second approximation is that from Sergei Winitzki (*A handy approximation for the error function and its inverse, 2008*):

$$\operatorname{erf}(x) \approx \operatorname{sgn}(x) \sqrt{1 - \exp(-x^2 \frac{a + ax^2}{1 + ax^2})}; \quad a = 0.147 \quad (4.1.63)$$

This is designed to be very accurate in a neighborhood of 0 and a neighborhood of infinity, and the error is less than 0.00012. This approximation provides the inverse as:

$$\operatorname{erf}^{-1}(x) \approx \operatorname{sgn}(x) \left[ \sqrt{\left( \frac{2}{\pi a} + \frac{\ln(1-x^2)}{2} \right)^2 - \frac{\ln(1-x^2)}{2}} - \left( \frac{2}{\pi a} + \frac{\ln(1-x^2)}{2} \right) \right]^{1/2} \quad (4.1.64)$$

**4.1.20 Bessel Function of First Kind**

Bessel function is defined as the solution of Bessel's equation:

$$x^2 \frac{d^2 y}{dx^2} + x \frac{dy}{dx} + (x^2 - \alpha^2)y = 0 \quad (4.1.65)$$

The Bessel function of the first kind of order  $\alpha$ , where  $\alpha$  is an integer, may be computed from:

$$J_\alpha(x) = \frac{1}{\pi} \int_0^\pi \cos(\alpha\theta - x \sin\theta) d\theta \quad (4.1.66)$$

For  $x \gg |\alpha^2 - 1/4|$ , this may be approximated by:

$$J_\alpha(x) = \sqrt{\frac{2}{\pi x}} \cos\left(x - \frac{\alpha\pi}{2} - \frac{\pi}{4}\right) \quad (4.1.67)$$

**4.1.21 Gamma Function ( $x > 0$ )**

The gamma function is an extension of the factorial function, with its argument shifted down by 1, to real and complex numbers with a positive real part. i.e., if  $n$  is a positive integer:

$$\Gamma(n) = (n - 1)! \quad (4.1.68)$$

Gamma function can be defined for all complex numbers except the non-positive integers; it is defined as an improper integral that converges only for complex number with a positive real part.

Gamma function can be defined as a definite integral for  $R[x] > 0$  (Euler's integral form)

$$\Gamma(x) = \frac{1}{\sqrt{\pi}} \int_0^\infty t^{x-1} e^{-t} dt \quad (4.1.69)$$

The approximation used here is that by Lanczos (*Numerical Recipes in C, by Teukolsky, Vetterling, Flannery, and Page No: 214*):

$$\Gamma(x + 1) = x \Gamma(x) = \sqrt{2\pi} (x + \gamma + 0.5)^{x+0.5} e^{-(x+\gamma+0.5)} \left[ C_0 + \frac{C_1}{x+1} + \frac{C_2}{x+2} + \dots + \frac{C_N}{x+N} \right]; x > 0 \quad (4.1.70)$$

In this approximation,  $\gamma= 5$ ,  $N=6$ ,  $C_0=1$ ,  $C_1=6.18009172947146$ ,  $C_2=86.50532032941677$ ,  $C_3 = 24.01409824083091$ ,  $C_4 = -1.231739572450155$ ,  $C_5= 0.001208650973866179$ ,  $C_6= -0.000005395239384953$

**4.1.22 Gate Function**

$$\Phi = \beta AB \sqrt{|\xi - C|}; \beta = 1 \text{ if } \xi > C; \text{ else } \beta = 1 \quad (4.1.71)$$

**4.1.23 Sutherland Function**

$$\Phi = A \frac{B+C}{\xi+C} (\xi/B)^{3/2} \quad (4.1.72)$$

**4.1.24 Multivariate Linear Sum**

$$\Phi = \sum_{n=1}^N \xi_n + \Phi_0 \quad (4.1.73)$$

$$\Phi = \sum_{n=1}^N A_n \xi_n + A_{N+1} \quad (4.1.74)$$

**4.1.25 Multivariate Square Sum**

$$\Phi = \sum_{n=1}^N \xi_n^2 + \Phi_0 \quad (4.1.75)$$

$$\Phi = \sum_{n=1}^N A_n \xi_n^2 + A_{N+1} \quad (4.1.76)$$

**4.1.26 Multivariate Square Root of Sum of Squares**

$$\Phi = \sqrt{\sum_{n=1}^N \xi_n^2} + \Phi_0 \quad (4.1.77)$$

$$\Phi = \sqrt{\sum_{n=1}^N A_n \xi_n^2} + A_{N+1} \quad (4.1.78)$$

## 4.2 COMMAND INPUT OF FUNCTIONAL FORM

The functional form of input can be used with a number of commands. Currently these include the **ABSORPTIVITY, BOREHOLE, BOUNDARY, CONDUCTION, DENSITY, DIFFUSION, DISTRIBUTION, FLOW, FRACTURE, GAS, RADIATION, RETARDATION, SET, SOURCE, SPECIFIC** heat, **STORAGE, SWIRL, TRANSFER**, and **VISCOSITY** commands. These commands are described in detail in the following sections. However, the functional specification for all these commands is identical.

Modifiers on the input command select the specific functional relation and the independent variable. These modifiers are referred to as "**func**" and  $\xi$ , respectively. Valid input options for **func** and their corresponding interpretations are given in Tables 4.2.1. Valid input options for independent variable  $\xi$  are listed in Tables 2.8.1-3. For functions that depend on only a single variable, one of the variables listed in Table 4.2.2 is also allowed as an independent variable.

The numerical input required depends on the type of function selected. For numerical input, these functions may be divided into four categories. The nature of the required input for each category is summarized in Tables 4.2.4 through 4.2.6. **By default, the value of the function at a location is computed from the value of the independent variable,  $\xi$  at the same location. However, if the modifier "STAC" is present on a command then the function is evaluated with the value of  $\xi$  at the most recent location specified by the STACK command.**

**Most of the commands described above (except the tabular and the discrete point-wise functions) have the additional option that the function may be truncated to retain only the positive or the negative values. This is achieved by specifying the modifiers "POSITIVE" and "NEGATIVE" with the functional relation.**

**TABLE 4.2.1: VALID INPUT FUNCTIONAL FORMS**

Function	Option	Dependent Variable Computed From
TABL		Equation 4.1.2 with STEP, LINE, QUAD or CUBI modifiers
TABL	PERI	Equations 4.1.2 and 4.1.3 with STEP, LINE, QUAD or CUBI modifiers
TABL	EXTR	Equations 4.1.2 and 4.1.4 with STEP or LINE or QUAD modifiers
NODE		Equation 4.1.5
NODE	TIME	Equation 4.1.7 with STEP ( $\alpha=0$ ) or LINE ( $\alpha=1$ ) modifier
RANDOM	UNIF	Equation 4.1.9
RANDOM	GAUS	Equation 4.1.12
RANDOM	LOG	Equation 4.1.15
CONS		Equation 4.1.18
EQUAL		Equation 4.1.19
ADD		Equation 4.1.20
SUBT		Equation 4.1.21
MULT		Equation 4.1.22
DIVI		Equation 4.1.23
COMP		Equation 4.1.24
LINE		Equation 4.1.25
CUBI		Equation 4.1.26
QUAD		Equation 4.1.27
POLY		Equation 4.1.25 through Equation 4.1.31
POWE		Equation 4.1.32
POWE	MODU	Equation 4.1.33
EXP or EXPO		Equation 4.1.34
EXP or EXPO	SERles	Equation 4.1.35
LN or LOG		Equation 4.1.36
LN or LOG	SERles	Equation 4.1.37
SIN or SINE		Equation 4.1.38
SIN or SINE	SERles	Equation 4.1.39
COS or COSI		Equation 4.1.40
COS or COSI	SERles	Equation 4.1.41
TAN or TANG		Equation 4.1.42
TAN or TANG	SERles	Equation 4.1.43
ASIN		Equation 4.1.44
ASIN	SERles	Equation 4.1.45
ACOS		Equation 4.1.46
ACOS	SERles	Equation 4.1.47
ATAN		Equation 4.1.48
ATAN	SERI	Equation 4.1.49
FOURIER		Equation 4.1.50
SINH		Equation 4.1.51
COSH		Equation 4.1.52
TANH		Equation 4.1.53
RATI		Equation 4.1.54
RATI		Equation 4.1.55

**TABLE 4.2.1: VALID INPUT FUNCTIONAL FORMS (Continued)**

Function	Option	Dependent Variable Computed From
POWER	TEN	Equation 4.1.56
EXP or EXPO	INVE	Equation 4.1.57
ERRO		Equation 4.1.62
ERRO	EXPO	Equation 4.1.63
ERRO	COMP	Equation 4.1.59 with approximation by Equation 4.1.62 or 4.1.63
ERRO	INVE	Equation 4.1.64
BESS	EXAC	Equation 4.1.66
BESS		Equation 4.1.66 with Equation 4.1.67 approximation
GAMM		Equation 4.1.70
GATE		Equation 4.1.71
SUTHERLAND		Equation 4.1.72
SUM		Equation 4.1.73 or Equation 4.1.74
SQUA		Equation 4.1.75 or Equation 4.1.76
ROOT		Equation 4.1.77 or Equation 4.1.78
USER		User-defined functions. Requires DLL or Source Code interface

**TABLE 4.2.2: VALID INDEPENDENT VARIABLES**

User Specification $\xi$	Denotes Independent Variable
TIME	t
X	Coordinate x
Y or R	Coordinate y (or r for radial geometry)
Z or THETA	Coordinate z (or $\theta$ for radial geometry)
Any of the symbols in Table 2.8.1-3	The corresponding variable listed in Table 2.8.1 -3

**TABLE 4.2.3: NUMERICAL INPUT FOR THE TABLE SPECIFICATION**

Numerical Value	Interpretation
N1	The number of sets of input in the table that are specified by Equation 4.1.2.
N2, ... , Nn	The pairs of values, $\xi_n$ and $\theta_n$ (in that order), for Equation 4.1.2. A total of $2*N1$ values (N1 sets) must be specified.

**TABLE 4.2.4: NUMERICAL INPUT FOR SIMPLE ANALYTIC FUNCTIONS**

Numerical Value	Interpretation
N1, ... , Nn	The constants $A_n$ for Equation 4.1.4 or the constants A through E for the functions defined by Equations 4.1.5 through 4.1.18, as appropriate. The number of values must match those required by the function. For Equation 4.1.4, the number of values must equal the number of nodes in the subregion for which the input is specified. The constant function is specified by a single numerical value. Two values are required for the linear relation, five for the polynomial and four for all other functions.

**TABLE 4.2.5: NUMERICAL INPUT FOR THE SERIES FUNCTIONS**

Numerical Values	Interpretation
N1	The number of sets of input in the series specified by Equations given above with the "SERIES" modifier.
N2, ... , Nn-1	The triplet of values of $A_n$ , $B_n$ and $C_n$ (in that order), for the series functions. A total of $3*N1$ values (N1 sets) must be specified.
Nn	The datum $A_0$ for the function.

**TABLE 4.2.6: NUMERICAL INPUT FOR USER SPECIFIED FUNCTIONS**

Numerical Value	Interpretation
N1	The function identification number assigned by the user. Valid values must lie between 1 and 1024 (inclusive). The user must assign a value. For implementation of the function the control is transferred to the USRFNC module and this identification number is returned to the user as the IDUSR variable. The software does not distinguish between the assigned numbers and the same identification number may be assigned more than once, if so desired.
N2, ... , Nn	The numerical values, if any, which are required by the user to implement the function. These values are returned to the user in the RUSR array when the control is transferred to the USRFNC array.

### 4.3 SPECIFICATION OF ADDITIONAL OPERATIONS FOR FUNCTIONS

Many commands that allow the use of functions further allow the values to be manipulated according to a number of operations that can be specified on the command. The available operations are denoted by the character string “**operation**” on the command that can be replaced by one (or more) of the modifiers summarized in the Table below. These options are available with the **DENSITY**, **SET**, **VISCOSITY** and a number of other commands. The availability for a particular command is indicated by the presence of the “**operation**” on the command.

**TABLE 4.3.1: ADDITIONAL OPERATIONS ON FUNCTIONS**

<b>operation</b>	<b>Operation Performed</b>
<b>REPLACE</b>	The computed value replaces the existing value for that variable. This is the default option.
<b>ADD</b>	The computed value is added to the existing value for that variable.
<b>SUBTRACT</b>	The computed value is subtracted from the existing value for that variable.
<b>MULTIPLY</b>	The computed value multiplies the existing value for that variable.
<b>DIVIDE</b>	The computed value divides the existing value for that variable. A value of 1.E-30 is added to the divisor if it is zero.
<b>ABS</b> or <b>ABSO</b>	The absolute value of the computed function is taken. This modifier may be present along with one of the <b>REPLACE</b> , <b>ADD</b> , <b>SUBTRACT</b> , <b>MULTIPLY</b> or <b>DIVIDE</b> modifiers
<b>POSITIVE</b>	The negative values for the computed function are set of zero This modifier may be present along with one of the <b>REPLACE</b> , <b>ADD</b> , <b>SUBTRACT</b> , <b>MULTIPLY</b> or <b>DIVIDE</b> modifiers
<b>NEGATIVE</b>	The positive values for the computed function are set of zero. This modifier may be present along with one of the <b>REPLACE</b> , <b>ADD</b> , <b>SUBTRACT</b> , <b>MULTIPLY</b> or <b>DIVIDE</b> modifiers

In case of multiple specification of **REPLACE**, **ADD**, **SUBTRACT**, **MULTIPLY** and **DIVIDE** modifiers, the first specification takes precedence over others. Similarly in case of multiple specification of **ABS** or **ABSO**, **POSITIVE** and **NEGATIVE**, the first specification takes precedence over others

4.4 EXAMPLES OF FUNCTIONAL FORM OF INPUT

The Tables 4.4.1 through 4.4.3 below provide a number of illustrative examples. The exact interpretation of the input depends upon the particular command and the attributes associated with the command. For example, whereas the **SET** command specifies the value of the dependent variable, the **SOURCE** command would specify the source for that variable and, the **BOUNDARY** command may specify the value of the variable or that of the flux of the variable.

**TABLE 4.4.1: EXAMPLES OF SIMPLE ANALYTIC FUNCTIONS**

Illustrative Commands
//// Constant boundary temperature of 30. BOUNDARY value at X+ is CONSTANT T = 30.
//// Source for temperature is a constant value for active subregion SOURCe for T: = 100 W/m^3 in currently SELEcted subregion
//// Field values of T set as a power law of pressure in previously identified subregion SET T as POWEr law: [1.020 * (P + 0.) ^ 0.50 + 100] in subregion ID=ZON1
//// Boundary temperature is a sinusoidal function of time BOUNDary value at Y+: function: T = 10. * SIN (0.003 * TIME +0.5) +10.
//// Field values of temperature set as a polynomial function of coordinate X SET T: POLYnomial in X: (10, 0.5, 0, -0.1, 0.)
//// Boundary temperature is a linear function of y coordinate BOUNDary value at X+ boundary LINEar function: T = 30. -0.015 * Y
//// Temperature is a cosine function of time SET T as 100 * COSIne (0.00274 * TIME)
//// Temperature is an exponentially decaying function of time SET T as 100 * EXP (-0.001 * TIME) +100.
//// Source for temperature is a linear function of Pressure SOURCe LINEar function (T = 0. -0.10 * P)
//// Source for temperature is a cosine function of time SOURCe T COSIne function of TIME: 100, 0.00274, 0., +10.5
//// Thermal conductivity for temperature is a linear function of temperature itself CONDuctivity for T LINEar function: 0. -0.10 * T

TABLE 4.4.2: EXAMPLES OF ANALYTIC FUNCTION SERIES

Illustrative Commands
<pre>//// Boundary P is a function of T (temperature) BOUNDary value for P at Y- boundary: COSIne SERLes in T: 3 sets       (0, 5, 1), (100, 10, 1), (200, 20, 3) datum = 10.</pre>
<pre>//// Field values for temperature set as sinusoidal function of Y coordinate SET T as SINE SERLes in Y: 3 sets: (0, 5, 1), (100, 10, 1), (200, 20, 3)</pre>
<pre>//// Source for temperature is a 3 term cosine series in velocity U SOURce T as COSIne SERLes in U: 3 sets       (0, 5, 1), (100, 10, 1), (200, 20, 3) datum = 10.</pre>
<pre>//// Boundary flux for T is a 3 part sinusoidal function of temperature BOUNDary T at Y- boundary FLUX type: SINE SERLes in T: 3 sets       (0., 5, 1), (100, 10, 1), (200, 20, 3) datum= 0., h=1.E-4</pre>
<pre>//// Source for temperature is a function of temperature (itself) SOURce T SINE SERLes in T: 3 sets: (0, 5, 1), (100, 10, 1), (200, 20, 3)</pre>
<pre>//// Source for T as an exponential function of time SOURce T: EXPOntial SERLes in TIME; with 3 terms       (0.1, 1.0) (0.05, 0.1) (0.001, 01), base value = 0.</pre>
<pre>//// Field value of U velocity is set as a function of Temperature SET value for U as COSIne SERLes in T: 3 sets       (0., 5, 1), (100, 10, 1), (200, 20, 3) datum = 10.</pre>
<pre>//// Pressure set as a 24 part cosine series; coefficients are contained in file named 'COSVALS' SET P as COSIne SERLes of T: 24 sets from file 'COSVALS'</pre>

TABLE 4.4.3: EXAMPLES OF TABULAR FUNCTIONS

Illustrative Commands
<p>//// The value of P at the boundary is a tabular function of time            BOUNDary P value at Y- boundary: TABLE of values: 3 sets (TIME, value)            (0., 0.01), (100, 0.10), (200, -0.20)</p>
<p>//// The flux of P at the boundary is a tabular function of time            BOUNDary P at X-: FLUX is a TABLE in TIME: 3 sets            (0., 5), (100, 15), (200, 3.5), h=0.001</p>
<p>//// The field values of temperature are set as a tabular function of time            SET T as TABLE of values: 3 sets: (0., 0.01), (100, 0.10), (200, -0.20)</p>
<p>//// As above except that step-wise, rather than the linear, interpolation is used            SET T is STEP wise TABLE: 3 sets: (0., 0.01), (100, 0.10), (200, -0.20)</p>
<p>//// As above except that the values will be repeated after 200 time units            SET T is PERIodic TABLE with STEP wise interpolation:            3 sets: (0., 0.01), (100, 0.10), (200, -0.20)</p>
<p>//// Table with linear interpolation but time input set to result in steep step-like interpolation            SET T for as TABLE of values: with 6 sets: (TIME, value)            (0., 0.0) (1.0000, 0.0) (1.0001, 1.0)            (2., 1.0) (2.0001, 0.5) (3.0000, 0.2)</p>
<p>//// Table with linear interpolation but input from a file            SET T for domain as function of TIME: 20 sets from file 'TIMEVALS'</p>

#### 4.5 NODE BY NODE INPUT FUNCTION

The node by node function allows very high flexibility in that individual values can be specified for each node as described in Section 4.1.2. **ACRi Software Tools** allows for both structured and unstructured grids and input may be provided in either format. The typical input syntax (illustrated with a **SET** command) is:

**SET**             $\{\Phi\}$  {**NODE|STRU|UNST|fname**} [**V<sub>1</sub>...V<sub>m</sub>**] [**subrgn**]

**Φ**                A symbol to denote the variable for which the values are specified. The valid symbols include those listed in Table 2.8.1-3.

**NODE**           The input is specified in a node-by-node manner. By default it is assumed that the input values are given in the **STRUCTURED** mode if the **subrgn** was defined with a **LOCATE** command for a **rectangular window** in terms of (I,J,K) grid indices or the window coordinates. Otherwise it is assumed to be in the **UNSTRUCTURED** mode otherwise.

**STRU**            The input is read in the manner of nested implied FORTRAN DO loops:

**Read (NUNIT, \*) (((VAR (I, J, K), I = ILO, IHI), J = JLO, JHI), K = KLO, KHI)**

Where **NUNIT** is an internally assigned unit number, **VAR** is the variable denoted by **Φ**, and **ILO, IHI, JLO, JHI, KLO, KHI** define, respectively, the starting and ending grid index values for the **subrgn** for a structured grid. For 2D grid, **KLO** and **KHI** are set to unity.

This mode is available only for a structured grid and only if the **subrgn** on the corresponding **LOCATE** command is given in terms of a **rectangular window** with (I,J,K) or grid coordinates. The order of the elements is the same as the one specified (or implied) by the corresponding **LOCATE** command. It should be noted that if the **LOCATE** command was given with a **COOR** modifier, then any exterior boundary nodes adjacent to the selected interior nodes will also be selected. Please see further details of specification of **subrgn** in Section 3.4.

**The order of the nodes, and the nodes included in the defined subrgn may be examined by including a file name on the LOCATE command.**

**UNST**            With this modifier is present, then input is read as a simple implied FORTRAN DO loop:

**Read (NUNIT, \*) (VAR (M), M = MLO, MHI)**

Where **NUNIT** is an internally assigned unit number, **VAR** is the variable denoted by **Φ**, and **MLO and MHI** are the starting and ending element numbers for the **subrgn**. The order of the elements is the same as the one specified (or implied) by the corresponding **LOCATE** command which can be examined by including a file name on the **LOCATE** command.

**fname**           The name of the file for numerical values unless the values are directly specified by **V<sub>1</sub>...V<sub>m</sub>** below. See Section 3.3 for additional information.

**V<sub>1</sub>...V<sub>m</sub>**        The values of the variable at each node selected by the **subrgn** and **dir** modifiers. These values must be present unless **fname** is present. In this case, one of the modifiers, **NODE**, **STRU**, or **UNST** must also be specified.

**subrgn**           The subregion for which the input is specified. If no subregion is specified, then entire computational domain is selected. Note that the modifiers **FIEL** and **dir** are applicable to this input if specified on the command. These modifiers and further details of specification of **subrgn** are given in Section 3.4.

#### EXAMPLES

**SET T** at each subdomain **NODE** from file '**ALLVAL**' input for the entire domain of computation  
**SET T** at subdomain **NODES** for **SELECTED** region from '**ACTIVAL**' in **STRUCTURED** mode  
**SET T** in **ID=RGN1** region from file '**RGN1VAL**' only for the **FIELD** nodes.

#### 4.6 NODE BY NODE INPUT AS A FUNCTION OF TIME

The node by node values may also be set as function of time as described in Section 4.1.3. This generally follows the format of the node by node input (see Section 4.5) The typical input syntax (illustrated by a **SET** command) is:

**SET**                    **{Φ} {TIME} [LINE|STEP] {NODE|STRU|UNST} fname) [subrgn]**

**Φ**                      A symbol to denote the variable for which the values are specified. The valid symbols include those listed in Table 2.8.1-3.

**TIME**                The presence of this modifier assumes that the user is specifying a set of values at each node that are specified as a function of time.

**LINE**                The time step interpolation is linear; *this is the default option.*

**STEP**                The time step interpolation is step function (Equation 4.1.4).

**fname**              The name of the file from which numerical values are read. **See Section B.1.1 for file format.** A file name must be specified; the input cannot be read directly from the command. The number of values for each record at each time step must equal the number of elements for the subdomain; any extra values present in the record are ignored.

**LINE**                The time step interpolation is linear (Equation 4.1.7). *This is the default option.*

**STEP**                The time step interpolation is step function (Equation 4.1.7).

**The interpretation of all other modifiers is the identical to that described in Section 4.5 for node by node input.**

#### REMARKS

---

It is highly recommended that the user examine the input data (for example by running the data set with **SOLVE OFF** command) to make sure that the input has been properly interpreted.

#### EXAMPLES

---

**SET T** as **NODE** by node function of **TIME** rom file '**ALLVAL**' input for the entire domain of computation  
**SET T** for **NODE** as function of **TIME** file='**ACTIVAL**' for **SELECTED** region in **STRUCTURED** mode  
**SET T** at **NODES** function of **TIME ID=RGN1** file= '**RGN1VAL**' only for the **FIELD** nodes.

#### 4.7 SPECIFICATION OF ERROR FUNCTION

The typical input syntax for Error function and its variants (illustrated with a **SET** command) is:

**SET**            **{Φ} {ERRO} [COMP|INVE] } [RATI|EXPO] [subrgn]**

**Φ**                A symbol to denote the independent variable for the Error function. The valid symbols include those listed in Table 2.8.1-3.

**COMP**            By default the specification is assumed to that for the Error function. If this modifier is present then the Complimentary Error Function (Equation 4.1.59) is computed.

**INVE**            By default the specification is assumed to that for the Error function. If this modifier is present then the Inverse of the Error Function (Equation 4.1.60) is computed from Equation 4.1.64.

**RATI**            The rational approximation of Equation 4.1.62 is used to compute the Error Function. This is the default option.

**EXPO**            If this modifier is present then Error Function is computed from the exponential approximation of Equation 4.1.63.

**subrgn**            The subregion for which the input is specified. If no subregion is specified, then entire computational domain is selected. Note that the modifiers **FIEL** and **dir** are applicable to this input if specified on the command. These modifiers and further details of specification of **subrgn** are given in Section 3.4.

#### EXAMPLES

---

**SET T** as **ERROR** function of **X** for **ID=REGION1**

**SET T** as **ERROR** function of **X** by **EXPONENTIAL** approximation for **ID=REGION1**

**SET T** as **INVERSE ERROR** function of **X**

**SET T** as **COMPLIMENTARY ERROR** function of **X**

## 4.8 SPECIFICATION OF BESSEL FUNCTION

The typical input syntax for Bessel function (illustrated with a **SET** command) is:

**SET**            **{Φ} {BESS} [EXAC] [HIGH= Φ<sub>High</sub>] [ORDE=α] [STEP=n] [subrgn]**

- Φ**            A symbol to denote the independent variable for the Bessel function. The valid symbols include those listed in Table 2.8.1-3.
- EXAC**        By default for  $\Phi < \Phi_{\text{High}}$  the value of the function is computed from Equation 4.1.66 and for  $\Phi > \Phi_{\text{High}}$  the asymptotic expression of Equation 4.1.62 is used. If **EXACT** modifier is present then only Equation 4.1.66 is used.
- HIGH**        By default  $\Phi_{\text{High}} = 10 |\alpha^2 - 1/4|$ . If **HIGH** modifier is present then the specified value of  $\Phi_{\text{High}}$  is used. This must be positive value.
- ORDE**        By default for  $\alpha = 0$ . If **ORDER** modifier is present then the specified value of  $\alpha$  is used. This must be a positive integer.
- STEP**        By default for  $n = 18$  which implies that the integral of Equation 4.1.66 is computed in 18 equal values of  $\delta\theta$  in the interval between 0 and  $\pi$ . This results in an interval of 10 degrees for each segment. If **STEP** modifier is present then the specified value of  $n$  is used. This must be a positive integer. It should be noted that the default value results in very high accuracy for  $0 < X < 1$  and moderate accuracy  $0 < X < 10$ . For very high accuracy  $n$  should be approximately equal to  $1/x$ .
- $\Phi_{\text{High}}$       The value beyond which the asymptotic expression is used. This value must be the first numerical value following the **HIGH** modifier. This specification is redundant if the **EXACT** modifier is specified. Default value is  $\Phi_{\text{High}} = 10 |\alpha^2 - 1/4|$ .
- $\alpha$             The order of the Bessel function. This value must be the first numerical value following the **ORDER** modifier. Default value is  $\alpha = 0$ .
- $n$              The number of segments by which the integral is computed. This value must be the first numerical value following the **STEP** modifier. Default value is  $n = 18$ .
- subrgn**      The subregion for which the input is specified. If no subregion is specified, then entire computational domain is selected. Note that the modifiers **FIEL** and **dir** are applicable to this input if specified on the command. These modifiers and further details of specification of **subrgn** are given in Section 3.4.

## EXAMPLES

**SET T** as **BESSEL** function of **X** for **ID=REGION1**

**SET T** as **BESSEL** function of **X** use **EXACT** formulation with **STEPS=100**

**SET T** as **BESSEL** function of **X** use **ORDER=1**, **STEPS=100**, **HIGH=20**

#### 4.9 SPECIFICATION OF GAMMA FUNCTION

The typical input syntax for Gamma function (illustrated with a **SET** command) is:

**SET**            **{ $\Phi$ } {GAMM} [ $\Phi_{High}$ ] [subrgn]**

**$\Phi$**             A symbol to denote the independent variable for the Gamma function. The valid symbols include those listed in Table 2.8.1-3. . **In the current implementation the function is only defined for values of  $\Phi > 0$ . For  $\Phi \leq 0$ , it is set to 0.**

**$\Phi_{High}$**         For  $\Phi > \Phi_{High}$ , the value of the function is set to  $10^{36}$ . **Default value is  $\Phi_{High} = 33$**

**subrgn**         The subregion for which the input is specified. **If no subregion is specified, then entire computational domain is selected.** Note that the modifiers **FIEL** and **dir** are applicable to this input if specified on the command. These modifiers and further details of specification of **subrgn** are given in Section 3.4.

#### EXAMPLES

---

**SET T** as **BESSEL** function of **X** for **ID=REGION1**

**SET T** as **BESSEL** function of **X** use **EXACT** formulation with **STEPS=100**

**SET T** as **BESSEL** function of **X** use **ORDER=1**, **STEPS=100**, **HIGH=20**

#### 4.10 USER-DEFINED INPUT OPTIONS

Some commands permit input of user-defined functions or options. These options are permitted for all the all commands listed in Section 4.2,. The user must have a source-code license to exercise these options. The user in terms of FORTRAN 77 statements in a reserved module called USRFNC may supply this input. If the modifier USER is used in the corresponding command, the user must supply FORTRAN 77 statements, functional formulations or numerical values in the USRFNC module. [Please contact ACRi for implementation of this option.](#)

(This section left intentionally blank)

## CHAPTER 5

## KEYWORD COMMANDS

This chapter describes the keyword commands that comprise the user's interface for all ACRi Software Tools including ANSWER® PORFLOW® and TIDAL® software package. This interface is based on the ACRi format-free command language FREEFORM™. Knowledge of the structure and syntax of this language is essential for understanding the descriptions of keyword commands in this chapter. The command language is fully described in Chapter 1 of this Keyword Command Manual. Descriptive notation for the keyword commands is explained at the beginning of this chapter.

## 5.1 ACRi

**PURPOSE** To select **ACRi Software Tool** to be used for simulation purposes.

**SYNTAX** **ACRi** {**ANSWER** | **PORFLOW** | **TIDAL**}

**ANSWER** The **ANSWER®** Software Tool is used for simulation of the specified problem.

**PORFLOW** The **PORFLOW®** Software Tool is used for simulation of the specified problem.

**ANSWER** The **TIDAL®** Software Tool is used for simulation of the specified problem.

### REMARKS

---

Some installations of **ACRi Software** allow use of multiple tools. Each tool is specialized for a class of fluid flow problems but they collectively cover a wide range of applications in diverse fields. ACRi Software tools are designed so that the optimal tool is automatically selected based on the user input commands. However, there may be certain selections of input commands that can be handled by more than one tool. This command allows the user to select the tool to be used. [The dynamic response through a prompt \(“?”\) character \(See section 1.2.7\) is not allowed for this command. A valid syntax of the command must be present.](#)

### EXAMPLES

---

**ACRi** tool **ANSWER** to be used for this problem.

**ACRi** tool **PORFLOW** to be used for this problem.

**ACRi** tool **TIDAL** to be used for this problem.

## 5.2 ACTIVE

**PURPOSE** To set a default subregion for commands that follow.

**SYNTAX** **ACTI** {**ID=subrgn** | **OFF**}

**ID=subrgn** The subregion that is to become the active default subdomain. The modifier **ID** must be present. This subdomain must have been previously defined with a **LOCATE** or **SELECT** command. See Section 3.4 for further information.

**OFF** Any previous active subdomain is deactivated and the active subdomain is again **reset to be the total computational domain**. This modifier takes precedence over the **ID=subrgn** modifier.

### REMARKS

---

This command sets a default subdomain for all commands that follow till the next **ACTIVE** command is encountered. For a command that requires the specification of a domain, **the default option in ACRi Software Tools is to apply the command to the whole of the computational domain** if a **subrgn** modifier is not explicitly specified on the command itself.

Caution must be exercised in the application of this command, since the new default subdomain will apply to **ALL commands that do not have an explicit subrgn or LOCATE or SELECT specification**.

**The recommended practice** is to apply this command just before a group of commands that require a default subregion and then immediately follow with another **ACTIVE** command with **OFF** modifier to reset the default domain to the whole of the computational domain.

### EXAMPLES

---

**ACTIVE** default subdomain to be set as **ID= LOWER\_SUBDOMAIN**

**ACTIVE** default subdomain **OFF**

**ACTIVE** previous active **ID= LOWER\_SUBDOMAIN** is now **OFF**

## 5.3 ADAPTIVE

**PURPOSE** To compute a new locally refined grid by adaptive methodology from the currently specified grid. The output of this command is an unstructured dataset. This command is effective only for the ANSWER® and PORFLOW® Software Tools.

**SYNTAX** **ADAP** { $\Phi$ } {N1} [casename]

**$\Phi$**  **One, and only one**, symbol for the dependent variable that is used to determine the criterion for adaptive grid refinement. There is no default value.

**N1** Any element where the dependent variable is greater than N1 will be a candidate for being refined.

**casename** A string, enclosed in quotes, that serves as a filename prefix, to hold the coordinates, vertex connectivity, split connectivity, sub-regions, and initial guesses for the new mesh.

**REMARKS**

This command provides a very powerful means to automate the time-consuming gridding process that often precedes the numerical simulation.

A selected variable is used to set a threshold criterion for local grid refinement. If the local value of the variable exceeds the bounds set by the criterion, then the element is split into 4 (for 2D) or 8 (for 3D) elements.

It should be noted that the selected variable does not mean just the phase space variables. Any variable, including those defined by the user can be used. For example, the user can define a combination of gradients of say, velocity, or vorticity, or temperature and density gradients, etc. by appropriate SET commands and then use this defined variable as the criterion for grid refinement.

**EXAMPLES**

/ The first two commands set-up the flag for mesh refinement.

/ First create DUDY.

```
ALLOcate 'DUDY- Gradient of U with Y'
SET DUDY GRADient of U with Y
```

/ Now generate a new unstructured refined dataset.

```
ADAPt mesh wherever DUDY exceeds 1.2 casename: 'Level-1'
```

/ The new connectivity, split connectivity, coordinates, sub-regions, and initial guesses will be generated in files called 'Level-1.cnc', 'Level-1.blk', 'Level-1.xyz', 'Level-1.loc', 'Level-1.init' etc.

## 5.4 ADIABATIC

**PURPOSE** To specify global adiabatic or fixed value boundary conditions at all wall boundaries including internal obstacle walls. This command is effective only for the ANSWER® Software Tools.

**SYNTAX** ADIA [ $\Phi$ ] [OFF]

$\Phi$  The symbol for the dependent variable for which the adiabatic wall conditions are specified. The valid symbols are listed in Table 2.7.1. **One and only one** character string must be specified for each command. If no symbol is specified then the input is assumed to be for the temperature (or enthalpy) variable.

**OFF** The wall is assumed to be a fixed value boundary. The boundary value may be specified by a **BOUNDARY** command or is obtained from default conditions for the variable.

**REMARKS**

By default the velocity components, the temperature and, the turbulence variables at all walls and internal obstacles are fixed. The default wall boundary conditions for the species transport variables, on the other hand, are those of adiabatic or zero flux. The actual wall value may vary from one location to another and can be specified by the **BOUNDARY, INITIAL, READ or SET** commands. The **ADIABATIC** command provides a convenient means to set the zero flux condition globally at walls and obstacles. The boundary conditions at external walls may also be controlled explicitly by the **BOUNDARY** command. For the species transport variables, the **ADIABATIC OFF** command must be used if alternative boundary conditions are to be specified. It should be noted that this command only sets the flux at the wall or the obstacle to zero; it does not update the boundary or obstacle value.

**EXAMPLES**

<b>ADI</b> Abatic conditions	! By default for temperature
<b>ADI</b> Abatic conditions for T at all walls	! Same as above
<b>ADI</b> Abatic OFF for C transport variable	! Change from default to fixed type of walls

## 5.5 ALLOCATE

**PURPOSE** To allocate space for problem-specific input

**MODE 1:** **Allocate Space for A New User Defined Variable**

**SYNTAX** **ALLO** {**name**} [**INTE**]

**name** The name of the user defined variable. It may consist of up to 64 alphanumeric characters of which the **1<sup>st</sup> character must be an alphabetic (A-Z) character**. If more than 32 characters are specified then the name must be enclosed in single or double quotes. If the name is longer than 32 characters but it is not enclosed in quotes then all characters except the first 32 are ignored. **The intended symbol for name must be the first modifier on the ALLOCATE command.**

**The 1<sup>st</sup> string of up to 8 characters is used as the symbolic name** to define input for, or properties of, the new variable. If there are more than 8 characters in the 1<sup>st</sup> string, then the subsequent characters are ignored. **No distinction is made between the lower and upper case characters.** The allowable set of characters includes the alphabetic (A-Z), the numeric (0-9), the underscore (\_), the minus (-), the plus (+) and the period (.) characters. The symbolic name is terminated by the 1<sup>st</sup> separator character. A list of separator characters is given in Chapter 1.

All operations, which can be performed on the built-in default variables, may be performed on this new variable. For example, a transport equation may be solved for this variable. Alternately, it may be used to build compound functions through **SET** commands to define complex boundary conditions and sources for another variable.

**INTE** By default the array allocated is of the FORTRAN type REAL\*8. If this modifier is present then the array allocated is of FORTRAN type INTEGER. If present this modifier must occur after the **name** of the variable.

**REMARKS**

This command provides the user with a powerful tool to dynamically allocate variables, incorporate new physics, and solve new transport equations without FORTRAN or C programming. This, together with the built-in library of dynamic functions (Section 4.1.6), is an important component of the FORTRANless™ technology built into ACRi Software Tools.

The number of variables that can be allocated by the user varies. The default capability is set such that up to 32 variables can be defined by the user. However, if a particular problem uses the memory resources for other purposes, such as large input tables, then the number of allowable user defined variables may be less.

**EXAMPLES**

```

ALLOCATE VARIABL1 ! VARIABL1 will be used as symbol for this variable
ALLOCATE MY_NEW_Variable_that_i_defined ! MY_NEW_V will be used as symbol for this variable
ALLOCATE "VARIABL1 - A New Variable Defined Specifically for this Problem"
ALLOCATE R1 ! A variable to hold reaction rates for other variables
ALLOCATE 'R1 - The reaction rate for the first chemical species'
ALLOCATE "IWORK - A working array" of type INTEGER

```

**MODE 2:**      **Allocate Space for User Input Tables**

**SYNTAX**      **ALLO {TABL} {N1}**

**TABL**          By default 10,000-word memory is allocated to store the problem-specific input values by the user. This command should be used to increase the allocated memory if an error message saying insufficient real table space is encountered.

**N1**              The problem-specific memory space (in words) to be allocated.

**EXAMPLES**

---

**ALLOCATE TABLE** space for 50000 words of problem specific input

## 5.6 ARRHENIUS

**PURPOSE** To specify the Arrhenius reaction rate constants for the default 4-step hydrocarbon reaction. This command is effective only for the ANSWER® Software Tool.

**SYNTAX** **ARRH {Φ} [N1, N2, N3, N4, N5]**

**Φ** One of the character strings: FU, CH, CO and H2. It denotes the dependent variable (Table 2.7.1) for which the Arrhenius reaction rate constants are specified.

**N1** The Arrhenius pre-factor,  $C_A (>0)$ , of rate expression equation\* as given below

$$S_{m_j,k} = C_A S_o \exp(-C_E / T) \rho^{(a+b+c)} m_j^a m_k^b m_l^c$$

**N2** The Arrhenius activation energy constant,  $C_E (\geq 0)$ , as shown in above equation .

**N3, N4, N5** The exponents a, b, and c of the above equation are the primary, secondary and tertiary reaction species.

**TABLE OF DEFAULT VALUES FOR ARRHENIUS COMMAND**

Symbol	N1	N2	N3	N4	N5
FU	2.0893x10 <sup>22</sup>	2.48x10 <sup>4</sup>	0.50	1.07	0.40
CH	5.0117x10 <sup>19</sup>	2.50x10 <sup>4</sup>	0.90	1.18	-0.37
CO	3.9811x10 <sup>19</sup>	2.00x10 <sup>4</sup>	1.00	0.25	0.50
H2	3.3113x10 <sup>18</sup>	2.05x10 <sup>4</sup>	0.85	1.42	-0.56

**EXAMPLES**

**ARRHENIUS** constants for FU: 3.9E11, 1.5E8, 1.0, 1.0, 0.0

\* The Rate Expression equation  
where

$S_{m_j,k}$  is the kinetic rate of reaction of  $j^{\text{th}}$  species,  $m_j$ ,  $m_k$  and  $m_l$  are mass fractions of the participating species,

$C_A$ ,  $C_E$ , a, b and c are empirical constants.

The quantity  $S_o$  is unity for all reactions except that for the CO reaction

## 5.7 BANNER

**PURPOSE** To print a banner page to the output directed to the standard output device (unit 16). The banner page contains the ACRi Software Tool version and date identification, time and date stamp, and user identification.

**SYNTAX** BANN

**EXAMPLES**

---

**BANNER** page output at this stage

## 5.8 BLOCK

**PURPOSE** To define a solid object or blockage within the flow domain. This command is effective only for the ANSWER® and TIDAL® Software Tools.

**MODE 1:** Solid Blocks Immersed in Fluid

**SYNTAX** BLOC {subrgn} [FLUI] [EXCL] [FLOO | DEFA] [ROUG=Z<sub>rough</sub>] [TYPE=C<sub>rough</sub>]

**subrgn** The subregion to be defined as a blocked or solid region. See Section 3.4. The subregion must be interior to the computational domain and must not contain any boundary elements. A subregion must be specified; there is no default value.

**FLUI** By default, a blocked region, once defined is assumed to be permanent. If this modifier is present, then the block is treated as a transient object that may be subsequently removed by a BLOCK OFF command.

**EXCL** By default the fluid saturation, and hence the volume of the fluid, inside a blocked element is set to zero unless the FLUID modifier is present or the SOLID command is given for conjugate heat transfer. If this modifier is present, then the volume of the fluid saturation and the volume of the fluid in the element are forced to be zero.

**FLOO** This mode of the command is operational only for the TIDAL® software. By default, a blocked region is assumed to be solid block with no fluid. If this modifier is present, then the blocked region can be flooded based on the topographical information specified for the block, if the water level rises above the level of the block.

**DEFA** This mode of the command is operational only for the TIDAL® software. If this modifier is present, then the external boundaries of the region are defined to be blocked region. No fluid can enter or exit from these boundaries. By default, the external boundaries are open for flow.

**ROUG** This mode of the command is operational only for the ANSWER® software. If this modifier is present, then the wall of the block is assumed to be hydro-dynamically rough for turbulent flow. The height and type of roughness is specified by the Z<sub>ROUGH</sub> and C<sub>ROUGH</sub> constants (below).

**Z<sub>ROUGH</sub>** The average roughness height for the wall. It should be noted that the roughness should be small compared to the grid size in the vicinity of the wall. Further the roughness theory is only valid for the range of Y+ values that put it in fully turbulent region of the log-law of the wall.

**TYPE** This mode of the command is operational only for the ANSWER® software. This modifier specifies the nature of the roughness. Roughness can be of many types including those of sand-grain, ribbed, finned and random. A factor in the corresponding roughness relations (C<sub>ROUGH</sub>) accounts for these differences. This modifier is effective only if the modifier ROUG is also present.

**C<sub>ROUGH</sub>** The factor that accounts for roughness type. The default value is 0.5 which is generally considered to be suitable for sand-grain type of roughness. The recommended values generally lie between 0.5 and 1.0.

**REMARKS**

Multiple BLOCK commands may be specified to accommodate complex or multiple internal obstacles.

**EXAMPLES**

BLOCK for SELECTED subregion ! Currently active selected subregion  
 BLOCK at subregion ID=BLK1 ! Subregion defined with ID=BLK1  
 BLOCK with FLOODING option for subregion ID=BLK1  
 BLOCK at subregion ID=BLK1 FLUID !Block will be removed later by a BLOCK OFF command  
 BLOCK at subregion ID=BLK1 ROUGH with height=0.001 and TYPE factor =0.6

**MODE 2:** Solid Block Immersed and Moving in Fluid

**SYNTAX** BLOC {subrgn} {MOVE | MOVI} [FLUI] [EXCL] [ROUG=Z<sub>rough</sub>] [TYPE=C<sub>rough</sub>]

**subrgn** The subregion to be defined as a blocked or solid region. See Section 3.4. The subregion must be interior to the computational domain and must not contain any boundary elements. A subregion must be specified; there is no default value.

**MOVE** By default, a blocked region is assumed to be stationary in a moving fluid. If this modifier is present, then the block is treated as a moving object. The velocity components for the moving blocks can be specified by SET, INITIAL or one of the other commands for specification of field values.

**MOVI** Same as MOVE.

**FLUI** See Mode 1 of command.

**EXCL** See Mode 1 of command.

**ROUG** See Mode 1 of command.

**Z<sub>ROUGH</sub>** See Mode 1 of command.

**TYPE** See Mode 1 of command.

**C<sub>ROUGH</sub>** See Mode 1 of command.

#### APPLICABILITY

---

This command mode is available only for the ANSWER® Software Tool. Current implementation of this option is suitable only for small or oscillatory motions. It ignores the local changes in fluid volume due to the motion of block. It is assumed that the volume of the surrounding fluid elements stays unchanged. However full account is taken of the effects of the block motion on the surrounding fluid due to momentum and other interaction. It is equivalent to assuming that the fluid displacement is small compared to the total volume of the fluid.

#### EXAMPLES

---

BLOCKage at subregion ID=BLK1 is MOVIng

**MODE 3:** A Dissolving or Melting Block of Salt or other Material

**SYNTAX** **BLOC** {subrgn} [SALT | MELT] [INSO= $\xi$ ] [ALL | NONE] [ [RATE= $\lambda$ ] | [MULT= $\beta$ ] ] [PLAN] [I  
| J | K=N<sub>IJK</sub>] [AVER=N<sub>avg</sub> | OFFS=N<sub>offset</sub>] [ISOT] [GAMA] [STOR=N<sub>STOR</sub>]  
[ROUG=Z<sub>rough</sub>] [TYPE=C<sub>rough</sub>]

**subrgn** The subregion to be defined as a blocked or solid region. See Section 3.4. The subregion must be interior to the computational domain and must not contain any boundary elements. A subregion must be specified; there is no default value.

**SALT** If this modifier is present, then the block is treated as a block of salt that dissolves in water as a function of time. The density of the resulting brine solution, and the parameters used to dissolve the salt may be specified with **DENSITY SALT** command. The built-in rate of dissolution of salt is computed from a correlation developed at Sandia National Laboratory (Saberian and Podio, 1977). The rate of dissolution of a vertical face in touch with fluid is given by:

$$\frac{dr}{dt} = \alpha_1 \gamma^4 - \alpha_2 \gamma^3 + \alpha_3 \gamma^2 - \alpha_4 \gamma + \alpha_5 - \frac{\alpha_6}{\gamma}$$

Where  $\gamma$  is the specific gravity ( $=\rho/\rho^*$ ) of brine and the superscript \* denotes the reference density. This rate is modified for an inclined surface at an angle of  $\theta$  ( $\theta = 90$  is an upward facing and  $\theta = -90$  is a downward facing surface) by the correlation:

$$\left[ \frac{dr}{dt} \right]_{\theta \geq 0} = \left[ \frac{dr}{dt} \right]_{\theta=0} [\text{Cos}\theta]^{1/2}$$

$$\left[ \frac{dr}{dt} \right]_{\theta < 0} = \left[ \frac{dr}{dt} \right]_{\theta=0} \left[ 1 + 0.22 - 0.22 [1 + \theta/45]^{1/3} \right]$$

The constants for this correlation as recommended in the User's Manual for SANSMIC code (Russo, 1983) are summarized in the Table below.

Units	$\alpha_1$	$\alpha_2$	$\alpha_3$	$\alpha_4$	$\alpha_5$	$\alpha_6$
Ft/hr	45.654996	232.2931	469.5247	470.37554	232.73686	45.203241
m/sx10 <sup>3</sup>	3.8654563	19.667482	39.753091	39.825129	19.705054	3.8272077

The surface dissolution rate drops off very quickly with the specific gravity of brine. The table below summarizes some of these values for a vertical surface.

$\psi = \rho / \rho^*$	1.0	1.05	1.10	1.15	1.20
dr/dt (m/s) x10 <sup>6</sup>	3.783	2.409	1.368	0.5371	0.01146
dr/dt (m/day)	0.327	0.208	0.118	0.046	0.001

**MELT** Same as **SALT**.

**INSO** If this modifier is present, then it is assumed that a fraction of salt block is composed of insoluble material. As salt dissolves the insoluble material dislodges and becomes part of brine. A fraction of this insoluble material settles down to the floor of the cavern and the rest leaves with the outflow. With V as the upward fluid velocity (m/s), the settling fraction by volume, f, is given by the correlation:

$$f = \frac{0.5}{1. + 27.283V} + 0.5 e^{-23.622V}$$

**$\xi$**  The fraction of salt block that is insoluble.

**ALL** By default the fraction of the insolubles that settles is computed from the formula given above. If this modifier is specified, then this fraction is set to unity.

**NONE** None of the insolubles settle to the floor; all are flushed out of with the outgoing brine.

**RATE** If this modifier is present, then the built-in correlation is bypassed and the rate of surface

	dissolution or melting is assumed to be equal to $\lambda$ .
$\lambda$	The specified surface dissolution or melting rate if the modifier <b>RATE</b> is present.
<b>MULT</b>	If this modifier is present, then the rate computed from the built-in correlation is multiplied by the specified factor $\beta$ .
$\beta$	The multiplicative factor for the dissolution or melting rate if the modifier <b>MULT</b> is present.
<b>PLAN</b>	The computation of dissolution rate (dr/dt) requires the location of specific gravity, $\psi$ . By default, $\psi$ is computed at the fluid element next to the wall between the fluid and the solid (salt). If this modifier is present, $\psi$ is taken at a location which corresponds to the element directly in line with the wall element but located at the plane defined by the <b>I</b> , <b>J</b> , <b>K</b> , and <b>N<sub>IJK</sub></b> (below). This modifier can only be used for a structured grid.
<b>I</b>	The plane is defined by the <b>I</b> (x-direction) grid index of a structured grid.
<b>J</b>	The plane is defined by the <b>J</b> (y-direction) grid index of a structured grid.
<b>K</b>	The plane is defined by the <b>K</b> (z-direction) grid index of a structured grid.
<b>N<sub>IJK</sub></b>	The I, J or K location of the plane at which the specific gravity is computed.
<b>AVER</b>	The specific gravity is computed from an average of values prevailing at a number of elements opposite the wall element or with reference to the plane specified by <b>N<sub>IJK</sub></b> and the depth of averaging specified by <b>N<sub>avg</sub></b> .
<b>N<sub>avg</sub></b>	The depth of averaging in the <b>I</b> , <b>J</b> , <b>K</b> direction. In the absence of the <b>PLAN</b> modifier, the averaging starts from the element located next to the salt wall. If the <b>PLAN</b> modifier is present then the averaging starts from the location on the <b>N<sub>IJK</sub></b> plane but in line with the wall element. If <b>N<sub>avg</sub></b> is positive, then the averaging is in the increasing direction of the coordinate index <b>I</b> , <b>J</b> , <b>K</b> . If it is negative then it is in the decreasing direction. If any solid block elements are encountered in the averaging, then the averaging process stops at the last element before the solid element; any elements on the other side of the block are ignored. If <b>N<sub>avg</sub></b> is too large so that it refers to elements beyond the computational domain, then only the points lying on the transect within the computational domain are selected.
<b>OFFS</b>	The specific gravity is computed from a location that is offset to the reference location.
<b>N<sub>offset</sub></b>	The offset in the <b>I</b> , <b>J</b> , <b>K</b> direction. In the absence of the <b>PLAN</b> modifier, the offset refers to the location that is offset in the <b>I</b> , <b>J</b> , <b>K</b> direction from the element located next to the salt wall. If the <b>PLAN</b> modifier is present then the offset refers to the location on the <b>N<sub>IJK</sub></b> plane but in line with the wall element. If <b>N<sub>offset</sub></b> is positive, then the offset is in the increasing direction of the coordinate index <b>I</b> , <b>J</b> , <b>K</b> . If it is negative then it is in the decreasing direction. If any solid block elements are encountered then the last element before the solid is selected. If <b>N<sub>offset</sub></b> is too large so that it refers to elements beyond the computational domain, then the last element within the computational domain is selected.
<b>ISOT</b>	By default, the computed rate for the dissolution of the wet surface is modified for the inclination of the surface to the vertical direction. If this modifier is present, then the rate is as computed from the correlation without any correction for the angle of inclination.
<b>GAMA</b>	By default, the diffusion coefficient for the governing equation is not modified for the elements that consist partially of brine (fluid) and partially of salt (block). This is reasonable given that the diffusion process is dominated by the gradients normal to the wall and is mostly independent of whether the element is partially or totally filled with brine. If this modifier is present, then the diffusion is proportional to the fluid saturation of the element.
<b>STOR</b>	Memory space is automatically allocated to take account of the changing boundaries (wall facets) of a melting block. By default the amount of space allocated is $N * N_{23D}$ where <b>N</b> is number of elements in the blocks and <b>N<sub>23D</sub></b> is 2 for 2D and 3 for 3D flows. This is typically an overly conservative estimate based on the assumption that every alternate element melts leaving a checkerboard of isolated elements. This can be wasteful for problems with orderly melting at a defined surface. If this modifier is specified, then the amount of memory for wall facets is set to <b>N<sub>STOR</sub></b> below.
<b>N<sub>STOR</sub></b>	The amount of storage allocated for the wall segments of the evolving block if the <b>STOR</b> modifier is present. A fatal error occurs if the amount of allocated memory is not sufficient.

<b>ROUG</b>	See Mode 1 of command.
<b>ZROUGH</b>	See Mode 1 of command.
<b>TYPE</b>	See Mode 1 of command.
<b>CROUGH</b>	See Mode 1 of command.

#### APPLICABILITY

---

This command mode is available only for the **ANSWER®** Software Tool.

#### REMARKS

---

The time step for this option must be such that no more than one element (normal to the contact surface) can be fully dissolved in any given time step. If the time step is larger then dissolution will be limited to 1 element. This will appear as an effective slower rate of dissolution than that indicated by the elapsed time.

#### EXAMPLES

---

**BLOCKage** at subregion ID=BLK1 is dissolving SALT  
**BLOCKage** at subregion ID=BLK1 is MELTing at a rate of 0.0001 m/s  
**BLOCKage** SALT subregion ID=BLK1 RATE=0.0001 and STOR 10000 boundary segments.  
**BLOCKage** SALT subregion ID=BLK1 max ISOTropic and MULTiPLY by 100 the built-in rate  
**BLOCKage** SALT subregion ID=BLK1 max INSOLUBLE fraction=0.08 by volume  
**BLOCKage** MELTing subregion ID=BLK1 PLANE J=20, AVERaging depth =10.  
**BLOCKage** MELTing subregion ID=BLK1 OFFSET by J=20.  
**BLOCKage** MELTing subregion ID=BLK1 max STOR=10000 boundary segments.

<b>MODE 4:</b>	<b>A Growing Block with Deposition of Insolubles from a Salt Block</b>
<b>SYNTAX</b>	<b>BLOC {subrgn} [GROW  DEPO] [STOR=N<sub>STOR</sub>] [ROUG=Z<sub>rough</sub>] [TYPE=C<sub>rough</sub>] [DELA=N<sub>delay</sub>] [EXCL= subrgn_ex]</b>
<b>subrgn</b>	The subregion to be defined as a blocked or solid region. See Section 3.4. The subregion must be interior to the computational domain and must not contain any boundary elements. A subregion must be specified; there is no default value.
<b>GROW</b>	The block is treated as a block that receives part of the insoluble material that settles from the dissolving salt blocks. It is assumed that at any time the insolubles fall uniformly on the exposed surface of the block. The elements that have inflow, outflow or mass sources are automatically excluded from deposition.
<b>DEPO</b>	Same as <b>GROW</b> .
<b>STOR</b>	Memory space is automatically allocated to take account of the evolving block elements of a growing block. The default value is set to assume that the block can grow to be five times the number of initial elements in the block. If this modifier is specified, then the maximum number of elements that the block can grow to is set to the initial number of elements in the block plus <b>N<sub>STOR</sub></b> additional elements.
<b>N<sub>STOR</sub></b>	The amount of storage allocated for the maximum number of elements that the block can grow to.
<b>ROUG</b>	See Mode 1 of command.
<b>Z<sub>ROUGH</sub></b>	See Mode 1 of command.
<b>TYPE</b>	See Mode 1 of command.
<b>C<sub>ROUGH</sub></b>	See Mode 1 of command.
<b>DELA</b>	For formation of caverns by salt or other dissolution, often in the early stages of a computation process, enough surface area is not available for deposition. In this case, it is preferable to delay start of deposition unless a minimum number of elements can participate. This modifier is used to specify the number of minimum elements before deposition can start.
<b>N<sub>DELAY</sub></b>	The minimum number of elements before deposition can start. A value of more than 5 is recommended for most cases.
<b>EXCL</b>	Some regions may need to be excluded from the deposition process. For example, it may be necessary to exclude additional regions such as in the immediate vicinity of an inflow boundary so that the inflow is not isolated from contact with other computational regions. This modifier is used to specify the sub domain excluded from receiving any deposition.
<b>Subrgn_ex</b>	The subregion to be excluded from receiving any deposition material. See Section 3.4. The subregion must be interior to the computational domain. A subregion must be specified; there is no default value.

**APPLICABILITY**

This command mode is available only for the **ANSWER®** Software Tool.

**REMARKS**

The time step for this option must be such that no more than one element (normal to the contact surface) can be added in any given time step. If the time step is larger then the growth will be limited to 1 element. This will appear as an effective slower rate of growth than that indicated by the elapsed time.

**EXAMPLES**

**BLOCK**age at subregion ID=BLK1 is GROWing with time  
**BLOCK**age at subregion ID=BLK1 can GROW to be STOR= 1000 max elements.  
**BLOCK**age at subregion ID=BLK1 GROW with DELAY=10, EXCLUDE ID=BLK2

**MODE 5:** Remove Previously Specified Blocks

**SYNTAX** BLOC {subrgn} {OFF}

**subrgn** The subregion to be defined as a blocked or solid region. See Section 3.4. The subregion must be interior to the computational domain and must not contain any boundary elements. A subregion must be specified; there is no default value.

**OFF** A previously specified block with the subregion identified above is removed.

#### EXAMPLES

---

**BLOCKage** ID=BLK1 OFF !Previously specified Block is removed

## 5.9 BOREHOLE

**PURPOSE** Define embedded borehole in the porous matrix. This command is effective only for the PORFLOW® Software Tool.

**MODE 1:** Specification of a Borehole with fixed Diameter

**SYNTAX** **BORE** {ID=subrgn} {D<sub>outer</sub>} [D<sub>inner</sub>] [INCL] [dir | ALL | NONE] [DETA]

**ID** The modifier to indicate that the input is for a previously defined subregion.

**subrgn** The subregion in which the borehole is located. **The subregion must be previously specified by a LOCATE IJK or LOCATE COOR command.**

**D<sub>outer</sub>** The outer diameter of the borehole. **There is no default value; a value must be specified.**

**D<sub>inner</sub>** Inner diameter of the borehole. **By default it is assumed that the inner diameter is zero.**

**INCL** BY default only the boundary faces in the principal direction (determined from the **LOCATE** command) are included in the definition of the borehole faces. If this modifier is present, together then the boundary faces specified are included based on the **dir**, **ALL** or **NONE** modifier.

**dir** One of the direction modifiers (**X-**, **X+**, **Y-**, **Y+**, **Z-**, **Z+**) which indicates the face to be included if the **INCL** modifier is present.

**ALL** All boundary faces in the sub domain are included if the **INCL** modifier is present.

**NONE** No boundary faces in the sub domain are included if the **INCL** modifier is present.

**DETA** If this modifier is present, then a detailed table of the elements and faces that form the fracture is printed out to the Standard Output Device

**REMARKS**

**Each borehole must be specified by a different LOCATE command.** All connected elements in the sub-domain are selected to be part of the borehole.

The effective cross-sectional area of the Borehole is computed as:

$$\text{Area} = (\pi/4) (D_{\text{outer}}^2 - D_{\text{inner}}^2)$$

**EXAMPLES**

**BOREHOLE** for ID=BOREHOLE\_1 dia = 0.2

**BOREHOLE** ID= BOREHOLE\_2 Y: D=0.2, Inner d=0.1 DETAIL

<b>MODE 2:</b>	<b>Specification of a Borehole with Diameter as a Functional Form</b>
<b>SYNTAX</b>	<b>BORE</b> {ID=subrgn} {D <sub>outer</sub> } [func( $\xi$ )] [N <sub>1</sub> , N <sub>2</sub> , ..., N <sub>n</sub> ] [ADJU] [INCL] [dir ALL NONE] [DETA]
<b>ID</b>	The modifier to indicate that the input is for a previously defined subregion.
<b>subrgn</b>	See Mode 1.
<b>D<sub>outer</sub></b>	Nominal width of the borehole which is replaced by a value from the specified function.
<b>func</b>	One of the modifiers listed in Table 4.2.1 that denotes the functional form for borehole diameter. <i>If no function is specified, the value is assumed to be constant.</i>
<b><math>\xi</math></b>	One of the independent variables listed in Table 4.2.2.
<b>N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub></b>	The numerical constants for the selected function. See Section 4.2 for more details. <i>There are no default values for this input.</i>
<b>ADJU</b>	If this modifier is present, and a functional form for the diameter is specified, then the hydraulic conductivity of the fracture is modified as a cubic function.
<b>INCL</b>	See Mode 1.
<b>dir</b>	See Mode 1..
<b>ALL</b>	See Mode 1..
<b>NONE</b>	See Mode 1..
<b>DETA</b>	See Mode 1.

#### EXAMPLES

---

**BOREHOLE** for ID=BOREHOLE\_1 dia = 0.2

**BOREHOLE** ID= BOREHOLE\_2 Y: D=0.2; varies LINEARLY as 0.2 +0.00001 P DETAL

5.10 BOUNDARY

**PURPOSE** To specify boundary conditions for the computational domain.

**MODE 1:** **Constant Boundary Conditions**

**SYNTAX** **BOUN** [ $\Phi$  | **ALL SPEC**] [**option**] [**subopt**] [**dir**] [**subrgn**] {**V<sub>B</sub>**} [**h**]

$\Phi$  **One or more symbols** for the dependent variable(s) for which the boundary conditions are specified. The valid symbols are listed in Table 2.7.1. **In this (constant value) mode of the command, a maximum of 100 symbols may be specified on a single command. If no symbol is specified then the boundary condition is applied to ALL dependent variable for which transport equations are solved.**

**ALL** If **ALL** and the **SPEC** modifiers are present together, then the boundary condition is applied to ALL species variables. That means the velocity and pressure (momentum) and thermal energy equation are excluded from its effect.

**SPEC** See above.

**option** The type of the boundary condition.

option	INTERPRETATION
<b>VALU</b>	Value of the variable at the boundary node is specified as: $\Phi = \Phi_0$ where $\Phi_0$ is the specified numerical value. <b>This is the default option.</b>
<b>FLUX</b>	$q$ , the flux of the variable at the boundary node is specified according to the equation: $q = -\Gamma \frac{\partial \Phi}{\partial N}$ where $\Gamma$ is the diffusion coefficient for the variable and N is the normal direction at the boundary. <b>The flux is positive if it enters the computational domain and negative if it leaves it.</b>
<b>GRAD</b>	The normal gradient of the variable at the boundary, $\frac{\partial \Phi}{\partial N}$ , is specified. The gradient is taken to be <b>positive</b> in the direction of the outward normal at the surface. A positive gradient will result in a positive flux entering the computational domain.
<b>MIXE</b>	The flux of the variable at the boundary is specified according to the equation: $q = -\Gamma \frac{\partial \Phi}{\partial N} = \pm h (\Phi - \Phi_0)$ where h is a transfer coefficient, and $\Phi_0$ is the equilibrium (or ambient) value. The sign is chosen based on the direction vector of the boundary so that the incoming flux is positive.
<b>FLOW</b>	The boundary value depends on the direction of flow. For an incoming flow, the boundary value is fixed (VALU). For an outgoing flow, the normal gradient at the boundary is assumed to be zero.
<b>EXTR</b>	The normal gradient at the boundary is extrapolated from the neighboring values immediately inside the boundary. <b>In general, this is not a well-posed boundary condition for transport equations. However in certain circumstances it may be used to compute the values of secondary variables.</b>

**subopt** The nature of the **MIXED** boundary condition.

subopt	INTERPRETATION
<b>FLUX</b>	Mixed boundary condition is specified in terms of the flux of the variable. <b>This is the default option.</b>
<b>GRAD</b>	Mixed boundary condition is specified in terms of the gradient of the variable (the <b>MIXED</b> condition with $\Gamma = 1$ ).

**dir** The orientation index for the outward normal at the boundary..See Section 3.5 for available choices. **If no value is specified, then the boundary condition is implemented at all**

outermost boundaries of the selected subregion.

- subrgn** The subregion for which the input is specified. If no subregion is specified, the outermost boundary of the entire computational domain is selected. See Section 3.4.
- V<sub>B</sub>** The value of the variable  $\Phi_0$  if **VALUE** or **MIXED** modifier is present, the flux,  $q$ , if the **FLUX** modifier is present, and the outward normal gradient  $\partial\Phi/\partial N$  if the **GRAD** modifier is present. If no modifier is specified, then the **VALUE** modifier is assumed to be present by default. There is no default value.
- h** The transfer coefficient  $h$  for the **MIXED** option. This input is ignored for other options. There is no default value.

## REMARKS

In general, boundary conditions at any boundary segment may be specified only once. Two independent **BOUNDARY** specifications must not refer to the same element for the same boundary orientation index. Unpredictable consequences may arise if the boundary information for any segment is repeated. However, the specification for a boundary segment that was previously identified by a unique identity (**ID=idsub**; see Section 3.4 and **LOCATE** command) may be replaced by a new specification provided the previously specified boundary condition is explicitly disabled by the **BOUNDARY OFF** (see Mode 3 specification) command.

If no boundary conditions are explicitly specified at a particular boundary, then the initial values of the variable at that boundary are used as the boundary conditions. For structured grids, if the number of grid nodes (see **GRID** command) in any direction is less than or equal to 3, then the boundary conditions at the boundaries normal to that direction are assumed to be those of zero flux.

## EXAMPLES

**BOUND**ary for P: at X- boundary, value = 0  
**BOUND**ary GRADient at X+ boundary for all variables is = 0.  
**BOUND**ary for T: orientation index X+ value = 10.  
**BOUND**ary for T at X+: value = 10 at ID=UPPER subregion  
**BOUND**ary for T at Y- boundary: FLUX = 10 at the selected segment  
**BOUND**ary for P at Y- boundary: GRAD = -2.5 at the selected segment  
**BOUND**ary for T at Y+ boundary: MIXEd type: equilibrium value=5 h=0.5  
**BOUND**ary T: Y+; MIXEd FLUX: v=1, h=0.1 for segment with ID=MIDDLE  
**BOUND**ary for C, C2, C3, C4 Y+; Value = 0 for SELEcted subregion

**MODE 2:** Boundary Condition as a Function of another Variable**SYNTAX** **BOUN** [ $\Phi$ ] [**func**[\$]] (**N**<sub>1</sub>, **N**<sub>2</sub>, ..., **N**<sub>n</sub> | **fname**) [**option**] [**subopt**] [**dir**] [**subgrn**] [**h**]

- $\Phi$  **One, and only one**, symbol for the dependent variable for which the boundary conditions are specified. The valid symbols are listed in Table 2.7.1.
- func** One of the modifiers listed in Table 4.2.1 that denotes the functional form of the input. The function specifies the boundary value, flux or gradient of the dependent variable based on the presence of one or more of the **VALUE**, **FLUX**, **GRAD** and **MIXED** modifiers. If no function is specified then the value is assumed to be constant.
- $\xi$  One of the independent variables listed in Table 4.2.2. If no variable is specified, then the independent variable is assumed to be time.
- N**<sub>1</sub>, **N**<sub>2</sub>, ..., **N**<sub>n</sub> The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.
- fname** The name of the file containing numerical values **N**<sub>2</sub> through **N**<sub>n</sub>. This option can be used only if the selected function is a table or one of the series functions. See Section 3.3 for additional information.
- option** See Mode 1 specification.
- subopt** See Mode 1 specification.
- dir** See Mode 1 specification.
- subgrn** See Mode 1 specification.
- h** The transfer coefficient. In the **FLUX** mode, the input units are those of velocity, heat or mass flux divided by the units of the dependent variable,  $\Phi$ . In the **GRADIENT** mode, the input units are those of  $\Phi$  divided by those of distance. The default value is 0. In the presence of the **fname** modifier, the input value must be the 2<sup>nd</sup> numerical value in the command because all functional input is obtained from the file.

**EXAMPLES**

Generic examples for this command are given in Section 4.4. The command keyword (**BOUNDARY**) must replace the keyword used in these examples. The boundary orientation (**dir**) identifier and the transfer coefficient must also be specified, if appropriate.

//// Some additional examples

**BOUND**ary value at X+ boundary LINEar function: T = 30. -0.015 \* Y

**BOUND**ary for T: at X+ boundary FLUX given by USER function #7

**BOUND**ary value for P at Y- COSine SERIes in T: 3 sets: (0., 5, 1), (100, 10, 1), (200, 20, 3) datum = 10.

**BOUND**ary P GRADient: Y- boundary: TABLE of values: 3 sets (TIME, value)  
(0., 0.01), (100., 0.10 ), (200, -0.20)

**BOUND**ary P at X+: TABLE of 32 sets (TIME, value) from file='BVALUES'

//// Examples with transfer coefficient for MIXEd type

**BOUND**ary T: X+: MIXEd: POLYnomial in Y: (10, 0.5, 0, -0.1, 0): h\_F=0.01

// Next 2 lines specify boundary T as a function of T

**BOUND**ary T at Y- boundary MIXEd type: SINE SERIes in T: 3 sets  
(0., 5, 1), (100., 10, 1 ), (200, 20, 3) datum= 0., h=1.E-4

//// Next 2 lines specify mixed type P boundary as a function of time

**BOUND**ary T at X-: MIXEd GRADient TABLE of Z: 3 sets: (0., 5), (100, 15), (200, 7.5), h=0.001

**BOUND**ary MIXEd GRAD P at Y+: SINE SERIes in TIME 24 sets 'BMIXEDV' h=0.01

**MODE 3: Total Pressure Boundary Condition****SYNTAX** **BOUN** {**TOTA**} {**P**} [**func**{**ξ**}] {**N**<sub>1</sub>, **N**<sub>2</sub>, ..., **N**<sub>n</sub> | **fname**} [**N**<sub>n+1</sub>, ..., **N**<sub>n+m</sub>] [**dir**] [**subrgn**]**TOTA** The total pressure at the boundary is specified. The pressure and velocity components at the boundary are computed such that these satisfy the relation:
$$P_{\text{Total}} = P + \frac{1}{2} \rho ( u^2 + v^2 + w^2 )$$

Where **P**<sub>Total</sub> is the total pressure, **P** is the local pressure,  $\rho$  is the density, and **u**, **v**, and **w** are Cartesian velocity components. There is no default value; this modifier must be present on the command.

**P** The symbol to denote that the input is for the pressure variable. There is no default value; this symbol must be present on the command.**func** One of the modifiers listed in Table 4.2.1 that denotes the functional form of the input. The function specifies the value of total pressure. If no function is specified then the value is assumed to be constant.**ξ** One of the independent variables listed in Table 4.2.2. If no variable is specified, then the independent variable is assumed to be time.**N**<sub>1</sub>, ..., **N**<sub>n</sub> The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.**fname** The name of the file containing numerical values **N**<sub>2</sub> through **N**<sub>n</sub>. This option can be used only if the selected function is a table or one of the series functions. See Section 3.3 for additional information.**N**<sub>n+1</sub>, ..., **N**<sub>n+m</sub> If present, these specify the relative values of the Cartesian velocity components at the total pressure boundary. If these are not present, then either a zero gradient or a normal flow at the boundary is assumed. If present, then 2 values are required for 2D and 3 for 3D flow.**dir** See Mode 1 specification.**subrgn** See Mode 1 specification.**APPLICABILITY**This mode of the command is only available with the **ANSWER®** Software Tool.**EXAMPLES**Generic examples for this command are given in Section 4.4. The command keyword (**BOUNDARY**) must replace the keyword used in these examples. The boundary orientation (**dir**) identifier and the transfer coefficient must also be specified, if appropriate.

/// Some additional examples

**BOUND**ary TOTAL P at X+ boundary = 2.E5**BOUND**ary TOTAL P at X- boundary = 1.E6; velocity components (0.5, 0.5, 0.) ! flow at 45 deg to x-axis**BOUND**ary TOTAL P X-: POLYnomial in Y: (10, 0.5, 0., -0.1, 0.)**BOUND**ary TOTAL P X-: POLYnomial in Y: (10, 0.5, 0., -0.1, 0.) (1, 0., 0.) ! flow at 0 deg to x-axis

**MODE 4:** Discharge Coefficient or Gate Boundary Conditions for Velocity Components

**SYNTAX** **BOUN** { $\Phi$ } {**DISC|GATE**} {**dir**} [**subrgn**] {**C<sub>D</sub>**} [**P|PRES=P<sub>B</sub>**] [**AREA=A<sub>B</sub>**] [**IN|OUT**]

**DISC|GATE** The modifier to denote that the discharge coefficient or gate type of boundary condition is being specified. This mode is not available for the **PORFLOW®** software tool. The boundary velocity component is computed from:

$$v = C_D \sqrt{\frac{2(P_B - P)}{\rho}} \frac{A_B}{A}$$

Where  $v$  is the boundary value of the velocity component,  $C_D$  is a discharge or drag coefficient,  $P_B$  is the boundary pressure,  $P$  is the computed pressure in the flow field interior to the boundary,  $\rho$  is the density at the boundary,  $A_B$  if the specified area and  $A$  is the actual area of the boundary face for the element.

- $\Phi$**  **One of the symbols U, V or W** to denote the velocity component to be set by the command.
- dir** See Mode 1 specification.
- subrgn** See Mode 1 specification.
- C<sub>D</sub>** The discharge coefficient for the boundary
- P|PRESS** The modifier to indicate that the pressure at the boundary is being specified explicitly.
- P<sub>B</sub>** **By default it is assumed that the boundary pressure is equal to zero.** If **P** or **PRES** modifier is present, then **P<sub>B</sub>** is taken to be the value of the boundary pressure.
- AREA** The modifier to indicate that the effective flow at the boundary is being specified explicitly.
- A<sub>B</sub>** **By default it is assumed that the area through which the flow occurs is the actual area of the boundary segment; that is A<sub>B</sub>=A.** If the modifier **AREA** is present, then this is taken to be area through which the flow occurs.
- IN** If this modifier is present, then only inflow is allowed at the boundary. Any pressure gradient that indicates an outflow is set to zero.
- OUT** If this modifier is present, then only outflow is allowed at the boundary. Any pressure gradient that indicates an inflow is set to zero.

#### APPLICABILITY

---

This mode of the command is only available with the **ANSWER®** and **TIDAL®** Software Tools.

#### EXAMPLES

---

- BOUN** for **U** at **ID=GATE** surface: **DISCHARGE** coef = 0.5
- BOUN** for **V** at **ID=OPEN** surface: **DISCHARGE** coef = 0.5 , P=100
- BOUN** for **U** at **ID=OPEN** surface: **DISCHARGE** coef = 0.5 , P=100, AREA=0.02

<b>MODE 5:</b>	<b>Gate boundary conditions for Tide Height</b>
<b>SYNTAX</b>	<b>BOUN {ETA} {GATE} {IN OUT} [func[ξ] {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub>} fname] [dir] [subrgn]</b>
<b>ETA</b>	The boundary condition is for water elevation or tide height for <b>TIDAL®</b> Software Tool.
<b>GATE</b>	The boundary is considered to be operated by a gate that allows the water to go in or out based on the nature of the gate (see <b>IN</b> and <b>OUT</b> below).
<b>IN</b>	The gate is assumed to open inwards. When the specified water elevation ( $\eta$ ) exceeds the elevation just inside the boundary, the flow is allowed to go in; otherwise the normal velocity at the boundary ( $V_n$ ) is assumed to be zero. Let S denote the elevation computed from the specification above, B the value at the boundary node, and F that at the field node immediately inside; then the conditions imposed are:  $\eta_S > \eta_F \rightarrow \eta_B = \eta_S; V_{nB} = V_{nF}$ $\eta_S \leq \eta_F \rightarrow \eta_B = \eta_F; V_{nB} = 0$
<b>OUT</b>	The gate is assumed to open outwards. When the elevation just inside the boundary exceeds the specified water elevation ( $\eta$ ), the flow is allowed to go out; otherwise the normal velocity at the boundary ( $V_n$ ) is assumed to be zero. The imposed boundary conditions are:  $\eta_F > \eta_S \rightarrow \eta_B = \eta_S; V_{nB} = V_{nF}$ $\eta_F \leq \eta_S \rightarrow \eta_B = \eta_F; V_{nB} = 0$
<b>func</b>	One of the modifiers listed in Table 4.2.1 that denotes the functional form of the input. The function specifies the value of elevation, $\eta$ , at the boundary. <i>If no function is specified then the value is assumed to be constant.</i>
<b>ξ</b>	One of the independent variables listed in Table 4.2.2. <i>If no variable is specified, then the independent variable is assumed to be time.</i>
<b>N<sub>1</sub>,...,N<sub>n</sub></b>	The numerical constants and coefficients for the selected function. See Section 4.4 for more details. <i>There are no default values for this input.</i>
<b>fname</b>	The name of the file containing numerical values <b>N<sub>2</sub></b> through <b>N<sub>n</sub></b> . <i>This option can be used only if the selected function is a table or one of the series functions.</i> See Section 3.3 for additional information.
<b>dir</b>	See Mode 1 specification.
<b>subrgn</b>	See Mode 1 specification.

**APPLICABILITY**

This mode of the command is only available with the **TIDAL®** Software Tool.

**EXAMPLES**

Generic examples for this command are given in Section 4.4. The command keyword (**BOUNDARY**) must replace the keyword used in these examples and the **ETA**, **GATE** and one of the **IN** or **OUT** modifiers must be specified. The **dir** boundary orientation must also be specified.

//// [An additional example \(2 lines\)](#)

```
BOUNDary ETA at X+ is GATE IN type: SINE function of TIME
      Amplitude=0.1 frequency: 1.405634E-4, phase=0; offset=0, for region ID=SOUTH
```

**MODE 6:** Seepage boundary conditions for PORFLOW®

**SYNTAX** BOUN {SEEP} {N<sub>1</sub>, N<sub>2</sub>} [dir] [subrgn]

**SEEP** This boundary condition is for PORFLOW® Software Tool.

**N<sub>1</sub>** The maximum value of the infiltration at negative pressure head. This is a constant value.  
There are no default values for this input.

**N<sub>2</sub>** The slope of the linear discharge function when the pressure head is positive. This number should be less than zero as a consequence of outflow due to the positive pressure head.  
There are no default values for this input.

**dir** See Mode 1 specification.

**subrgn** See Mode 1 specification.

#### APPLICABILITY

---

This mode of the command is only available with the PORFLOW® Software Tool. This boundary condition combines recharge and drainage into a single boundary condition that is dependent on the pressure head. A negative pressure head (water table below the ground surface) results in infiltration and a positive pressure head results in a discharge, with a continuous transition (based on a cubic polynomial fit) between these two conditions.

#### EXAMPLES

---

**MODE 7:**      **Disable Previously Specified Boundary Conditions**

**SYNTAX**      **BOUN { $\Phi$ } {OFF} {dir} {ID=idsub}**

**$\Phi$**             **One, and only one**, symbol for the dependent variable for which the boundary conditions were previously specified. The valid symbols are listed in Table 2.7.1.

**OFF**            Previously specified boundary conditions for  **$\Phi$**  for the identified subregion and the boundary direction identified by the orientation index N1 are deactivated. A new specification for this subregion may follow.

**dir**            See Mode 1 specification.

**idsub**          Identifier for the boundary segment which must have previously appeared with the same identity on a **LOCATE** or **SELECT** command.

**REMARKS**\_\_\_\_\_

The OFF command can only be used to disable previous commands that were specified for a single dependent variable. The boundary conditions specified with multiple list of variables (Mode 1) cannot be disabled by this command. These can however be superseded by new specifications.

**EXAMPLES**\_\_\_\_\_

**BOUNDARY** for **T** at **Y+** boundary for **ID=UPPER** turned **OFF**

5.11 BRINKMAN-FORCHHEIMER

**PURPOSE** To include Brinkman-Forchheimer terms in the governing Momentum equations. This command is effective only for the **ANSWER®** and **PORFLOW®** Software Tools.

**SYNTAX** **BRIN** [**C<sub>F</sub>**] [**subrgn**]

**C<sub>F</sub>** The Forchheimer coefficient for the non-linear drag terms for flow in a porous medium. The default value is 0.55.. If this command is present then the Brinkman-Forchheimer terms are included in the governing momentum equations.

For **ANSWER®** the governing equation for momentum is modified to account for porosity of the medium as shown below.

$$\frac{\partial}{\partial t}(\theta \rho V_j) + \frac{\partial}{\partial x_i}(\theta \rho V_i V_j) = \frac{\partial}{\partial x_i} \left( \theta \mu \frac{\partial V_j}{\partial x_i} \right) - \theta \frac{\partial p_j}{\partial x_j} + \theta \rho g_j - S_{BF} + S_j$$

$$S_{BF} = \theta^2 (1 + C_F Re_K) \frac{\mu}{K} V_j$$

$$Re_K = \frac{\theta \rho |V|}{\mu} \sqrt{K}$$

Where (the dimensions given in square brackets):

- t is the time [t],
- X<sub>i</sub> is the coordinate in the i<sup>th</sup> direction [L],
- ρ is the mass density of the fluid, [M/L<sup>3</sup>]
- θ is the porosity of medium, [0]
- V<sub>j</sub> is the j<sup>th</sup> component of fluid (particle) velocity [L/t]
- μ is fluid viscosity [M/(L t)]
- p is the fluid thermodynamic pressure [M/(L t<sup>2</sup>)],
- g<sub>j</sub> is gravitational acceleration in the j<sup>th</sup> direction [L/t<sup>2</sup>]
- S<sub>BF</sub>** is Brinkman-Forchheimer drag term [M/(L<sup>2</sup> t<sup>2</sup>)],
- S<sub>j</sub> is the source (or sink) term of the j<sup>th</sup> component of velocity [M/(L<sup>2</sup> t<sup>2</sup>)].
- C<sub>F</sub> is Forchheimer-Ergun drag coefficient [0],
- Re<sub>K</sub> is the Reynolds number based on permeability length scale [0].
- K is the permeability of the medium [L<sup>2</sup>],

For **PORFLOW®** the governing equation for momentum is modified as follows:

$$V_{dj} = - \frac{K_H}{1 + C_F Re_K} \left( \frac{\partial P}{\partial x_j} - \frac{\rho g_j}{\rho^* g} \right)$$

$$K_H = \frac{\rho^* g}{\mu} K; \quad P = \frac{p - p^*}{\rho^* g}$$

Where the superscript '\*\*' denotes a reference value and:

- V<sub>dj</sub> is the Darcy velocity
- K<sub>H</sub> is the hydraulic conductivity of porous medium [L/t],
- P is the hydraulic head [L]

**subrgn** The subregion for which the input is specified. See Section 3.4. If no subregion is specified, the entire computational domain is selected.

**REMARKS**

For **ANSWER®** Software Tool the permeability of the porous medium (**PERMEABILITY command**) must be specified for this command to be effective. For the **PORFLOW®** Software Tool, the hydraulic conductivity specified through the **HYDRAULIC** or **CONDUCTIVITY** command is used to compute these terms.

Diiferent formulations of the Navier-Stokes equations with Brinkman-Forscheimer terms can be found in the literature. The formulation given above is consistent with the physics as well as field-tested Darcy formulation. When the inertial and diffusion terms can be ignored, the governing equations reduce to:

$$\theta^2(1 + C_F Re_K) \frac{\mu}{K} V_j = -\theta \frac{\partial P}{\partial x_j} + \theta \rho g_j$$

For low Reynolds number, the non-linear Brinkman-Forchheimer-Ergun term can be ignored and the equation then simplifies to the standard Darcy formulation:

$$\theta V_j = -\frac{K}{\mu} \left( \frac{\partial P}{\partial x_j} - \rho g_j \right)$$

## EXAMPLES

---

**BRINKMAN** terms to be included. For whole of domain

**BRINKMAN** terms to be included. For whole of domain

**BRINKMAN** terms to be included. For whole of domain

## 5.12 CAPILLARY

<b>PURPOSE</b>	To specify the coefficients for modification of capillary pressure due to changes in temperature or species concentration. This command is effective only for the <b>PORFLOW®</b> Software Tool.
<b>SYNTAX</b>	<b>CAPI</b> {func{ξ}} {N <sub>1</sub> , N <sub>2</sub> , ..., N <sub>n</sub>  fname} [ZERO] [SECO] [subgrn]
<b>func</b>	One of the modifiers listed in Table 4.2.1 that denotes the functional form of the input for capillary pressure. The function specifies the capillary pressure that is used to determine the phase saturation by the <b>MULTIPHASE</b> command. See <b>REMARKS</b> below.
<b>ξ</b>	One of the independent variables listed in Table 4.2.2. If no variable is specified, then the independent variable is assumed to be time.
<b>N<sub>1</sub>,...,N<sub>n</sub></b>	The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.
<b>fname</b>	The name of the file containing numerical values N <sub>2</sub> through N <sub>n</sub> . This option can be used only if the selected function is a table or one of the series functions. See Section 3.3 for additional information.
<b>ZERO</b>	If this modifier is present, then the capillary pressure is set to zero.
<b>SECO</b>	By default the specification is assumed to be for the 1 <sup>st</sup> phase fluid. If this modifier is present, then the input is applied to the capillary pressure of the 2 <sup>nd</sup> phase.
<b>subgrn</b>	The subregion for which the input is specified. See Section 3.4. If no subregion is specified, the entire computational domain is selected.

**APPLICABILITY**

This command is effective only for the **PORFLOW®** Software Tool.

**REMARKS**

The capillary pressure is meaningful only for vadose or multiphase problems (**MULTIPHASE** command), where the phase saturation is a function of the capillary pressure. By default the capillary pressure is computed from the computed values of the pressures for the different phases of the fluid. With P<sup>n</sup> as the pressure of the n<sup>th</sup> phase fluid, the capillary pressure, Ψ<sup>n</sup> is computed from:

$$\Psi^n = P^n - P^{n+1}$$

If the **CAPILLARY** command is present, then the computed value is further modified as per the specification on the command. The n<sup>th</sup> phase saturation, S<sup>n</sup>, is then computed from the specified soil-characteristic by the **MULTIPHASE** command. Though specifics of the S<sup>n</sup>(Ψ<sup>n</sup>) depend on the command, in general, the following conditions prevail for most characteristics.

$$\Psi^1 < 0 \rightarrow S^1 < 1; \Psi^1 \geq 0 \rightarrow S^1 = 1$$

$$\Psi^2 < 0 \rightarrow S^1 + S^2 < 1; \Psi^2 \geq 0 \rightarrow S^1 + S^2 = 1$$

The above guidelines may be violated for special cases especially those when hysteresis is present.

**EXAMPLES**

Generic examples for this command are given in Section 4.4. The command keyword (**CAPI**) must replace the keyword used in these examples.

### 5.13 CLOSE

**PURPOSE** To close output devices.

**SYNTAX** **CLOS** [**filename**] [**SAVE** | **TABL** | **NUNIT**]

**filename** The name of the file to be closed. See Section 3.3 for additional information. **If filename is specified, then the named file, if connected to a unit, will be closed.** If **NUNIT** refers to a different file unit, then that unit, if open, will also be closed.

**SAVE** Currently active archive file (**SAVE**) is closed; the default unit number for this file is 11.

**TABL** Currently active Tabulated archive file (**SAVE TABLE**) is closed; the default unit number for this file is 12.

**NUNIT** The file or I/O device unit number for the file for which the operation is performed. **The unit number is ignored if one of the SAVE or TABL modifier is present.**

#### REMARKS

---

In general FORTRAN language does not differentiate between lower and upper case characters. However that is not true of many operating systems (O/S). It is in fact the O/S that performs all file handling. O/S peculiarities have been reported where, for example, there was no case distinction for read/write operations but the O/S distinguished between upper and lower case for file closing. **It is therefore strongly recommended that the user should be consistent** in using the same case characters if a previous file is referred to during file operations.

#### EXAMPLES

---

**CLOSE** SAVE file

**CLOSE** archive TABLE output file now

**CLOSE** file by name 'MYOLDFILE'

5.14 COMPRESSIBLE

**PURPOSE** To select compressible flow option. This command is effective only for the ANSWER® Software Tool.

**MODE 1:** Mild Compressibility Option

**SYNTAX** COMP {MILD} [ATMO] [N1]

**MILD** By default, the COMPRESSIBLE command invokes the fully compressible mode where density is assumed to be a function of the local pressure. If, this modifier is present then the mildly compressible mode of the equations is invoked. In this mode  $\partial\rho/\partial p$  is assumed to be given from the gas law and, the variation of density due to pressure is taken account of only in the accumulation (time-dependent) term of the continuity equation. The fluid density is computed with a reference pressure rather than the local pressure. This form of the equations is primarily useful for acoustic purposes.

$$\frac{\partial\rho}{\partial t} = \frac{\partial\rho}{\partial p} \frac{\partial p}{\partial t} = \alpha \rho \frac{\partial p}{\partial t}$$

In this equation the constant  $\alpha$  has the units of 1/p.

**ATMO** The constant  $\alpha$  is assumed to be equal to 1./101325; the denominator being the value of the standard atmosphere in metric units.

**N1** If a value greater than zero is specified then constant  $\alpha$  is assumed to be equal to the specified value. If this value is omitted or is less than or equal to zero, and the **ATMO** is not specified, then  $\alpha$  is assumed to be a local variable and is computed as:

$$\alpha = \frac{1}{\gamma(p+p^*)}$$

Where  $\gamma$  is the specific heat ratio,  $p$  is the local density,  $p$  is the local pressure and  $p^*$  is the reference pressure.

**REMARKS**

For the compressible mode of solution, the enthalpy equation is modified to solve for stagnation enthalpy,  $h_s$ , rather than enthalpy,  $h_T$ . That is, the mean and turbulence kinetic energy of fluid is added to the enthalpy as defined by the below equation

$$h_s = h_T + \frac{1}{2}(U^2 + V^2 + W^2) + k$$

$$h_T = \sum_j m_j h_j ; \sum_j m_j = 1 ,$$

$$h_j = h_j^o + \int_T C_{p_j} dT ,$$

$$k = \frac{1}{2}(\overline{uu} + \overline{vv} + \overline{ww})$$

where

- $h_j$  is the enthalpy, and  $m_j$ , the mass fraction, of the  $j^{\text{th}}$  chemical species,
- $h_j^o$  is the heat of formation of the  $j^{\text{th}}$  species,
- $C_{p_j}$  is the specific heat of the  $j^{\text{th}}$  species at constant pressure,
- $T$  is the fluid temperature,
- $U, V, W$  are the mean, and  $u, v, w$  the fluctuating, velocity components, and
- $k$  is the turbulence energy of fluid.

**EXAMPLES**

---

**COMP**ressible flow

**COMP**ressible flow with  $C=1.0$

**COMP**ressible flow with MILD compressibility

**COMP**ressible ISENthalpic flow

**COMP**ressible flow with INViscid and ISOPressure assumptions

**COMP**ressible flow with INViscid and ISOPressure assumptions; use FIXED pressure option

**MODE 2:** Full Compressibility Option

**SYNTAX** COMP [DENS|SIMP|SYMM|DIRE|COUP] [ISEN|ISOT] [ISOP] [INVI] [FIXE] [RHOU] [N<sub>1</sub>]

**DENS** The density correction based DEFCON method is employed to solve the fully compressible version of the continuity equation. This is the default option.

**SIMP** The pressure correction based SIMPLE method is employed.

**SYMM** It is assumed that the changes in convective fluxes are directly proportional to changes in pressure gradient. This may lead to more robust numerical method for transonic flows. This option is not recommended for high Mach number flows.

**DIRE** By default after a new estimate of the pressure is computed at a given step, the velocity field, and therefore, convective mass flux, is updated to account for changes in pressure before going on to the next step. If this modifier is present, then no corrections are performed. This essentially leads to a direct sequential solution of the momentum and continuity equations.

**COUP** This is a reserved modifier. It implements a coupled solution method that is not yet available for general use.

**ISEN** By default, the temperature is computed from the stagnation enthalpy equation. If, this modifier is present then the stagnation enthalpy,  $h_s$ , is assumed to be conserved and the temperature is computed from:

$$T = (h_s - 0.5V^2) / C_p$$

where  $V$  is the flow speed and  $C_p$  is the specific heat at constant pressure. In this case, the stagnation enthalpy of the fluid must be supplied as input by the user.

**ISOT** The flow is assumed to be isothermal. Temperature is not computed; it is maintained at the initially specified value.

**ISOP** By default, the contribution of the  $\partial p/\partial t$  term is included in the computation of the enthalpy equation. If this modifier is present then this term is ignored.

**INVI** By default, the viscous heating terms are included in the computation of the enthalpy equation. If this modifier is present then the viscous heating effect in the stagnation enthalpy equation is ignored.

**FIXE** By default, for mildly compressible flow, the pressure is assumed to be in relative mode. The actual gas (datum) pressure is specified by the **GAS P** command. If this modifier is present, then the all input and output of pressure is in absolute mode.

**RHOU** By default, the convective flux at the interfaces of elements is computed from a product of averages of density and velocity. If this modifier is present, then the interface flux is computed as the average of the product of density and velocity.

**N<sub>1</sub>** The constant  $C$  ( $>0$ ) of the pressure perturbation equation as given below. The default value is 1.4.

$$P' = \rho' CRT$$

Where  $C$  is equal to unity for an isothermal process, and is equal to,  $\gamma$ , the ratio of the specific heat at constant pressure to that at constant temperature, for an isentropic process.

## REMARKS

---

For the compressible mode of solution, the enthalpy equation is modified to solve for stagnation enthalpy rather than enthalpy. See the comments under Mode 1.

**EXAMPLES**

---

**COMP**pressible flow

**COMP**pressible flow with C=1.0

**COMP**pressible flow with MILD compressibility

**COMP**pressible ISEnthalpic flow

**COMP**pressible flow with INViscid and ISOPressure assumptions

**COMP**pressible flow with INViscid and ISOPressure assumptions; use FIXED pressure option

5.15 CONDUCTIVITY

**PURPOSE** To specify conductivity or diffusivity and the treatment of interface diffusion.

**MODE 1:** Constant Hydraulic Conductivity in Principal Directions

**SYNTAX** COND {P} { $\Gamma_{xx}$ ,  $\Gamma_{yy}$ , [ $\Gamma_{zz}$ ]} [SATU|TOTA] [option] [subrgn]

**P** Hydraulic conductivity is specified. **This symbol must be specified.**

**$\Gamma_{xx}$ ,  $\Gamma_{yy}$ ,  $\Gamma_{zz}$**  The numerical values (>0) for the conductivity Two values must be specified for 2D geometry and three for 3D geometry. The default units are [L/t] such as [m/s].

**SATU** If this modifier is present, then it is assumed that the specified values are for the saturated hydraulic conductivity of the porous matrix. The saturated value is multiplied with the relative conductivity (e.g. of unsaturated media see MULTIPHASE and COND RELATIVE commands) to compute the effective conductivity. **This is the default option.**

**TOTA** If this modifier is present, then the input is assumed to be the absolute or the effective hydraulic conductivity. No multiplication by relative conductivity takes place.

**option** Option selected for the operation to be performed.

option	INTERPRETATION
<b>REPL</b>	The computed value replaces the existing value. <b>This is the default option.</b>
<b>ADD</b>	The computed value is added to the existing value.
<b>SUBT</b>	The computed value is subtracted from the existing value.
<b>MULT</b>	The computed value multiplies the existing value.
<b>DIVI</b>	The computed value divides the existing value. A value of 1.E-30 is added to the divisor if it is zero.
<b>OFF</b>	A previously specified command for the same variable and subregion is deactivated.

**subrgn** The subregion for which the input is specified. See Section 3.4. **If no subregion is specified, the entire computational domain is selected.**

**APPLICABILITY** \_\_\_\_\_

This input is applicable only for the **PORFLOW®** Software Tool..

**EXAMPLES** \_\_\_\_\_

**CONDUCTIVITY** for **P = 0.02, 0.01, 0.01** meters per year

**CONDUCTIVITY** for **P = 0.02, 0.01, 0.01** meters per year **ADD** to existing value in **ID=BLK1**

<b>MODE 2:</b>	<b>Hydraulic Conductivity in a Specified Direction</b>
<b>SYNTAX</b>	<b>COND {P} [xyz] {func [ξ] {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub>   fname} [ROBU] [SATU TOTA] [option] [subrgn]</b>
<b>P</b>	The hydraulic conductivity is specified.
<b>xyz</b>	One of the character strings: <b>XX</b> , <b>YY</b> , <b>ZZ</b> , <b>XY</b> , <b>YX</b> , <b>XZ</b> , <b>ZX</b> , <b>YZ</b> and <b>ZY</b> . It denotes the component of the conductivity tensor to which the input is applied. The first 3 denote the diagonal components of the tensor while the others denote the off-diagonal components. Since the tensor is symmetric, <b>XY</b> and <b>YX</b> , <b>XZ</b> and <b>ZX</b> , and <b>YZ</b> and <b>ZY</b> are equivalent. If no direction is specified, then the input is applied to all components of the tensor.  If any off-diagonal component is specified, then the tensor mode of computation is automatically initiated (see <b>COND TENSOR</b> command mode).
<b>func</b>	One of the modifiers listed in Table 4.2.1 that denotes the functional form of the value of the appropriate component(s) of hydraulic conductivity. If no function is specified then the value is assumed to be constant.
<b>ξ</b>	One of the independent variables listed in Table 4.2.2. If a function is explicitly specified but no variable is specified, then the independent variable is assumed to be time.
<b>N<sub>1</sub>,...,N<sub>n</sub></b>	The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.
<b>fname</b>	The name of the file containing numerical values <b>N<sub>2</sub></b> through <b>N<sub>n</sub></b> . This option can be used only if the selected function is a table or one of the series functions. See Section 3.3 for additional information.
<b>ROBU</b>	This modifier is effective only <b>xyz</b> is such that tensor computation is initiated. In this case the tensor computations are carried out in the robust mode described in the <b>COND TENSOR</b> command.
<b>SATU</b>	See Mode 1 of the command.
<b>TOTA</b>	See Mode 1 of the command.
<b>option</b>	See Mode 1 of the command.
<b>subrgn</b>	See Mode 1 of the command.

**APPLICABILITY**

This input is applicable only for the **PORFLOW®** Software Tool..

**EXAMPLES**

Generic examples for this command are given in Section 4.4. The command keyword (**COND**) must replace the keyword used in these examples and the modifier **P** must appear on the command

<b>MODE 3:</b>	<b>Hydraulic Conductivity in a Specified Direction</b>
<b>SYNTAX</b>	<b>COND {P} {RELA} {func[ξ]} {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub> fname} [option] } [GRAD] [subrgn]</b>
<b>P</b>	The hydraulic conductivity is specified.
<b>RELA</b>	The input is assumed to be for the relative hydraulic conductivity. The actual hydraulic conductivity is obtained by multiplying the saturated conductivity with the relativity conductivity.
<b>func</b>	One of the modifiers listed in Table 4.2.1 that denotes the functional form of the value of the appropriate component(s) of hydraulic conductivity. <b>If no function is specified then the value is assumed to be constant.</b>
<b>ξ</b>	One of the independent variables listed in Table 4.2.2. <b>If a function is explicitly specified but no variable is specified, then the independent variable is assumed to be time.</b>
<b>N<sub>1</sub>,...,N<sub>n</sub></b>	The numerical constants and coefficients for the selected function. See Section 4.4 for more details. <b>There are no default values for this input.</b>
<b>fname</b>	The name of the file containing numerical values <b>N<sub>2</sub></b> through <b>N<sub>n</sub></b> . <b>This option can be used only if the selected function is a table or one of the series functions.</b> See Section 3.3 for additional information.
<b>option</b>	See Mode 1 of the command.
<b>GRAD</b>	<b>By default the relative hydraulic conductivity at boundaries is computed from the specified function.</b> If this modifier is present, then the value of relative conductivity <b>at all external boundaries</b> is set equal to the field value at the nearest neighbor.
<b>subrgn</b>	See Mode 1 of the command.

#### APPLICABILITY

---

This input is applicable only for the **PORFLOW™** Software Tool..

#### EXAMPLES

---

Generic examples for this command are given in Section 4.4. The command keyword (**COND**) must replace the keyword used in these examples and the modifiers **P RELA** must appear on the command

**MODE 4: Thermal Conductivity of Primary Fluid Phase**

**SYNTAX** **COND** {HT|T} [xyz] {func[ξ]} {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub>|fname} [ROBU] [MOLE|TOTA] [ALWA]  
[option] [subrgn] [FIEL|dir]

**HT** The symbol that denotes that the value specified is the thermal conductivity of fluid. The thermal conductivity must be specified in mass units [e.g. J/(m-s °K)].

**T** Same as above.

**xyz** One of the character strings: **XX**, **YY**, **ZZ**, **XY**, **YX**, **XZ**, **ZX**, **YZ** and **ZY**. It denotes the component of the conductivity tensor to which the input is applied. The first 3 denote the diagonal components of the tensor while the others denote the off-diagonal components. Since the tensor is symmetric, **XY** and **YX**, **XZ** and **ZX**, and **YZ** and **ZY** are equivalent. If no direction is specified, then the input is applied to all components of the tensor.

Currently, the off-diagonal components can only be specified with the **PORFLOW®** Software Tool. If any off-diagonal component is specified, then the tensor mode of computation is automatically initiated (see **COND TENSOR** command mode).

**func** One of the modifiers listed in Table 4.2.1 that denotes the functional form for the value of the thermal conductivity.

**ξ** One of the independent variables listed in Table 4.2.2. If a function is explicitly specified but no variable is specified, then the independent variable is assumed to be time.

**N<sub>1</sub>,...,N<sub>n</sub>** The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.

**fname** The name of the file containing numerical values **N<sub>2</sub>** through **N<sub>n</sub>**. This option can be used only if the selected function is a table or one of the series functions. See Section 3.3 for additional information.

**ROBU** This modifier is effective only **xyz** is such that tensor computation is initiated. In this case the tensor computations are carried out in the robust mode described in the **COND TENSOR** command.

**MOLE** If this modifier is present, then it is assumed that the specified value is for the molecular component of the thermal conductivity. The dispersion (see **DISPERSION** command) is added to the computed value.

**TOTA** If this modifier is present, then it is assumed that the specified value is for the **total thermal conductivity of the system**. No dispersion is added and no account is taken of the fact that the total system may consist of multiple fluid phases or a solid-fluid matrix. This is the default option.

**ALWA** If this modifier is present, then the thermal conductivity is computed at every step of calculation. By default, the value is computed once (at the time the command is encountered) and stored in the thermal conductivity array. This modifier is only required if the computed value is a function of a variable that will change during computations.

**option** See Mode 1 of the command.

**subrgn** See Mode 1 of the command.

**FIEL** See Mode 1 of the command.

**dir** See Mode 1 of the command.

**REMARKS**

If the computed system has multiple fluid phases or is a fluid-solid matrix, then the effective value of thermal conductivity is computed as the weighted sum of the fluid and solid components. The weighting function is the volumetric saturation of each phase. The dispersion, if any, is then added to this effective value to arrive at the total thermal conductivity of the system.

**EXAMPLES**

Generic examples for this command are given in Section 4.4. The command keyword (**COND**) must replace the keyword used in these examples and the modifier **T** or **HT** must appear on the command

**MODE 5:** Thermal Conductivity of the Second Fluid Phase

**SYNTAX** COND {HT|T} {SECO} {Γ}

**HT** The symbol that denotes that the value specified is the thermal conductivity of fluid. The thermal conductivity must be specified in mass units [e.g. J/(m-s °K)].

**T** Same as above.

**SECO** The thermal conductivity of the 2<sup>nd</sup> fluid phase is specified.

**Γ** The constant and uniform value of the thermal conductivity for the 2<sup>nd</sup> phase fluid. The default value is zero.

#### REMARKS

---

Current versions of the [ACRI Software Tools](#) only allow a constant and uniform value for the thermal conductivity of the 2<sup>nd</sup> phase fluid. If a 3<sup>rd</sup> fluid phase is present then the thermal conductivity for the 3<sup>rd</sup> phase is assumed to be zero; user can not specify a value for the 3<sup>rd</sup> phase.

#### EXAMPLES

---

**CONDUCTIVITY** for **T SECOND** phase value =**1.E-5**

<b>MODE 6:</b>	<b>Thermal Conductivity of the Solid Component of a Fluid-Solid Matrix</b>
<b>SYNTAX</b>	<b>COND {HT T} {SOLI} {func[ξ]} {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub> fname} [option] [subrgn] [FIEL dir]</b>
<b>HT</b>	The symbol that denotes that the value specified is the thermal conductivity of fluid. The thermal conductivity must be specified in mass units [e.g. J/(m-s °K)].
<b>T</b>	Same as above.
<b>SOLI</b>	The thermal conductivity of the solid component of a fluid-solid matrix is specified.
<b>func</b>	One of the modifiers listed in Table 4.2.1 that denotes the functional form for the value of the thermal conductivity.
<b>ξ</b>	One of the independent variables listed in Table 4.2.2. If a function is explicitly specified but no variable is specified, then the independent variable is assumed to be time.
<b>N<sub>1</sub>,...,N<sub>n</sub></b>	The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.
<b>fname</b>	The name of the file containing numerical values N <sub>2</sub> through N <sub>n</sub> . This option can be used only if the selected function is a table or one of the series functions. See Section 3.3 for additional information.
<b>option</b>	See Mode 1 of the command.
<b>subrgn</b>	See Mode 1 of the command.
<b>FIEL</b>	See Mode 1 of the command.
<b>dir</b>	See Mode 1 of the command.

**REMARKS**

If the computed system has multiple fluid phases or is a fluid-solid matrix, then the effective value of thermal conductivity is computed as the weighted sum of the fluid and solid components. The weighting function is the volumetric saturation of each phase. The dispersion, if any, is then added to this effective value to arrive at the total thermal conductivity of the system.

**EXAMPLES**

Generic examples for this command are given in Section 4.4. The command keyword (**COND**) must replace the keyword used in these examples and the modifier **T** or **HT** and **SOLI** must appear on the command

<b>MODE 7:</b>	<b>Conductivity or Diffusivity of a Mass Species</b>
<b>SYNTAX</b>	<b>COND {<math>\Phi</math>} [xyz] {func[<math>\xi</math>]} {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub> fname} [ROBU] [MOLE TOTA] [KINE MASS] [option] [subrgn] [FIEL dir]</b>
<b><math>\Phi</math></b>	A symbol that denotes the dependent variable for which the conductivity or diffusivity is specified. Valid symbols are listed in Table 2.7.1. <i>There is no default value; a symbol must be specified.</i>
<b>xyz</b>	One of the character strings: <b>XX, YY, ZZ, XY, YX, XZ, ZX, YZ</b> and <b>ZY</b> . It denotes the component of the conductivity or diffusivity tensor to which the input is applied. <i>The first 3 denote the diagonal components of the tensor while the others denote the off-diagonal components. Since the tensor is symmetric, XY and YX, XZ and ZX, and YZ and ZY are equivalent. The off-diagonal components are only allowed for the PORFLOW® software. If no direction is specified, then the input is applied to all diagonal components of the tensor.</i>  <i>Currently, the off-diagonal components can only be specified with the PORFLOW® Software Tool. If any off-diagonal component is specified, then the tensor mode of computation is automatically initiated (see COND TENSOR command mode).</i>
<b>func</b>	One of the modifiers listed in Table 4.2.1 that denotes the functional form of the value of the appropriate conduction or diffusion coefficient for the corresponding $\Phi$ variable. <i>If no function is specified then the value is assumed to be constant.</i>
<b><math>\xi</math></b>	One of the independent variables listed in Table 4.2.2. <i>If a function is explicitly specified but no variable is specified, then the independent variable is assumed to be time.</i>
<b>N<sub>1</sub>,...,N<sub>n</sub></b>	The numerical constants and coefficients for the selected function. See Section 4.4 for more details. <i>There are no default values for this input.</i>
<b>fname</b>	The name of the file containing numerical values <b>N<sub>2</sub></b> through <b>N<sub>n</sub></b> . <i>This option can be used only if the selected function is a table or one of the series functions.</i> See Section 3.3 for additional information.
<b>ROBU</b>	This modifier is effective only <b>xyz</b> is such that tensor computation is initiated. In this case the tensor computations are carried out in the robust mode described in the <b>COND TENSOR</b> command.
<b>MOLE</b>	If this modifier is present, then it is assumed that the specified value is for the molecular component of the diffusivity. The dispersion (see <b>DISPERSION</b> command) is added to the computed value.
<b>TOTA</b>	If this modifier is present, then it is assumed that the specified value is for the total diffusivity. No dispersion is added to the computed value. <i>This is the default option.</i>
<b>KINE</b>	By default the units of diffusivity are mass units [e.g. kg/m-s] for <b>ANSWER®</b> and <b>TIDAL®</b> Software Tools. If <b>KINE</b> modifier is present, the specification is assumed to be in kinetic units (e.g. m <sup>2</sup> /s). The kinetic units are converted to mass units by multiplication with reference density (see <b>REFERENCE</b> command).
<b>MASS</b>	By default <b>PORFLOW®</b> input for mass species is assumed to be in kinetic units (e.g. m <sup>2</sup> /s). If this modifier is present, then the input is assumed to be in mass units [e.g. kg/m-s]. The conversion to kinetic units is done by division with reference fluid density (see <b>REFERENCE</b> command).
<b>option</b>	See Mode 1 of the command.
<b>subrgn</b>	The subregion for which the input is specified. See Section 3.4. <i>If no subregion is specified, the entire computational domain is selected.</i>

## APPLICABILITY

For **PORFLOW®** Software Tool the vector (with diagonal components) mode of the command is operational for all equations. The tensor (with off-diagonal components) mode is applicable only for the thermal and species transport equations.

For **ANSWER®** Software Tool this command can be employed for all equations. However, if the vector or

tensor mode (**dir** modifier) is invoked, then the skew diffusion terms (see Mode 6) are ignored and the wall diffusive flux is set to zero for all except the momentum equations.

For **TIDAL®** Software Tool this command is implemented only for the species transport equations.

#### REMARKS

---

The conductivity for a species is generally a tensor. This command allows the components of the tensor to be specified in many different forms. For flow in porous media, the conductivity is often modified by dispersivity and tortuosity. See the **DISPERSIVITY** and **TORTUOSITY** commands for further information.

#### EXAMPLES

---

**CONDUCTIVITY** for C =1.E-6

! Diffusivity for the species transport equation

See also generic examples for functional specification in Section 4.4. The command keyword (**COND**) must replace the keyword used in these examples. The modifier XX, YY or ZZ may be additionally specified on the command.

<b>MODE 8:</b>	<b>Specification of Conductivity or Diffusivity on a Node by Node Basis</b>
<b>SYNTAX</b>	<b>COND {<math>\Phi</math>} {NODE STRU UNST fname} [V<sub>1</sub>,...V<sub>m</sub>] [xyz] [KINE MASS] [option] [subrgn] [FIEL]</b>
<b><math>\Phi</math></b>	A symbol that denotes the dependent variable for which the conductivity or diffusivity is specified. Valid symbols are listed in Table 2.7.1. There is no default value; a symbol must be specified.
<b>NODE</b>	The input is specified in a node-by-node manner. By default it is assumed that the input values are given in the <b>STRUCTURED</b> mode if the <b>subrgn</b> was defined with a <b>LOCATE</b> command in terms of grid indices or in the <b>UNSTRUCTURED</b> mode otherwise.
<b>STRU</b>	The input is read in the manner of nested implied FORTRAN DO loops: <pre>Read (NUNIT, *) (((VAR (I, J, K), I = ILO, IHI), J = JLO, JHI), K = KLO, KHI)</pre> Where <b>NUNIT</b> is an internally assigned unit number, <b>VAR</b> is the variable denoted by <b><math>\Phi</math></b> , and <b>ILO, IHI, JLO, JHI, KLO, KHI</b> define, respectively, the starting and ending grid index values for the <b>subrgn</b> for a structured grid. For 2D grid, <b>KLO</b> and <b>KHI</b> are set to unity.
<b>UNST</b>	The input is read in the manner of a simple implied FORTRAN DO loop: <pre>Read (NUNIT, *) (VAR (M), M = MLO, MHI)</pre> Where <b>NUNIT</b> is an internally assigned unit number, <b>VAR</b> is the variable denoted by <b><math>\Phi</math></b> , and <b>MLO and MHI</b> are the starting and ending element numbers for the <b>subrgn</b> . The order of the elements is the same as the one specified (or implied) by the corresponding <b>LOCATE</b> command which can be examined by including a file name on the <b>LOCATE</b> command.
<b>fname</b>	The name of the file for numerical values unless the values are directly specified by <b>V<sub>1</sub>...V<sub>m</sub></b> below. See Section 3.3 for additional information.
<b>V<sub>1</sub>...V<sub>m</sub></b>	The values of the variable at each node selected by the <b>subrgn</b> and <b>dir</b> modifiers. These values must be present unless <b>fname</b> is present. In this case, one of the modifiers, <b>NODE</b> , <b>STRU</b> , or <b>UNST</b> must also be specified.
<b>xyz</b>	One of the character strings: <b>XX, YY, ZZ, XY, YX, XZ, ZX, YZ</b> and <b>ZY</b> . It denotes the component of the conductivity or diffusivity tensor to which the input is applied. The first 3 denote the diagonal components of the tensor while the others denote the off-diagonal components. Since the tensor is symmetric, <b>XY</b> and <b>YX</b> , <b>XZ</b> and <b>ZX</b> , and <b>YZ</b> and <b>ZY</b> are equivalent. The off-diagonal components are only allowed for the <b>PORFLOW®</b> software. If no direction is specified, then the input is applied to all diagonal components of the tensor.  Currently, the off-diagonal components can only be specified with the <b>PORFLOW®</b> Software Tool. If any off-diagonal component is specified, then the tensor mode of computation is automatically initiated (see <b>COND TENSOR</b> command mode).
<b>KINE</b>	See Mode 1 of command..
<b>MASS</b>	See Mode 1 of command..
<b>option</b>	See Mode 1 of command..
<b>subrgn</b>	The subregion for which the input is specified. See Section 3.4. If no subregion is specified, the entire computational domain is selected.

**APPLICABILITY**

See Mode 2 of the Command.

**EXAMPLES**

**CONDuctivity** by NODE for P variable for ID=SUBRGN1 ! 3 nodes in this region – **PORFLOW®**  
1.5, 11.5, 10.

**CONDuctivity** for P NODE in STRUctured mode from File “KHYD.FIL”

**CONDuctivity** for P STRUctured mode from File “KHYD.FIL” for Field NODES only

**CONDuctivity** by NODE ID=MIDDLE T from file ‘SPATIAL’ in UNSTructured mode from ‘KT.FIL’

<b>MODE 9:</b>	<b>Specification of Contact or Film Transfer Coefficient</b>
<b>SYNTAX</b>	<b>COND {CONT FILM} {Φ} {func[ξ]} {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub> fname} [REPL ADD TOTA OFF] {subrgn} [dir]</b>
<b>CONT</b>	The contact or film transfer coefficient at a surface is specified. This transfer coefficient can replace or augment the conduction or diffusion that occurs at a surface.
<b>FILM</b>	Same as <b>CONTACT</b> Modifier.
<b>Φ</b>	A symbol that denotes the dependent variable for which the input is specified. Valid symbols are listed in Table 2.7.1. <b>There is no default value; a symbol must be specified.</b>
<b>func</b>	One of the modifiers listed in Table 4.2.1 that denotes the functional form of the input. The function specifies the value of the appropriate transfer coefficient for the specified <b>Φ</b> variable. <b>If no function is specified then the value is assumed to be constant.</b>
<b>ξ</b>	One of the independent variables listed in Table 4.2.2. <b>If no variable is specified, then the independent variable is assumed to be time.</b>
<b>N<sub>1</sub>,...,N<sub>n</sub></b>	The numerical constants and coefficients for the selected function. See Section 4.4 for more details. <b>There are no default values for this input.</b>
<b>fname</b>	The name of the file containing numerical values <b>N<sub>2</sub></b> through <b>N<sub>n</sub></b> . <b>This option can be used only if the selected function is a table or one of the series functions.</b> See Section 3.3 for additional information.
<b>REPL</b>	<b>This is the default mode.</b> Any internal computations for the conductive (diffusive) flux between two nodes across a surface are replaced by: $q_{\Phi} = \frac{A_f (\Phi_1 - \Phi_2)}{\frac{ds_1}{\Gamma_1} + \frac{ds_2}{\Gamma_2} + \frac{1}{h}}$
	where, $q_{\Phi}$ is flux from node "1" to node "2", $A_f$ is the interface area, $ds_1$ and $ds_2$ are distances, $\Gamma$ is the diffusion coefficient and, $h$ is the specified transfer coefficient.
<b>ADD</b>	The transfer flux is added to the internally computed flux (which may have been modified by effects such as wall functions for turbulent flow). In this case: $q_{\Phi} = q_{\Phi_{\text{internal}}} + h A_f (\Phi_1 - \Phi_2)$
<b>TOTA</b>	The diffusive flux at the surface is computed as: $q_{\Phi} = h A_f (\Phi_1 - \Phi_2)$
<b>OFF</b>	A previously specified command for the same variable and subregion is deactivated.
<b>subrgn</b>	The subregion for which the input is specified. See Section 3.4. <b>If no subregion is specified, the entire computational domain is selected.</b>
<b>dir</b>	The orientation index for the element boundary associated with the transfer coefficient. See Section 3.5 for available choices. <b>If no dir is specified, then all surfaces of the selected subrgn are selected.</b>

**APPLICABILITY**

This command mode is not available for the **TIDAL®** Software Tool

**EXAMPLES**

**COND**uction for T with contact heat transfer coefficient of 0.01 at Y+ of ID=WALL  
**COND**uction for T with contact heat transfer coefficient of 0.01 at Y+ of ID=WALL **TOTA**  
 Generic examples for this command are given in Section 4.4. The command keyword (**COND**) must replace the keyword used in these examples. The modifier **FILM** or **CONTACT** must appear in the command.

**MODE 10:**     **Modification of Diffusion Coefficient by Richardson Number**

**SYNTAX**     **COND {RICH } {dir } [UNLI | ZERO] [α, β, α<sub>φ</sub>, β<sub>φ</sub>]**

**RICH**         The viscosity and diffusion coefficient in the specified direction are modified due to the effects of density gradient for the Richardson number effect. The viscosity,  $\mu_j$ , and the diffusion coefficient,  $\Gamma_j$ , in the selected direction are computed from:

$$\mu_j = \frac{\mu}{[1 + \beta Ri]^\alpha} ; \quad \Gamma_j = \frac{\Gamma}{[1 + \beta_\phi Ri]^{\alpha_\phi}}$$

Where subscript “j” denotes the direction specified by the **dir** modifier,  $\mu$  and  $\Gamma$  are, respectively, the scalar values of the viscosity and diffusion coefficients (specified by user or computed from built-in formulae), and  $\alpha$ ,  $\beta$ ,  $\alpha_\phi$  and  $\beta_\phi$  empirical constants. The Richardson number, Ri, is defined as:

$$Ri = g_i \frac{\frac{1}{\rho} \frac{\partial \rho}{\partial x_i}}{\sum_{j, j \neq i} \left( \frac{\partial u_j}{\partial x_i} \right)^2}$$

where  $g_i$  is the gravitational acceleration in the  $i^{\text{th}}$  direction (specified by the **dir** modifier),  $\rho$  is the fluid density and  $u_j$  represents the velocity components in directions other than the  $i^{\text{th}}$  direction.

A final consequence of Richardson number modification is that the production term in the turbulence kinetic energy equation is modified to include the additional term for production/destruction of turbulence energy due to buoyancy:

$$- \frac{\mu_e}{\rho^2} \frac{\partial \rho}{\partial x_i} \frac{\partial P}{\partial x_i}$$

If this command with a **RICH** modifier is not specified, then this additional term is excluded from the turbulence kinetic energy equation.

**dir**           **One** of the **X, Y, Z, R** or **THETA** modifiers that, respectively, denotes directional viscosity and diffusivity that are modified. **If no direction is specified, then y-direction in the 2D and Z direction in the 3D geometry is automatically selected.**

**UNLI**         **By default the modification of the production term in the turbulence kinetic equation is performed with the constraint that the net production term is not less than zero; that is the total production can never become negative. However if the UNLIMITED modifier is present, then the production term is allowed to become negative. If this option is activated, then it is possible that turbulence kinetic energy, k, may become negative. This is physically unrealistic; it is therefore strongly recommended that the LIMIT command should be specified to place a reasonable lower limit on k.**

**ZERO**         The additional term in the turbulence production is identically set to zero.

**α, β, α<sub>φ</sub>, β<sub>φ</sub>**     The empirical constants for the Richardson number formula. **The default values are 0.5, 10, 1.5 and 3.33, respectively.**

#### **APPLICABILITY** \_\_\_\_\_

This command mode is available only for the **ANSWER®** Software Tool

**EXAMPLES**

---

**COND**uction modified by RICHardson Number

**COND**uction modified by RICHardson Number with constants: 0.67, 6.67, 1., 5

**MODE 11** Treatment of Skew Diffusion Terms

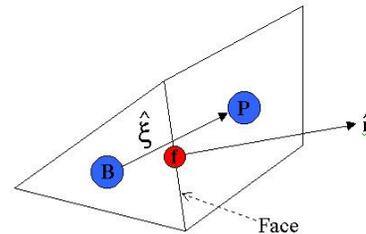
**SYNTAX** COND {SKEW|SECO} [ROBU] [ $\Phi$ ]

**SKEW** Skew conduction or diffusion terms at the element interfaces, that arise if the grid arrangement is non-orthogonal, are included in computing interface diffusive fluxes. Typically these are 2<sup>nd</sup> order terms and have very little influence in convection-dominated flow but the computing time is generally increased by 10 to 20 percent. In many practical situations, these terms can be ignored. By default these terms are not included; hence this modifier must be specified to account for skew diffusion.

**SECO** Same as **SKEW** modifier.

**ROBU** The total diffusive flux for the variable  $\Phi$  at the face "f" between two elements shown in the picture below is given by:

$$Q_D = -\Gamma_{ij}^f \frac{\partial \Phi}{\partial x_j} n_i A_f ,$$



where summation convention is implied, and  $\Gamma_{ij}^f$  is the diffusion coefficient for the variable  $\Phi$  at face f,  $A_f$  is the area of the face and  $n_i$  is the direction vector in the  $i^{\text{th}}$  direction. The direction of flux is positive if it is incoming for the element at P (aligned with the normal vector at face f). In general, for numerical robustness of the algorithm, it is preferable that:

$$\frac{Q_D}{\Phi_B - \Phi_P} > 0; \quad \Phi_B - \Phi_P \neq 0$$

$$Q_D = 0; \quad \Phi_B - \Phi_P = 0$$

For highly skewed grids, above relation may be violated. If the modifier **ROBU** is specified, then the diffusive flux  $Q_D$  is computed only from the values of  $\Phi_B$  and  $\Phi_P$  and any contribution of the 2<sup>nd</sup> order skew terms is ignored.

$\Phi$  **One or more** of the symbols that denote the dependent variables for which the skew terms are to be included. The valid symbols are listed in Table 2.7.1. If no symbol is specified, then skew terms are included for all variables.

## APPLICABILITY

This command mode is not available for the TIDAL® Software Tool

## EXAMPLES

**CONDUCTION** due to **SKEW** terms to be included for all variables

**CONDUCTION** due to **SECONDARY** terms to be included for **U** and **T**

**CONDUCTION** due to **SECONDARY** and enforce **ROBUST** relation between flux and node values.

**MODE 12:** Tensor Diffusivity with off-Diagonal Terms

**SYNTAX** COND {TENS} [Φ] [USER] [ROBU]

**TENS** By default only the diagonal components of the diffusion tensor,  $\Gamma_{ij}$ , are included in computations. It is assumed that the off-diagonal components ( $i \neq j$ ) are zero. If this modifier is present, the off-diagonal components of the diffusion tensor are also included (see Mode 4 for mathematical expression of diffusive flux). It should be noted, that this mode automatically and simultaneously invokes Mode 4 of command.

This mode of the command is currently available only for the **PORFLOW®** software. Unless individual values of the diffusion components are specified by Mode 2 of the command, the general form of the diffusivity tensor is that given by Scheidegger (1961):

$$\Gamma_{ij} = \alpha_T \delta_{ij} \underline{V} + (\alpha_L - \alpha_T) \frac{v_i v_j}{\underline{V}},$$

$$\underline{V} = (v_i v_i)^{1/2} ; v_i = \frac{V_i}{\theta_E}$$

where  $\alpha_L$  and  $\alpha_T$  are, respectively, the longitudinal and transverse dispersivities for the porous matrix,  $\delta_{ij}$  is the Kronecker delta function,  $v_i$  is the fluid particle velocity vector,  $\underline{V}$  is its magnitude,  $V_i$  is the Darcy velocity vector and  $\theta_E$  is the soil moisture.

**Φ** One or more of the symbols that denote the dependent variables for which the skew terms are to be included. The valid symbols are listed in Table 2.7.1. If no symbol is specified, then the off-diagonal terms are included for all variables.

**USER** By default the tensor diffusivity components are computed from Scheidegger (1961) relations as given above. If this modifier is present, then this computation is by-passed. In this case, each component of the diffusivity tensor must be explicitly specified by the user through Mode 1 of this command.

**ROBU** See COND {SKEW} Mode of the command.

**APPLICABILITY** \_\_\_\_\_

This command mode is not available for the **TIDAL®** Software Tool

**EXAMPLES** \_\_\_\_\_

- CONDuction due to SKEW terms to be included for all variables
- CONDuction with TENSOR diffusivity
- CONDuction TENSOR diffusivity with USER specified values and POSITIVE relation with del\_phi
- CONDuction due to SECONdary terms to be included for U and T

**MODE 13:** Treatment of Conductivity and Diffusivity at External Boundary

**SYNTAX** COND {BOUN} [ $\Phi$ ]

**BOUN** For elements that lie next to the exterior boundary, the conductivity or diffusivity at the external boundary (edge of the element) may differ from the value specified at the interior (node) of the element. The user is therefore expected to specify these properties. If the **BOUNDARY** modifier is specified, then the boundary value of conductivity or diffusivity, for all elements next to boundary, is set equal to the value at the interior node of the element.

$\Phi$  **One or more** of the symbols that represent the variables for which the boundary conductivity or diffusivity is specified. The valid symbols are listed in Table 2.7.1. **If no symbols are specified then the boundary conductivity or diffusivity for all the variables is set equal to the interior node value.**

#### APPLICABILITY

---

This command mode is not available for the TIDAL® Software Tool

#### EXAMPLES

---

**CONDuctivity** BOUNdary for T : Set boundary value equal to interior.

**CONDuctivity** BOUNdary for C : Set boundary value equal to interior. value

**MODE 14:** Elimination of Diffusion Terms

**SYNTAX** COND {OFF|ON} [ $\Phi$ ]

**OFF** The conduction or diffusion terms for the specified variable are completely and globally omitted from the governing equation.

**ON** The conduction or diffusion terms for the specified variable are activated if they were previously turned **OFF**. If there was no previous **OFF** command, then this command is ignored.

$\Phi$  A symbol that denotes the dependent variable for which the diffusion terms are to be eliminated. The valid symbols are listed in Table 2.7.1. *There is no default value; a symbol must be specified.*

#### REMARKS

---

This purpose of this command is to completely omit the diffusion terms from the governing equations. For this command to be effective, there should be no subregion or any other modifier or numerical specification present on the command except.

#### EXAMPLES

---

**COND**uction OFF for the temperature variable: T

## 5.16 CONNECTIVITY

**PURPOSE** To specify element to vertex connectivity for unstructured grid. This command is effective only for the ANSWER® and PORFLOW® Software Tools.

**MODE 1:** Vertex Connectivity for Quad or Hex Elements

**SYNTAX** **CONN** {**VERT**} {**fname**}

**VERT** The element to vertex connectivity is specified. The file must contain as many records as the number of elements specified on the **GRID** command. Each record consists of the element number followed by 4 (for 2D) or 8 (for 3D) vertex numbers of the element corners. Each record is read by the FORTRAN statement:

```
READ (IFILE,*) M,( NV(K),K = 1, NBRMX )
```

where IFILE is an internally assigned file unit number, M, is the element number, NBRMX takes the value of 4 in 2D and value of 8 in 3D, and NV are the vertex numbers for the element corners.

For **2D** geometry, the vertices must be specified in a **counter-clock wise fashion in the x-y plane**, such that the local  $(\xi^n, \eta)$  and the direction normal to the plane form a right handed system. For **3D** geometry, the vertices on "**bottom**" side must be specified first (in counter-clockwise order) followed by the corresponding vertices on "**top**" side, such that the local  $(\xi^n, \eta, \zeta)$  direction forms a right handed system. (Any side may be chosen as the "bottom", then the topologically opposite side is considered to be the "top"). The local  $(\xi^n, \eta, \zeta)$  direction for each element is defined by the order in which the vertices appear on this record. The local  $\xi^n$  axis is oriented from vertex 1 to vertex 2, the  $\eta$  axis from vertex 1 to vertex 4, and the  $\zeta$  axis from vertex 1 to vertex 5. These then determine the local side number (1, 2, 3, 4) or the local X-, X+, Y-, Y+, Z-, Z+ sides which are used to specify the boundary and boundary conditions. These concepts are illustrated in **Figures 1 and 2**. Some further details are also given in Section 3.5. This is the default option.

**fname** The name of the file containing the numerical input for the connectivity. There is no default value; a file name must be specified.

**REMARKS**

An unstructured mesh is defined by:

- (a) A list of vertex numbers (**integer**) and their space coordinates (**real**), and
- (b) A list of element numbers (**integer**) and their vertex numbers (**integer**).
- (c) The list (b) defines the "mapping" from each element to its vertices in list (a).

**EXAMPLES**

**CONN**nectivity information on file "VERT2ELM.CNC"  
**CONN**nectivity of **VERT**ices to elements on file "VERT2ELM.CNC"

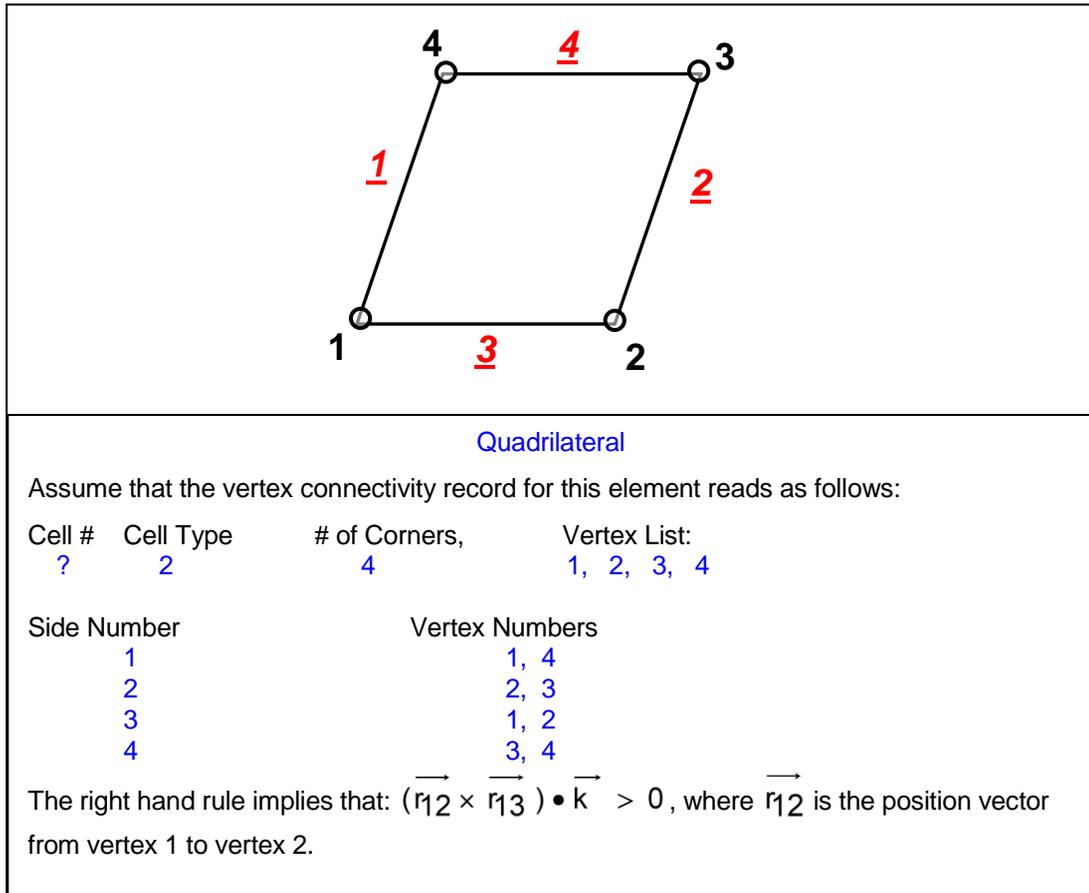
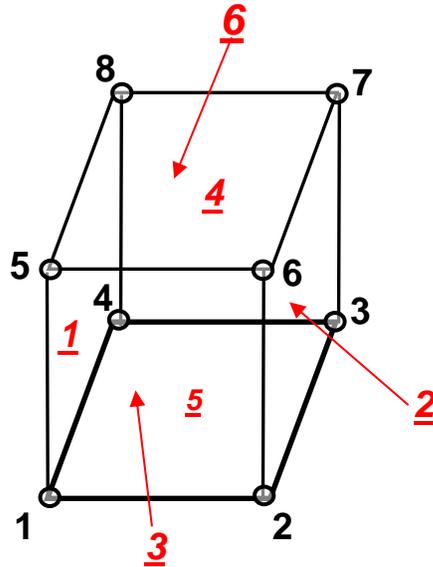


Figure 1: Relation between Vertex Numbers and Side Numbers for a Quadrilateral illustrating the application of the right hand rule. (2D only).



## Hexahedron

Assume that the vertex connectivity record for this element reads as follows:

Cell #,	Cell Type,	# of Corners,	Vertex List:
?	6	8	1, 2, 3, 4, 5, 6, 7, 8

Side Number	Vertex Numbers
1	1, 4, 8, 5
2	3, 2, 6, 7
3	2, 1, 5, 6
4	4, 3, 7, 8
5	1, 2, 3, 4
6	5, 8, 7, 6

The right hand rule implies that:  $(\vec{r}_{12} \times \vec{r}_{14}) \cdot \vec{r}_{15} > 0$ , where  $\vec{r}_{12}$  is the position vector from corner 1 to corner 2.

Note that the side numbering scheme in the hexahedron follows a different scheme from that of other shapes. Here, the opposite sides are numbered consecutively.

**Figure 2: Relation between Vertex Numbers and Side Numbers for a Hexahedron illustrating the application of the right hand rule. (3D only).**

**MODE 2:** Vertex Connectivity for Mixed Hybrid Elements.

**SYNTAX** CONN {HYBR} {fname}

**HYBR** The element to vertex connectivity is specified for a grid with mixed type of elements. Currently 6 different types of elements are allowed. These are given in the Table below.

Element Type	# of Vertices	# of Sides	Geometry	Description
1	3	3	2D	Triangle
2	4	4	2D	Quadrilateral
3	4	4	3D	Tetrahedron
4	5	5	3D	Quad-Based Pyramid
5	6	5	3D	Triangle-based Prism
6	8	6	3D	Hexahedron

The file must contain as many records as the number of elements specified on the **GRID** command. Each record must specify (in order), the element number, element type (given in the Table above), the total number of vertices for that element (given in Table above) and the vertex numbers for the element corners. Each record is read by the FORTRAN statement:

```
READ (IFILE,*) M, MTYPE, NVRTX,( NV(K),K=1,NVRTX)
```

where IFILE is an internally assigned file unit number, M, is the element number, MTYPE is the element type, NVRTX is the number of vertices for the element, and NV are the vertex numbers for the element corners.

Schematic of each element type and its connectivity describing the relationship of the local side numbering to the vertex connectivity is illustrated in Figures 1 through 6. Further details are given in Section 3.5).

**fname** The name of the file containing the numerical input for the connectivity. There is no default value; a file name must be specified.

**EXAMPLES**

---

CONNectivity for HYBRID elements on file "MIXED\_ELEMENTS.CNC"

---

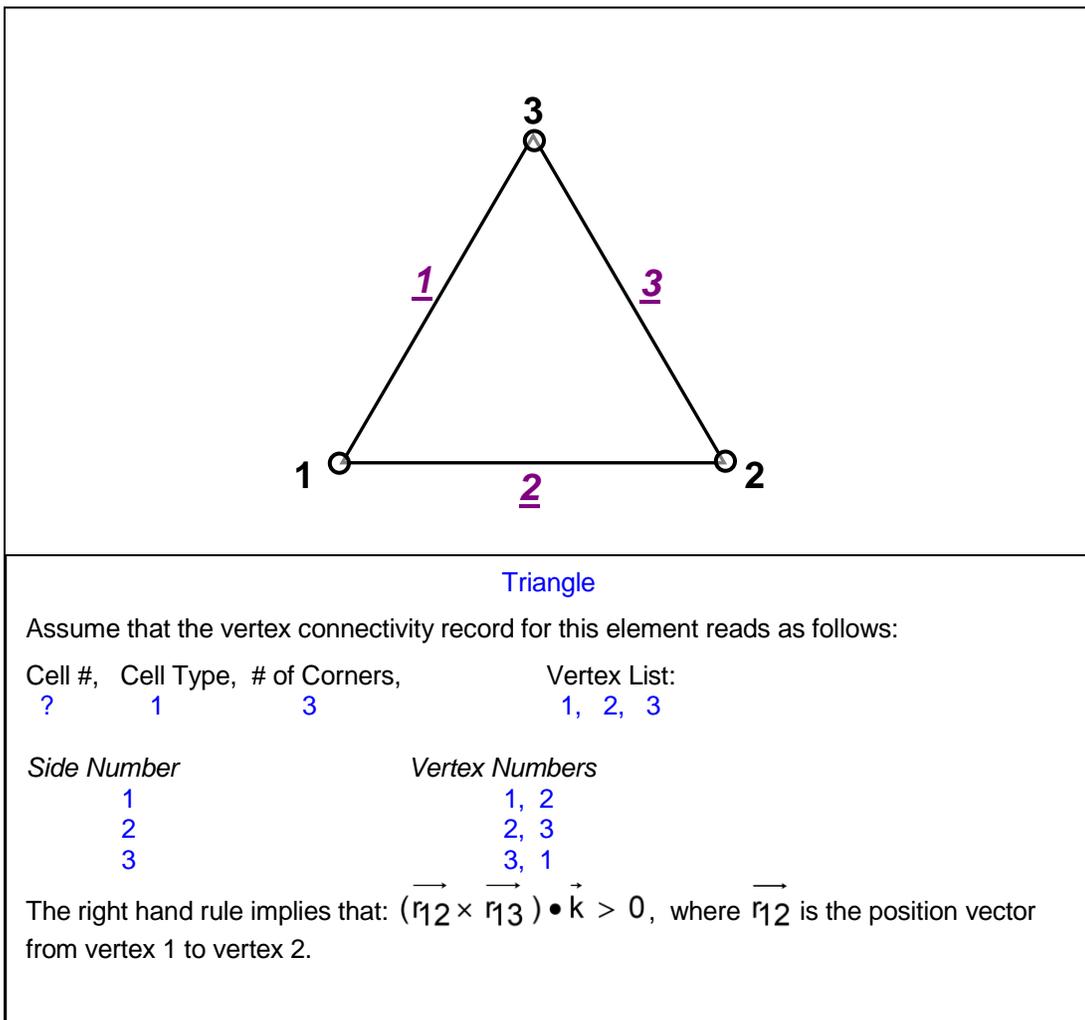


Figure 3: Relation between Vertex Numbers and Side Numbers for a Triangle illustrating the application of the right hand rule. (2D only).

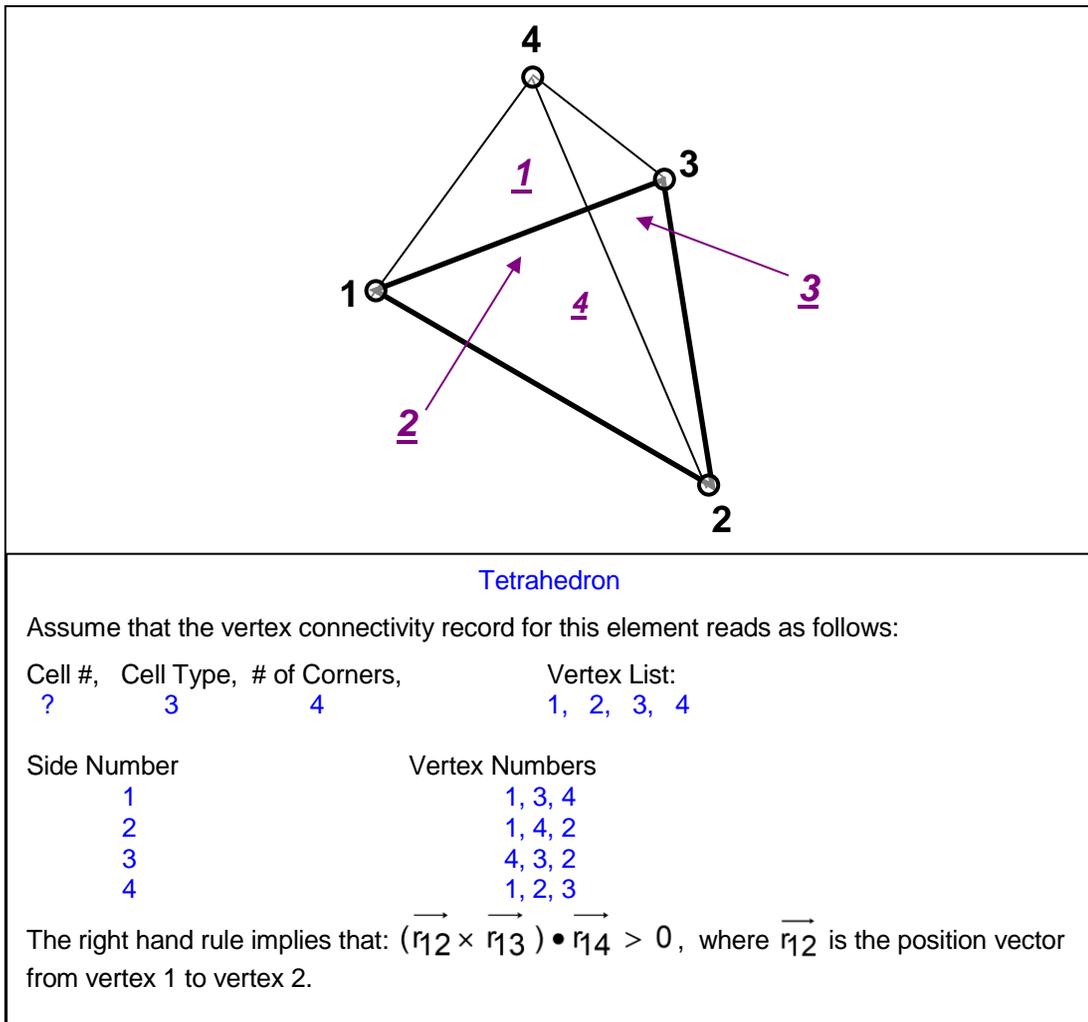
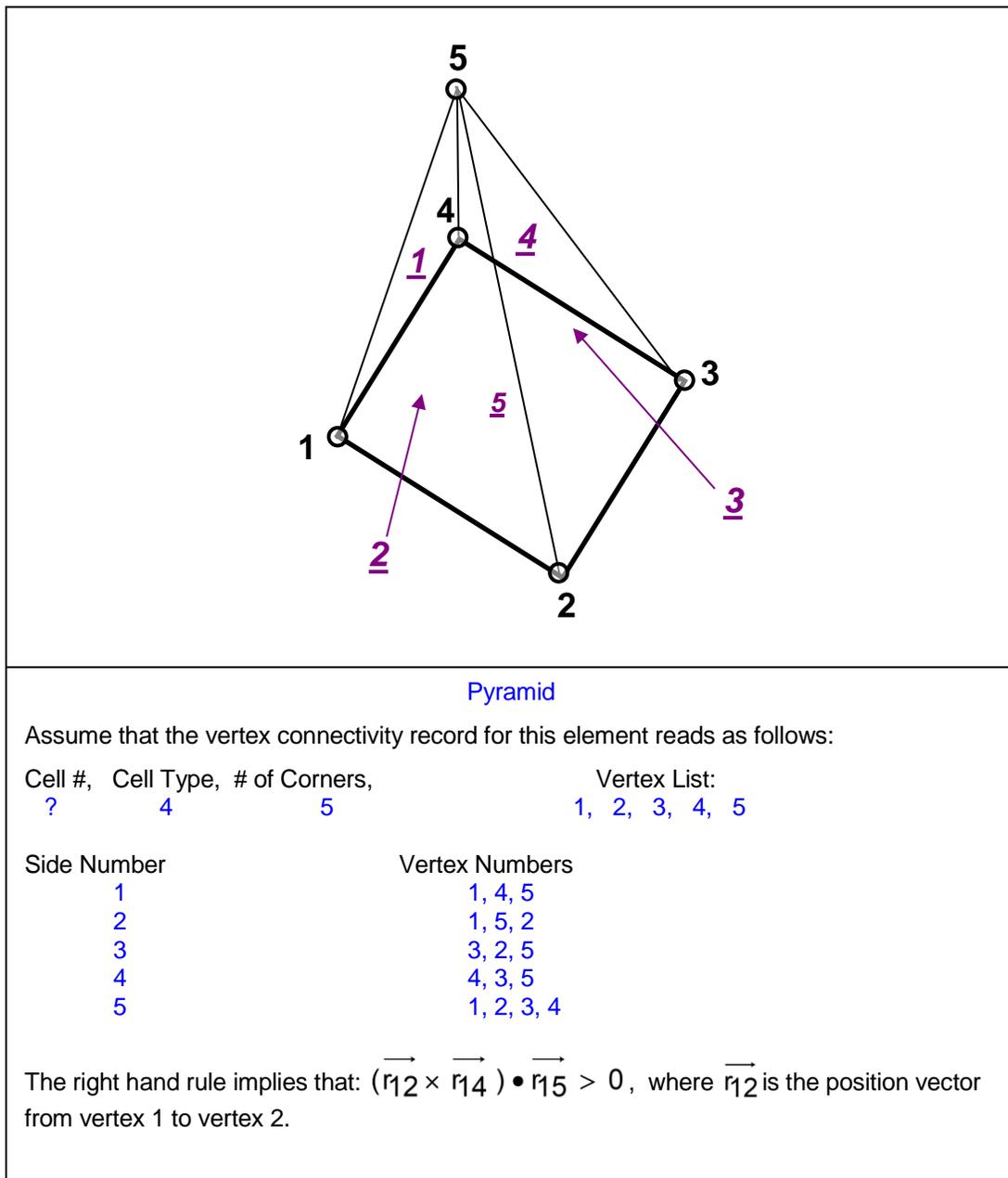


Figure 4: Relation between Vertex Numbers and Side Numbers for a Tetrahedron illustrating the application of the right hand rule. (3D only).



**Figure 5: Relation between Vertex Numbers and Side Numbers for a Pyramid illustrating the application of the right hand rule. (3D only).**

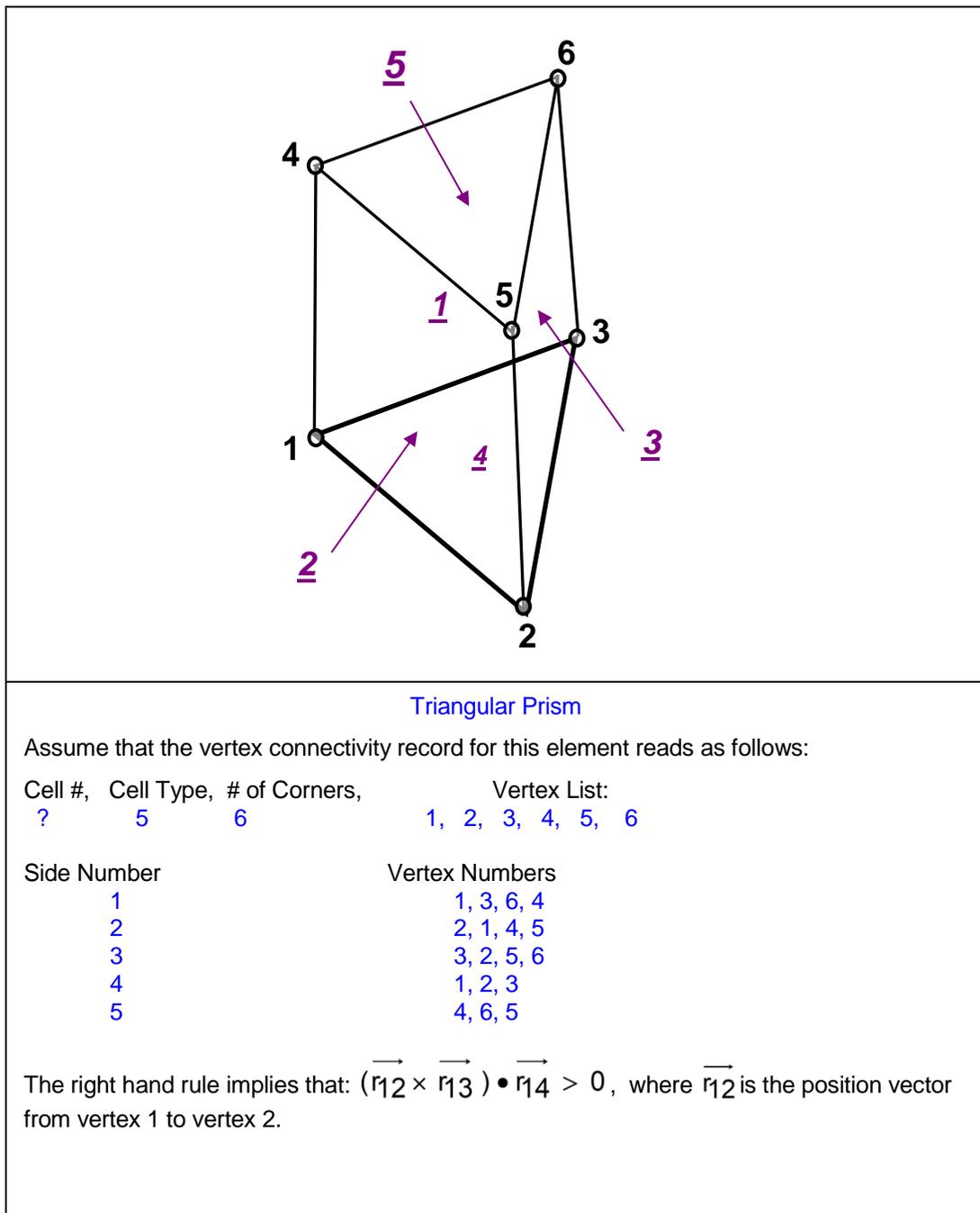


Figure 6: Relation between Vertex Numbers and Side Numbers for a Triangular Prism illustrating the application of the right hand rule. (3D only).

**MODE 3:** Connectivity for Quad or Hex Elements with Split Sides

**SYNTAX** **CONN** {**SPLI**} {**fname**}

**SPLI** The vertex and element connectivity for the split elements is specified. This is a supplementary mode of the command to enable local grid refinement or adaptation of the mesh in selected parts of the domain, based on solution features. (Split sides are element sides with more than one attached neighboring element). It can be used in conjunction with Mode 1 but is not available with Mode 2 of the command.

By default all **ACRi Software Tools** assume that each element is connected to 4 other elements in 2D and 6 other element in 3D geometry. However if the grid is locally refined then a element may be split into multiple “child” elements and some of the elements may be connected to more than the default number of neighboring elements. This supplementary connectivity is specified in the following manner.

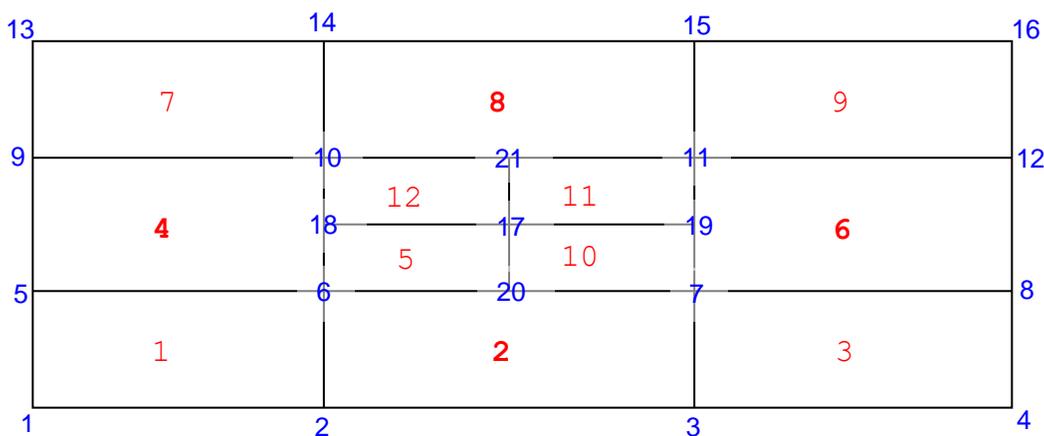
The 1<sup>st</sup> record in the file consists of a header with two numbers: the number of split elements and the total number of data items in the rest of the file. The header is followed by a number of sets of data equal to the number of split elements. The 1<sup>st</sup> record of each set consists of the element number that is split followed by a side index for each side (4 for 2D and 6 for 3D) of the element which denotes the number of neighboring elements (if > 1) connected to that face. The index is zero if there is only one element connected to the side (no split). This record is followed by a list of element numbers that adjoin the split side in the order of the side index. The final record of the set consists of the local side number (from 1 to 6) for the adjoining elements that are attached to the split side.

The entire file is read in using the following two FORTRAN statements:

```
READ(IFILE,*) N_SPLIT_M, N_ITEMS
READ(IFILE,*) ( N_SPL(K), K = 1, N_ITEMS )
```

**fname** The name of the file containing the numerical input for the split connectivity. There is no default value; a file name must be specified.

## EXAMPLES



If the ( Mode 1 ) vertex connectivity for the above mesh is as follows:

Element #	Vertex Connectivity			
1	1	2	6	5
2	2	3	7	6
3	3	4	8	7
4	5	6	10	9
5	6	20	17	18
6	7	8	12	11
7	9	10	14	13
8	10	11	15	14
9	11	12	16	15
10	20	7	19	17
11	17	19	11	21
12	18	17	21	10

Then the SPLIt connectivity command is:

CONNectivity SPLIT on file "SPLIT.CON.

Contents of the file SPLIT. CON are: **(the text in red is for clarity and must NOT be present in the file)**

```

4      36                                ! 4 Split elements; 36 numeric items follow
2                                           ! Split Element Number
0      0      0      2                    ! Face # 4 is split in to 2 segments
5      10                                       ! Element numbers that split face connects to
3      3                                ! Face # 3 (of Element 5) and Face # 3 (of Element 10)
4                                           ! Split Element Number
0      2      0      0                    ! Face # 2 is split in to 2 segments
5      12                                       ! Element numbers that split face connects to
1      1                                ! Face # 1 (of Element 5) and Face # 1 (of Element 12)
6                                           ! Split Element Number
2      0      0      0                    ! Face # 1 is split in to 2 segments
10     11                                       ! Element numbers that split face connects to
2      2                                ! Face # 2 (of Element 10) and Face # 2 (of Element 11)
8                                           ! Split Element Number
0      0      2      0                    ! Face # 3 is split in to 2 segments
12     11                                       ! Element numbers that split face connects to
4      4                                ! Face # 4 (of Element 12) and Face # 4 (of Element 12)

```

## 5.17 CONVERGENCE

**PURPOSE** To specify the convergence criterion for solution of the system of equations.

**MODE 1:** Specification of Convergence Tolerance and Iterations

**SYNTAX** CONV [ $\Phi$ ] [phase] [GLOB|LOCA|DIFF|NORM] [OLD|NEW] [ $\epsilon_{Tol}$ , NITER, NMIN Xref]

**$\Phi$**  A symbol that denotes the dependent variable for which input is specified. The valid symbols are listed in Table 2.7.1. If no symbol is specified then the input is applied to control the outer (total) flow loop, which consists of all the momentum (velocity) and mass equations that are active.

**phase** The fluid phase for which the input is specified. See Section 3.6 for available options. By default the input pertains to the 1<sup>st</sup> phase of the fluid. This modifier is currently available only for the **PORFLOW®** Software Tool.

**GLOB** Convergence is judged by the average residual defined by one of the following equations:

$$\begin{aligned}\epsilon &= \frac{1}{N} \sum_{i=1}^N |A_{ij} X_j - B_i| \\ \epsilon &= \frac{1}{N} \sum_{i=1}^N |X_j^{new} - X_j^{old}| \\ \epsilon &= \frac{1}{N} \sum_{i=1}^N \left| \frac{X_j^{new} - X_j^{old}}{X_j^{old}} \right|; |X_j^{old}| \geq X^{ref}\end{aligned}$$

where A is coefficient matrix, X is the variable vector and B is the forcing function. By default this option is active unless the LOCA modifier is specified. Also, by default, the 1<sup>st</sup> equation is used unless the DIFF modifier is specified.

**LOCA** Convergence is judged by the maximum of the local residual defined by one of the following equations:

$$\begin{aligned}\epsilon &= \max_i |A_{ij} X_j - B_i| \\ \epsilon &= \max_i |X_j^{new} - X_j^{old}| \\ \epsilon &= \max_i \left| \frac{X_j^{new} - X_j^{old}}{X_j^{old}} \right|; |X_j^{old}| \geq X^{ref}\end{aligned}$$

By default the first equation is used unless the DIFF modifier is present.

**DIFF** Convergence is judged on the basis of the 2<sup>nd</sup> or 3<sup>rd</sup> equation above based on whether the GLOB or LOCA mode of the command is selected. In each case, the 2<sup>nd</sup> equation is used by default unless the NORM modifier is present.

**NORM** This modifier is effective only in the presence of the DIFF modifier. In this case, the convergence residual is normalized as given by the 3<sup>rd</sup> equation based on whether the GLOB or LOCA mode of the command is selected.

**OLD** This modifier is effective only for **PORFLOW®** and then only if the automatic time step mode is invoked. In this case, if the time step is decreased, then, by default, the most recent computed values (with a larger time step) are retained if the convergence residue was decreasing monotonically for the previous 3 iterations. However, if this modifier is present, then the computations are restarted from the values at the end of the previous time step.

**NEW** This modifier is effective only if the automatic time step mode of **PORFLOW®** is invoked. If the time step is decreased, then computations restart from the most recent computed values (with a larger time step) rather than the values at the end of the previous time step.

**$\epsilon_{Tol}$**  The maximum threshold tolerance of the residual  $\epsilon$  of equations. The default value is 1.E-6. If the computed  $\epsilon$  exceeds this value, then the matrix solution process is repeated for the

specified number of iterations. For steady state flow, the solution is assumed to have converged if the computed  $\epsilon$  falls below this value for the chosen **REFERENCE** variable (see Mode 2).

- N<sub>ITER</sub>** Maximum number of “outer” iterations for convergence (see comments below). The default value may change for each installation. For most installations, the value is set to 1 with one exception. For the **PORFLOW®** Software Tool, the number of iterations is set to 100 for the mass balance equation if the multi-phase or free-surface mode of the flow (pressure) equation is invoked.
- N<sub>MIN</sub>** Minimum number of iterations for the specified variable. The default value is 1.
- X<sub>ref</sub>** The X<sub>ref</sub> of the normalized residual equations above. The default value is 1.E-7.

## REMARKS

---

If the matrix coefficients are independent of the values of the dependent variable (linear equation) then a solution of the algebraic matrix equations ensures that the differential equation is satisfied. However, if the coefficients are functions of the dependent variable (non-linear equation), then the solution of the matrix equation does not guarantee that the differential equation is satisfied.

This command controls the “outer” iterations. The term “outer” refers to a complete pass through the governing transport equation where (1) the coefficients of the matrix are assembled, (2) the matrix is solved and (3) the convergence norm is checked against the convergence tolerance. For non-linear equations, it is necessary to iteratively perform these 3 steps till the differential equation is satisfied. The **N<sub>ITER</sub>** on this command specifies the number of passes through these steps. The solution of the algebraic matrix of equations is referred to as the process of “inner” iteration and is controlled by the **MATRIX** command.

If no variable ( $\Phi$ ) is specified on this command, then the convergence tolerance applies to the total flow loop consisting of the momentum and mass flow equations (see also Mode 2 of this command). In this case, the number of iterations is interpreted as the total number of times the whole set of velocity and flow equations are sequentially solved till the convergence constraints are met.

The default options and values may depend on your installation. The defaults may be configured differently based on installation-specific optimizations.

## EXAMPLES

---

**CONV**ergence tolerance = 1.E-6  
**CONV**ergence criterion = 1.E-6 for SECOnd phase of fluid  
**CONV**ergence for U in LOCAL mode: value = 1.E-4  
**CONV**ergence for U in DIFFerence mode: value = 1.E-3  
**CONV**ergence for T as REFERENCE variable in GLOBal mode: value = 1.E-4  
**CONV**ergence for T in GLOBal SUM mode: tolerance = 1.E-4  
**CONV** for U: LOCAL mode, value = 1.E-4, max iterations 10, min iter= 5; min value 1.E-5  
**CONV** for flow: LOCAL mode, epsilon = 1.E-2, max iterations = 5  
**CONV** LOCAL, eps=0.001, max iter=50, min = 10, F\_threshold=1.e-5;  
**CONV**ergence epsilon = 1.E-6, 20 iterations; always use OLD values

**MODE 2:** Reference Option for Overall Convergence of Solution Process

**SYNTAX** CONV {REFE} [MOME|FLOW|ALL|Φ] [ε<sub>Tol</sub>, N<sub>ITER</sub>]

**REFE** The specified variable is used as a reference variable for monitoring the convergence of the solution process. The default option is set as follows:

- (1) ANSWER® Software Tool: the MOMEntum option if momentum equations are solved.
- (2) PORFLOW® Software Tool: the FLOW option if pressure equation is solved.
- (3) TIDAL® Software Tool: the FLOW option if momentum equations are solved.
- (4) The 1<sup>st</sup> dependent variable, if no flow equations are solved.

**MOME** The convergence of the solution process is judged on the basis of the individual convergence criteria specified for the momentum and mass balance equation. The process is assumed to converge only when the criteria for the momentum and mass equations are all simultaneously satisfied.

**FLOW** The convergence of the solution process is judged on the basis of mass balance. The flow equations must be solved for this option to be active.

**ALL** The convergence of the solution process is judged on the basis of the individual convergence criteria specified for all active equations. The process is assumed to converge only when the criteria for all equations are simultaneously satisfied.

**Φ** The convergence of the solution process is judged on the basis of the dependent variable denoted by the specified symbol. The valid symbols are listed in Table 2.7.1.

**ε<sub>Tol</sub>** The convergence tolerance ε for the reference variable or quantity. The default value is 1.E-6.

**N<sub>ITER</sub>** Maximum number of iterations for convergence for the iteration process for the selected variable or quantity. If the quantity is flow or momentum then the process consists of the total set of active momentum (velocity) and mass balance equations. The default value is 1.

## REMARKS

---

The default options and values may depend on your installation. The defaults may be configured differently based on installation-specific optimizations.

## EXAMPLES

---

**CONV**ergence REFERENCE based on FLOW with Tolerance = 1.E-6

**CONV**ergence REFERENCE FLOW with Tolerance = 1.E-6 with 10 outer loop iterations

**CONV**ergence REFERENCE based on MOMEntum with Tolerance = 1.E-5

**CONV**ergence REFERENCE based on ALL equations with tolerance =1.E-6

**CONV**ergence REFERENCE based on ALL equations with tolerance =1.E-6 and 5 outer iterations

**MODE 3:**      **Coupled Convergence Test for Species Transport Equations**

**SYNTAX**      **CONV {COUP} {SPEC} [NITER]**

**COUP**      By default, at any time step, the species equations are solved sequentially only once. This means that though inner iteration may be performed on any equation (see Mode 1 of command), each species equation is visited only once at each time step. If **COUPLED** modifier is present then an iterative loop is set up over the complete set of species equations. The equations are solved up to **NITER** times to meet the convergence criteria set up by Mode 1 of the command. The iterative loop is terminated only if either all of the convergence criteria for all the equations are met or the number of specified iterations has been completed.

**SPEC**      This modifier must be present along with the **COUPLED** modifier for this mode of the command to be effective.

**NITER**      Maximum number of iterations for convergence. If a value less than 2 is specified, then **NITER** is set to 2. The default value is 2.

#### **EXAMPLES**

---

**CON**vergence for **SPEC**ies in **COUPLED** mode

**CON**vergence for **SPEC**ies in **COUPLED** mode with a maximum of 10 iterations

**MODE 4: Convergence Based on Averaged Convergence Index****SYNTAX** CONV {AVER} [ NShort ] [ NLong ] [ NStart ] [ α ] [ SHOR|LONG ]**AVER** The convergence is judged by long and short term average of the convergence index computed according to one of the previous modes of the command. Two averaged values are computed:

$$\epsilon_{\text{Long}} = \frac{1}{N_{\text{Long}}} \sum_{i=1}^{N_{\text{Long}}} \epsilon \quad \epsilon_{\text{Long}} = \frac{1}{N_{\text{Long}}} \sum_{i=1}^{N_{\text{Long}}} \epsilon$$

Where  $\epsilon$  is the convergence index computed for the reference variable (See Mode 2 of command). The computations are terminated if:

$$\epsilon > \epsilon_{\text{Short}} > \epsilon_{\text{Long}} \quad \text{and} \quad \epsilon > \alpha \epsilon_{\text{Short}}$$

Where  $\epsilon_{\text{Tol}}$  is the maximum tolerance for the reference variable or quantity (see Mode 2)**NShort** The number of time (or iteration) steps over which the running short-time average is computed. After this number of steps is reached, the old average is discarded and the averaging process starts again. The default value is 7.**NLong** The number of time (or iteration) steps over which the running long-time average is computed. After this number of steps is reached, the old average is discarded and the averaging process starts again. The default value is 31**NStart** The time step or iteration number at which the averaging criterion starts to apply. Often, there are large fluctuations in the matrix residuals at the start due to poor initial conditions. It may be optimal then to start the averaging criteria after the solution process is stabilized. The default value is 1**α** The multiplying factor for termination criterion as in the equation above. The default value is 2**SHOR** By default, the convergence index (defined as  $\epsilon/\epsilon_{\text{Tol}}$ ) is printed out as a diagnostic output (see **DIAGNOSTICS** Command). If this modifier is present, then the short-term average is printed out.**LONG** If this modifier is present, then the long-term average is printed out.**EXAMPLES****CONVERGENCE AVERAGE** over 31 and 151 steps**CONVERGENCE AVERAGE** over 31 and 151 steps. Start at 50 steps and print **LONG** term average**CONVERGENCE AVERAGE** over 31 and 151 and start at 50 steps. Alpha=1.5 ; **LONG** term average

**MODE 5:** Convergence Threshold for Termination of Solution Process

**SYNTAX** CONV {TERM} [ $\Phi_{BIG}$ ]

**TERM** By default, the solution process is terminated if the selected convergence index (**CONVERGENCE REFERENCE** command) exceeds a value of  $10^{30}$  at any stage of the solution process. If this command is present, then the convergence threshold for termination is set to the value specified by  $\Phi_{BIG}$ .

$\Phi_{BIG}$  The threshold value for termination of solution process. The default value is  $10^{30}$ .

#### EXAMPLES

---

CONVergence TERMinate if value exceeds 1.E50.

CONVergence TERMinate if value reaches 1.E16.

## 5.18 COORDINATE

<b>PURPOSE</b>	To specify the grid locations for Cartesian (x, y, z) or cylindrical (x, y, $\theta$ ) coordinates.
<b>MODE 1:</b>	<b>Structured Grid: Coordinate Range</b>
<b>SYNTAX</b>	<b>COOR</b> { <b>RANG</b> } { <b>dir</b> } [ <b>CYLI</b> ] [ <b>DEGR</b> ] [ <b>NODE</b> ] { <b>X<sub>Range</sub></b> } [ <b>R<sub><math>\delta</math>x</sub></b> ]
<b>RANG</b>	The total range (total domain length in the corresponding direction) is specified. The coordinates are calculated internally with a geometric ratio between the successive grid intervals. It is assumed that the coordinate system is structured and orthogonal.
<b>dir</b>	One of the <b>X</b> , <b>Y</b> , <b>Z</b> , <b>R</b> or <b>THETA</b> modifiers that, respectively, denotes the x, y, z, r or $\theta$ coordinates for which input is specified. If <b>R</b> or <b>THETA</b> are specified, then the cylindrical coordinate system is automatically selected.
<b>CYLI</b>	By default the coordinate system is assumed to be Cartesian. If this modifier is present, then a cylindrical coordinate system is selected.
<b>DEGR</b>	By default, the angular input for $\theta$ is assumed to be in radian. If the modifier <b>DEGREE</b> is present, then the input values are in degrees.
<b>NODE</b>	By default, the total range is divided into the specified number of elements and the internal grid nodes are placed at the geometric center of each of these elements. If this modifier is present then the grid nodes are placed at the given locations on this command and the element interfaces are positioned at the middle of the grid nodes.
<b>X<sub>Range</sub></b>	The desired range for the computational domain length in the specified direction for an orthogonal grid. <i>There is no default value. A value must be specified.</i>
<b>R<sub><math>\delta</math>x</sub></b>	The geometric ratio by which the size of the grid element (or grid interval) changes between successive elements (or nodes). <i>The default value is 1.</i>

**EXAMPLES**

**COORDINATE X: RANGE** = 10.  
**COORDINATE X: RANGE** = 10, increase ratio = 1.05!!! Expanding grid  
**COORDINATE X: RANGE** = 10, decrease ratio = 0.95!!! Contracting grid  
**COORDINATE Z: RANGE** = 6.28 implement in the **NODE** mode  
**COORDINATE R: RANGE** = 10, ratio = 0.95  
**COORDINATE R: RANGE** = 10, ratio = 0.95 **CYLINDRICAL**  
**COORDINATE THETA: RANGE** = 270. **DEGREES**

**MODE 2:**      **Structured Grid: Coordinate Minimum and Maximum**

**SYNTAX**      **COOR** {**MINI** | **MAXI**} {**dir**} [**CYLI**] [**DEGR**] [**NODE**] {**X<sub>min</sub>**} {**X<sub>max</sub>**}, [**R<sub>δx</sub>**]

**MINI** | **MAXI**      The minimum and maximum values are specified; the presence of any one of these triggers this mode of the command.

**dir**                See Mode 1 specification.

**CYLI**              See Mode 1 specification.

**DEGR**             See Mode 1 specification.

**NODE**             See Mode 1 specification.

**X<sub>min</sub>**              The minimum coordinate value. **There is no default value; a value must be specified.**

**X<sub>max</sub>**              The maximum coordinate value. **There is no default value; a value must be specified.**

**R<sub>δx</sub>**                The geometric ratio by which the size of the grid element (or grid interval) changes between successive elements (or nodes). **The default value is 1.**

### EXAMPLES

---

**COORDINATE X: MINIMUM** = 0., maximum = 10.

**COORDINATE X: NODE** values: **MINIMUM** = 0., maximum = 10. ratio = 1.05

**COORDINATE X: MINIMUM** = 0., maximum = 10, ratio = 1.05 in **NODE** mode

**COORDINATE Z: MINIMUM** = 0., maximum = 6.28

**COORDINATE THETA: MINIMUM** 0., max = 270. **DEG** for the **NODES**

<b>MODE 3:</b>	<b>Multi-Part Structured Grid with Geometric Ratio Segments</b>
<b>SYNTAX</b>	<b>COOR {RANG   MINI   MAXI} [GLOB] {dir} [CYLI] [DEGR] { X<sub>Range</sub>   X<sub>Min</sub>, X<sub>Max</sub> }, [R<sub>1</sub>, X<sub>1</sub>, N<sub>1</sub>], [R<sub>2</sub>, X<sub>2</sub>, N<sub>2</sub>] ..... [R<sub>N</sub>, X<sub>N</sub>, N<sub>N</sub>]</b>
<b>RANG</b>	The range of the coordinates is specified; the minimum coordinate value is set to zero ( $X_{\min}=0$ ) and the maximum to the range ( $X_{\max}=X_{\text{Range}}$ ).
<b>MINI   MAXI</b>	The minimum and maximum values are specified; the presence of any one of these modifiers triggers this mode of the command.
<b>GLOB</b>	By default, the input of <b>R<sub>1</sub></b> , <b>R<sub>2</sub></b> , ..., <b>R<sub>N</sub></b> is interpreted to be the geometric expansion or compression ratio ( $\delta x_n / \delta x_{n-1}$ ) where $\delta x_n$ is the element (grid) size of the $n^{\text{th}}$ element. If this modifier is present then this input is interpreted to be the ratio of the last grid size to the 1 <sup>st</sup> ( $\delta x_N / \delta x_1$ )
<b>dir</b>	See Mode 1 specification.
<b>CYLI</b>	See Mode 1 specification.
<b>DEGR</b>	See Mode 1 specification.
<b>X<sub>Range</sub></b>	If the modifier <b>RANG</b> is present, then the 1 <sup>st</sup> numerical value is interpreted to be the range of the coordinates. <b>There is no default value; a value must be specified.</b>
<b>X<sub>Min</sub></b>	If the modifier <b>MINI</b> or <b>MAXI</b> is present, then the 1 <sup>st</sup> numerical value is interpreted to be the minimum coordinate value. <b>There is no default value; a value must be specified.</b>
<b>X<sub>Max</sub></b>	If the modifier <b>MINI</b> or <b>MAXI</b> is present, then the 2 <sup>nd</sup> numerical value is interpreted to be the maximum coordinate value. <b>There is no default value; a value must be specified. This value must be omitted if the modifier <b>RANG</b> is present,</b>
<b>R<sub>1</sub></b>	The geometric ratio for grid expansion or compression for the 1 <sup>st</sup> segment of the grid. If the <b>GLOB</b> modifier is present, then it is ratio of the last grid size to the 1 <sup>st</sup> , otherwise it is the ratio of the size of an element to that of the previous element. <b>If no value is specified, or the value is less than <math>10^{-7}</math>, then a value of 1 is assumed.</b>
<b>X<sub>1</sub></b>	The ending coordinate of the 1 <sup>st</sup> grid segment. The grid between the <b>X<sub>Min</sub></b> and <b>X<sub>1</sub></b> coordinates will be generated with a geometric ratio of <b>R<sub>1</sub></b> . If <b>R<sub>1</sub></b> is specified but <b>X<sub>1</sub></b> is not specified then <b>X<sub>1</sub></b> is assumed to be equal to <b>X<sub>Max</sub></b> .
<b>N<sub>1</sub></b>	The number of elements in the 1st grid segment (from <b>X<sub>Min</sub></b> to <b>X<sub>1</sub></b> ). If no value is specified then the default is assumed to be the maximum number of elements specified by the <b>GRID</b> Command.
<b>R<sub>2</sub></b>	The geometric ratio for grid expansion or compression for the 2 <sup>nd</sup> segment (from <b>X<sub>1</sub></b> to <b>X<sub>2</sub></b> ). <b>If no value is specified, or the value is less than <math>10^{-7}</math>, then a value of 1 is assumed.</b>
<b>X<sub>2</sub></b>	The ending coordinate of 2 <sup>nd</sup> grid segment. The grid between the <b>X<sub>1</sub></b> and <b>X<sub>2</sub></b> coordinates will be generated with a geometric ratio of <b>R<sub>2</sub></b> . If <b>R<sub>2</sub></b> is specified but <b>X<sub>2</sub></b> is not specified then <b>X<sub>2</sub></b> is assumed to be equal to <b>X<sub>max</sub></b> .
<b>N<sub>2</sub></b>	The number of elements in the 2nd grid segment (from <b>X<sub>1</sub></b> to <b>X<sub>2</sub></b> ). If no value is specified then the default is assumed to be the remaining number of elements specified by the <b>GRID</b> Command.
<b>R<sub>N</sub></b>	The geometric ratio for the $n^{\text{th}}$ segment (from <b>X<sub>N-1</sub></b> to <b>X<sub>N</sub></b> ) in the manner of <b>R<sub>1</sub></b> and <b>R<sub>2</sub></b> .
<b>X<sub>N</sub></b>	The ending coordinate of the $n^{\text{th}}$ grid segment in the manner of <b>X<sub>1</sub></b> and <b>X<sub>2</sub></b> . If this value is greater than <b>X<sub>max</sub></b> , then it is set equal to <b>X<sub>max</sub></b> .
<b>N<sub>N</sub></b>	The number of elements in the $n^{\text{th}}$ grid segment in the manner of <b>N<sub>1</sub></b> and <b>N<sub>2</sub></b> .

**REMARKS**

---

This command generates a multi-part grid for structured geometry with a different geometric grid size ratio in each segment of the grid. The grid is generated in multiple successive segments from  $X_{Min}$  to  $X_{Max}$  with the 1<sup>st</sup> segment from  $X_{Min}$  to  $X_1$ , the 2<sup>nd</sup> from  $X_1$  to  $X_2$ , and the last from  $X_n=X_{Max}$ . Each segment has its own geometric ratio multiplier and the number of elements. Any number of grid segments can be specified.

**EXAMPLES**

---

**COORDINATE X: RANGE**= 10, ratio = 1.05 in mode

**COORDINATE Y: MINI** = 0., max = 10, R1=1.05, X1=2, N1=5, R2=1.1, X2=5, N2=6, R3=1.2

<b>MODE 4:</b>	<b>Multi-Part Structured Grid with Geometric Ratio Segments and Grid Size Continuity</b>
<b>SYNTAX</b>	<b>COOR {AUTO} {RANG   MINI   MAXI} [GLOB] {dir} [CYLI] [DEGR] { XRange   XMin, XMax }, [R<sub>1</sub>, X<sub>1</sub>, N<sub>1</sub>], [X<sub>2</sub>, N<sub>2</sub>] ..... [X<sub>N</sub>, N<sub>N</sub>]</b>
<b>AUTO</b>	The grid expansion or compression ratios for 2 <sup>nd</sup> through last grid segment are automatically computed. See <b>REMARKS</b> below.
<b>RANG</b>	See the previous mode of the command.
<b>MINI   MAXI</b>	See the previous mode of the command.
<b>GLOB</b>	See the previous mode of the command.
<b>dir</b>	See the previous mode of the command.
<b>CYLI</b>	See the previous mode of the command.
<b>DEGR</b>	See the previous mode of the command.
<b>XRange</b>	See the previous mode of the command.
<b>XMin</b>	See the previous mode of the command.
<b>XMax</b>	See the previous mode of the command.
<b>R<sub>1</sub></b>	See the previous mode of the command.
<b>X<sub>1</sub></b>	See the previous mode of the command.
<b>N<sub>1</sub></b>	See the previous mode of the command.
<b>X<sub>2</sub></b>	See the previous mode of the command..
<b>N<sub>2</sub></b>	See the previous mode of the command.
<b>X<sub>n</sub></b>	See the previous mode of the command.
<b>N<sub>n</sub></b>	See the previous mode of the command.

**REMARKS**

This command generates a multi-part grid for structured geometry mode with a different geometric ratio for grid size in each segment of the grid. The geometric ratio for the 1<sup>st</sup> segment is specified as in the preceding mode of the command. The grid expansion or compression ratios for 2<sup>nd</sup> through last grid segment are automatically computed from the requirement that the starting grid size for each segment is equal to the ending value for the previous segment times the computed grid ratio. This provides continuity in grid size from one segment to the next.

**EXAMPLES**

**COORDINATE X: RANGE** 10. ratio = 1.05

**COORDINATE X: MINIMUM** = 0., maximum = 10, ratio = 1.05 in mode

**COORDINATE Y: RANGE=10 AUTO** Mode R1=1.05, X1=2, N1=5, X2=5, N2=6

**COORDINATE Y: AUTO** Mode **MINI** = 0., max = 10, R1=1.05, X1=2, N1=5, X2=5, N2=6

**MODE 5:**      **Structured Grid: Corners Points of the Computational Domain**

**SYNTAX**      **COOR [CYLI] [DEGR] {N1, ..., Nn}**

**CYLI**            See Mode 1 specification.

**DEGR**            See Mode 1 specification.

**N1, ..., Nn**      The coordinates of the corners of a quadrilateral for 2D simulation or those of a hexahedral for 3D simulations. Thus 8 values (x, y for each of 4 corners) must be specified for 2D and 24 values (x, y, z for each of the 8 corners) for a 3D simulation. In 2D mode, the coordinate values must be in the following order: the lower left, the lower right, the upper left, and the upper right corners. In 3D mode, the values must be in the same order as for the 2D first for the front plane (K=1) and then for the last plane (K=KMAX) of the grid nodes. For each grid line the computational domain is divided in to equal length elements. The nodes are then placed in the middle of each of the element except for the boundary nodes that are placed at the middle of the boundary face.

### EXAMPLES

---

**COOR**dinate corners are: (0., 0.) (1., 0.), (0., 1.) and (1., 1.) !! Unit Square

**COOR**dinate: (0,0) (0.707,0.707), (-0.707,0.707) and (0,1.414)! 45 Deg Square

**COOR**dinate: (0,0) (1., 0.), (0., 1.) and (1,1) CYLIndrical system

**COOR**dinate: 0,0,0 1,0,0 0,1,0 1,1,0 0,0,1 1,0,1 0,1,1 1,1,1 !! Unit Cube

**COOR**dinate: CYLIndrical in DEGRees 0,0,0 1,0, 0 0,1, 0 1,1, 0 0,0,45 1,0,45 0,1,45 1,1,45

**MODE 6:**      **Structured Grid: Individual Coordinates for Orthogonal Grids**

**SYNTAX**      **COOR {dir} [CYLI] [DEGR] [NODE] {N1, ..., Nn}**

**dir**            See Mode 1 specification

**CYLI**           See Mode 1 specification.

**DEGR**          See Mode 1 specification.

**NODE**          By default, the numerical values specify the coordinates of the vertices of the elements for an orthogonal grid. For an orthogonal grid these are the same as the coordinates of the element interfaces. If the modifier **NODE** is present then the numerical values are assumed to be the coordinates of the node points.

**N1,..., Nn**      The 1<sup>st</sup> through N<sup>th</sup> coordinate values for the specified direction. The number of values must equal the corresponding number of grid nodes or element interfaces in that direction as specified by the **GRID** command. In the presence of the **NODE** modifier, the number of values must be equal to the number of nodes (IMAX, JMAX or KMAX) in the specified direction. If **NODE** modifier is omitted, then the number of values must be equal to the number of vertices (IMAX-1, JMAX-1 or KMAX-1) in the specified direction.

#### EXAMPLES

---

**COOR**dinate X: 0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100 !!! element interfaces

**COOR**dinate Y: 0, 5, 15, 25, 35, 45, 55, 65, 75, 85, 95, 100 at NODES

**COOR**dinate R: 0, 5, 15, 25, 35, 45, 55, 65, 75, 85, 95, 100 at NODES

**COOR**dinate Y: 0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100 CYLIIndrical mode

**COOR**dinate THETA: 0, 5, 15, 25, 35, 45, 60, 75, 90, 105, 120 DEGRees

**MODE 7:      Structured or Unstructured Grid: Element Vertex or Grid Node Coordinates****SYNTAX**      **COOR {fname} {dir} [CYLI] [DEGR] [BLOC] [NODE] [PLAN] [JIK|JKI|KIJ|KJI|IKJ] [N1]**

**fname**      The name of the file that contains the coordinate values. See Section 3.3 for additional information. The number of values specified must equal the number of specified directions times the number of grid vertices or grid nodes, as applicable. The specification of grid node coordinates (**NODE** modifier) is available only for a structured grid.

**dir**      One or more of the **X, Y, Z, R** or **THETA** modifiers that, respectively, denote the x, y, z, r or  $\theta$  coordinates. Up to two symbols may be specified for 2D and, 3 for 3D simulations.

**CYLI**      See Mode 1 specification.

**DEGR**      See Mode 1 specification.

**BLOC**      By default, the coordinate values are assumed to be specified as sets of (x, y) or (x, y, z) for each vertex (or node) starting with the 1<sup>st</sup> vertex (or node) and ending with the last one. In the presence of this modifier, it is assumed that the values are specified in “block” mode; that is first all the x-coordinates, followed by the y and z coordinates, respectively.

**NODE**      See Mode 1 specification; this modifier is applicable only for a structured grid.

**PLAN**      This modifier is applicable only for a structured grid. By default, the number of coordinate values must equal the number of corners (or number of nodes). In the presence of this modifier, for a 3D grid, the input is assumed to be for a single z-plane; the other z-planes are assigned the same x and y values.

**JIK**      This modifier is applicable only for a structured grid. By default, the values are read according to the FORTRAN DO loop convention (I, J, K) corresponding to the x, y and z direction respectively. For example, if only x coordinate for a 2D grid is specified, then the grid values  $x_{ij}$  must be in the order:  $x_{11}, x_{21}, x_{31}, \dots, x_{n1}; x_{12}, x_{22}, x_{32}, \dots, x_{n2}; \dots$  and so on. If, say, both x and y coordinates for a 2D grid are simultaneously specified, then the set of values ( $x_{ij}, y_{ij}$ ) must be in the order:  $(x_{11}, y_{11}), (x_{21}, y_{21}), (x_{31}, y_{31}), \dots (x_{n1}, y_{n1}); (x_{12}, y_{12}), (x_{22}, y_{22}), (x_{32}, y_{32}), \dots (x_{n2}, y_{n2}); \dots$  and so on. However, if the JIK modifier is present, then it is assumed that the input data is in the order  $x_{jik}$ . The I and J values are then transposed internally to the ACRi  $x_{ijk}$  format.

**JKI**      This modifier is applicable only for a structured grid. Similar to the **JIK** modifier, except that the input is assumed to be in the (J, K, I) format.

**KIJ**      This modifier is applicable only for a structured grid. Similar to the **JIK** modifier, except that the input is assumed to be in the (K, I, J) format.

**KJI**      This modifier is applicable only for a structured grid. Similar to the **JIK** modifier, except that the input is assumed to be in the (K, J, I) format.

**IKJ**      This modifier is applicable only for a structured grid. Similar to the **JIK** modifier, except that the input is assumed to be in the (I, K, J) format.

**N1**      The number of header records at the beginning of the file. These header records are read and ignored. The coordinate values are assumed to start with the N1+1<sup>st</sup> record. The default value is 0.

**REMARKS**

---

If this command is used for cylindrical geometry in unstructured mode, then the vertices for elements must be defined such that the local element ( $\xi_n$ ,  $\eta$ ,  $\zeta$ ) coordinates are identical with the global ( $X$ ,  $R$ ,  $\theta$ ) coordinates. It is required that the external normal at Face number 3 points in the  $R^-$  and that at Face number 4 in the  $R^+$  direction. Further details are given in Section 3.5 and the **CONNECTIVITY** command.

**EXAMPLES**

---

**COOR**dinate X from file 'XGRID'

**COOR**dinate X, Y and Z from file 'XYZGRID'

**COOR**dinate X, Y and Z from file 'XYZGRID' in BLOCk mode; ignore 7 records

**COOR**dinate X, Y and Z from file 'XYZGRID' in BLOCk IJK mode; ignore 7 records

**COOR**dinate X, Y from file 'XRGRID' in CYLIndrical mode

**COOR**dinate X, R, THETA (DEGREes) from file 'XRTHETA'

**COOR**dinate X, Y and Z from file 'XYZ' in JIK mode for grid NODEs

**MODE 8:**        **Structured or Unstructured Grid: Vertex Coordinates**

**SYNTAX**        **COOR {VERT} {fname}**

**VERT**            The grid coordinates for a set of numbered vertices are specified in the input file.

**fname**           The name of the file that contains the vertex numbers and their corresponding coordinate values. See Section 3.3 for additional information.

#### REMARKS

---

The file format must consist of a number of records each of which must specify a vertex number followed by the set of x, y (for 2D) or x, y, z (for 3D) coordinates of that vertex. The vertex numbers on the record may be in arbitrary order but the total number of records must equal the number of vertices for the problem. **A unique set of coordinate values must be specified for each vertex; no two vertices may be co-located at the same point in space.**

#### EXAMPLES

---

**COOR**dinates of **VERT**ices on file 'PROBGRID'

**5.19 CORIOLIS**

**PURPOSE** To specify the Coriolis parameter for the momentum balance equations. This command is effective only for the **TIDAL®** Software Tool.

**SYNTAX** **CORI** { $\Omega$ }

$\Omega$  The value of  $\Omega$  ( $\Omega \geq 0$ ) the Coriolis parameter. The default value is 0.

**REMARKS**

---

This command is implemented only for the **TIDAL®** Software Tool.

**EXAMPLES**

---

**CORI**olis parameter for this location is 0.05

5.20 CORRELATION

**PURPOSE** Compute and output auto and cross correlations for dependent variables.

**MODE 1:** Auto Correlation for the Variables

**SYNTAX** CORR {Φ} [subrgn] [fname] [TIME] [NOW] [V<sub>frq</sub>] [OFF]

**Φ** A symbol that denotes the variable for which correlation is desired. Valid symbols are listed in Table 2.7.1. If Φ denotes the instant value of a variable at a given location, Φ<sup>n</sup> its value at n<sup>th</sup> time (or iterative) step, then its 1<sup>st</sup> and 2<sup>nd</sup> mean over N steps are defined as:

$$\langle \Phi \rangle = \frac{1}{N} \sum_{n=1}^N \Phi^n$$

$$\langle \Phi^2 \rangle = \frac{1}{N} \sum_{n=1}^N (\Phi^n)^2$$

The following correlation quantities are then computed and printed:

$$R_{11} = \langle \Phi^2 \rangle - \langle \Phi \rangle \langle \Phi \rangle$$

$$C_{11} = \frac{R_{11}}{\langle \Phi \rangle \langle \Phi \rangle}$$

$$D_{11} = \Phi - \langle \Phi \rangle$$

**subrgn** The subregion for computations. If no subregion is specified, the entire domain is selected.

**fname** The file name for output. If a file name is specified then the output is directed to the named file, otherwise the output is directed to the standard output device. If a file name is specified, it must be different from any previously specified file name.

**TIME** By default, V<sub>frq</sub> is interpreted as the frequency of output in terms of number of steps. In the presence of TIME this is the time intervals between successive outputs.

**NOW** A record of is written to the output device at the next time step. This is in addition to the output from the V<sub>frq</sub> specification.

**V<sub>frq</sub>** The frequency (step or time interval) at which the output is written to the output device. In the step interval mode, a specification of 10 results in output at step number 10, 20, 30, etc. In the time interval mode, whether or not the output is obtained exactly at the specified interval depends on the time step specified by the SOLVE command. If an exact time at which the output is due is not simulated, then the output is obtained at the first time step after the due time. By default the output is obtained only at the end of simulations.

**OFF** Any previously specified CORRELATION commands for the specified variable and subregion are disabled. New commands may be subsequently specified.

**EXAMPLES**

- CORRelation for variable U for the SELEcted subregion at the end of simulations
- CORRelation for variable U for the ID=LOC\_AUTO every 25 steps
- CORRelation OFF for variable U for the ID=LOC\_AUTO

**MODE 2:** Cross Correlation for Two Variables**SYNTAX** CORR {  $\Phi_1$ ,  $\Phi_2$  } [subrgn] [fname] [TIME] [NOW] [V<sub>frq</sub>] [OFF]

$\Phi_1$ ,  $\Phi_2$  **Two** symbols that denote the variables for which correlation is desired. Valid symbols are listed in Table 2.7.1. If  $\Phi_k$  denotes the instant value of the  $k^{\text{th}}$  variable at a given location,  $\Phi_k^n$  its value at  $n^{\text{th}}$  time (or iterative) step, then the means over  $N$  steps are defined as:

$$\langle \Phi_k \rangle = \frac{1}{N} \sum_{n=1}^N \Phi_k^n$$

$$\langle \Phi_k^2 \rangle = \frac{1}{N} \sum_{n=1}^N (\Phi_k^n)^2$$

$$\langle \Phi_1 \Phi_2 \rangle = \frac{1}{N} \sum_{n=1}^N (\Phi_1 \Phi_2)$$

The following correlation quantities are then computed and printed:

$$R_{12} = \langle \Phi_1 \Phi_2 \rangle - \langle \Phi_1 \rangle \langle \Phi_2 \rangle$$

$$C_{12} = \frac{R_{12}}{\sqrt{\langle \Phi_1^2 \rangle - \langle \Phi_1 \rangle^2} \sqrt{\langle \Phi_2^2 \rangle - \langle \Phi_2 \rangle^2}}$$

$$D_{12} = (\Phi_1 - \langle \Phi_1 \rangle) (\Phi_2 - \langle \Phi_2 \rangle)$$

**subrgn** The subregion for computations. If no subregion is specified, the entire domain is selected.

**fname** The file name for output. If a file name is specified then the output is directed to the named file, otherwise the output is directed to the standard output device. If a file name is specified, it must be different from any previously specified file name. The total number of open files in any simulation can not exceed 64.

**TIME** By default, **V<sub>frq</sub>** is interpreted as the frequency of output in terms of number of steps. In the presence of **TIME** this is the time intervals between successive outputs.

**NOW** A record of is written to the output device at the next time step. This is in addition to the output from the **V<sub>frq</sub>** specification.

**V<sub>frq</sub>** The frequency (step or time interval) at which the output is written to the output device. In the step interval mode, a specification of 10 results in output at step number 10, 20, 30, etc. In the time interval mode, whether or not the output is obtained exactly at the specified interval depends on the time step specified by the **SOLVE** command. If an exact time at which the output is due is not simulated, then the output is obtained at the first time step after the due time. By default the output is obtained only at the end of simulations.

**OFF** Any previously specified **CORRELATION** commands for the specified variable and subregion are disabled. New commands may be subsequently specified.

**EXAMPLES**

**COR**relation for variable U and V for the **SELE**cted subregion at the end of simulations

**COR**relation for variable U and T for the **ID=LOC\_AUTO** every 25 steps

**COR**relation **OFF** for variable U and T for the **ID=LOC\_AUTO**

<b>MODE 3:</b>	<b>Two Point Correlation for One or Two Variables</b>
<b>SYNTAX</b>	<b>CORR {TWO} { <math>\Phi_1</math> } [ <math>\Phi_2</math> ] [subrgn] [fname] [TIME] [NOW] [V<sub>frq</sub>] [OFF]</b>
<b>TWO</b>	<b>Two point correlation</b> with matched sets of pairs of elements is specified. The subregion for this option must be previously specified with a <b>LOCATE CORRELATION</b> command.
<b><math>\Phi_1, \Phi_2</math></b>	<b>One or Two</b> symbols that denote the variables for which correlation is desired. Valid symbols are listed in Table 2.7.1. The correlation formulae are identical to the two-variable cross-correlation described in Mode 2 except for the definition $\Phi_1$ and $\Phi_2$ . The $\Phi_1$ is always taken to be the values of the 1 <sup>st</sup> specified variable <b>at the elements of the 1<sup>st</sup> set of paired elements</b> . If <b>only one</b> variable is specified then $\Phi_2$ is taken to be the values at the 2 <sup>nd</sup> set of paired elements. If <b>two</b> symbols are specified then $\Phi_2$ is defined to be values of the 2 <sup>nd</sup> variable at the 2 <sup>nd</sup> set of elements.
<b>subrgn</b>	The subregion for computations. <b>The subregion must be previously specified with a LOCATE CORRELATION command. There is no default value.</b>
<b>fname</b>	The file name for output. <b>If a file name is specified then the output is directed to the named file, otherwise the output is directed to the standard output device.</b> If a file name is specified, it must be different from any previously specified file name. The total number of open files in any simulation cannot exceed 64.
<b>TIME</b>	<b>By default, V<sub>frq</sub> is interpreted as the frequency of output in terms of number of steps.</b> In the presence of <b>TIME</b> this is the time interval between successive outputs.
<b>NOW</b>	A record of is written to the output device at the next time step. This is in addition to the output from the <b>V<sub>frq</sub></b> specification.
<b>V<sub>frq</sub></b>	The frequency (step or time interval) at which the output is written to the output device. See Section 3.7 for further details. <b>By default the output is obtained only at the end of simulations.</b>
<b>OFF</b>	Any previously specified <b>CORRELATION</b> commands for the specified variable and subregion are disabled. New commands may be subsequently specified.

## EXAMPLES

---

**COR**relation is **TWO** point type for variable U at ID=LOC\_CORR2 at the end of simulations  
**COR**relation is **TWO** point type for variables U and V at ID=LOC\_CORR2 output on file 'TWOPCORR.UV'  
**COR**relation is **TWO** point type for variables U and V at ID=LOC\_CORR2 at the end of simulations  
**COR**relation is **TWO** point **OFF** for variable U and T for the ID=L OC\_CORR2

**5.21 CPU**

**PURPOSE** To specify the number of Central Processing Units for parallel processing.

**SYNTAX** **CPU** {**N<sub>CPU</sub>**}

**N<sub>CPU</sub>** The number of CPU's for parallel processing. The default value is the smaller of the number of physical processors present and the number of permitted licenses for the software installation.

**EXAMPLES**

---

**CPU** 4 CPU's for this computer system

**5.22 DATUM**

**PURPOSE** To specify datum coordinates,  $x$ , which are used to define the total hydraulic head,  $H$ . This command is effective only for the **ANSWER®** and **PORFLOW®** Software Tools.

**SYNTAX** **DATU** {**N1**, **N2**, **N3**}

**N1, N2, N3** The (x, y, z) or (x, r,  $\theta$ ) datum elevation coordinate values, respectively, which are used to define the total hydraulic head. The default value for all of these is 0.

**REMARKS**

---

The datum coordinates enter the computations only if the multi-phase or free-surface flow options are used, or if the gravitational field is explicitly set to a non-zero value by the **GRAVITY** command.

**EXAMPLES**

---

**DATUm** is at zero

**DATUm** coordinates for this problem are:  $x=0.$ ,  $y=1000$ ,  $z=50$

## 5.23 DEBUG

**PURPOSE** To obtain debug output related to specification of geometrical features, the error indicators and the matrix coefficients.

**MODE 1:** Check Validity of Geometric and Grid Input

**SYNTAX** DEBU {GEOM} [FILE] [OFF]

**GEOM** The geometrical features specified by the user explicitly or by default (such as inlets, outlets, walls, blocked elements, sources and, open, cyclic or symmetric boundaries) are checked for compatibility and validity. Any elements or surfaces with conflicts (such as duplicate or ambiguous assignments) are identified and listed in the standard output file.

Also the grid is checked for orthogonality and uniformity and a summary of grid statistics is written to the standard output file.

**FILE** A detailed output of non-orthogonality angle and grid nodal distance for each element is written to a file. The output is directed to a file with the same name as the Standard Output file but with the extension removed and replaced by “\_GRID\_QUALITY.TMP”. For example, if the Standard Output file is “PROBLEM.OUT”, then the file is named “PROBLEM\_GRID\_QUALITY.TMP”.

The orthogonality index for any element is defined as the maximum of the skewness angles for the faces of the element. For each face the skewness angle is defined as the angle between the normal vector at the face and the vector connecting the two grid nodes, one on either side of the face. The angle is expressed in degrees. A value of 0 implies that the grid vector is orthogonal to the face.

$$\alpha_f = \text{Cos}^{-1} \frac{\sum_{i=1}^{N23} A_i \delta x_i}{A_f \delta S}$$

The nodal distance for any element is defined as the maximum of the Euclidean distance for the nodes straddling the element faces. For any face, the Euclidean distance is the magnitude of the vector connecting the two grid nodes, one on either side of the face.

**OFF** By default these checks are always performed if this modifier is present, then the diagnostic and grid checks are omitted.

**EXAMPLES**

DEBU GEOMetry is on by default

DEBU GEOMetry checks OFF

**MODE 2:** Diagnostic Output of Numerical Error Indicators

**SYNTAX** DEBU { $\Phi$ } [fname] [subrgn] [V<sub>freq</sub>] [TIME]

- $\Phi$**  One and only one symbol for the dependent variable for debug output; valid symbols are listed in Table 2.7.1. There is no default value. A valid symbol must be specified.
- fname** The name of the output file. If no name is specified then the output is directed to a file with the same name as the Standard Output file but with the extension removed and replaced by “\_DEBUG.TMP”. For example, if the Standard Output file is “PROBLEM.OUT”, then the debug file is named “PROBLEM\_DEBUG.TMP”. A summary of output is also printed to the standard output device at the end of simulations.
- subrgn** The subregion for computation of the error parameters. See Section 3.4. If no subregion is specified, the entire computational domain is selected.
- V<sub>freq</sub>** The frequency (step or time interval) at which the output is written to the output device. See Section 3.7 for further details. By default the output is obtained only at the end of simulations.
- TIME** By default, V<sub>freq</sub> is the computation (and output) frequency in terms of number of steps. If this modifier is present, then V<sub>freq</sub> is the time interval between successive outputs.

## REMARKS

---

The debug output consists of the local Peclet and Courant numbers. These error-indicators provide a measure of the numerical error in the solution. Peclet number, the ratio of the convection to the diffusion term, is generally considered to be indicative of the “numerical diffusion” in the solution. The Courant number, the ratio of the convection to the accumulation term, is generally considered to be indicative of the “dispersion” in the transient component of the solution. These error indicators are based on the Taylor series analysis of the governing equation and are known to be inadequate for prediction of numerical error in many practical situations. Therefore caution should be exercised in the interpretation of these error indicators. Peclet number can not be defined for the pressure equation since it does not have a convection term. Similarly, no Courant number can be defined for steady state computation since the accumulation term is identically equal to zero.

## EXAMPLES

---

DEBU output of error indicators for T

DEBU output for T every 12 steps

DEBU output for T for SELEcted subregion at TIME interval of 1.5 units

DEBU output for T for subregion with ID=SUBZone at TIME interval of 1.5 units

DEBU specified output: for T on 'MYFILE.DBG'

**MODE 3:** Diagnostic Output of Matrix Coefficients

**SYNTAX** DEBU {MATR} [NStep]

**MATR** The debug output pertaining to the matrix of the algebraic equations is produced on a file at the computation step specified by the N1 input. The governing algebraic matrix equation is:

$$A_{ij}X_j = B_i$$

where, for the  $i^{\text{th}}$  element,  $A_{ij}$  are the matrix coefficients,  $X_j$  are the values of the dependent variable and  $B_i$  is the forcing function.

One file is generated for each governing transport equation. Each file is named `acr_MATRxxxx.TMP` where "xxxx" are replaced by the non-blank characters of the 1<sup>st</sup> four characters of the symbolic name of the corresponding variable. If the name of the variable has fewer than 4 non-blank characters, then the trailing characters will stay as `xx`. The records on each file are in the order described below.

The first record consists of: the symbolic name of the variable, the order number of the variable in the master storage array, the step number at which the output is produced, the number of internal field elements, the number of neighbors for each element, the grid dimensions in each direction and, the total number of nodes.

This is followed by one record for each field element in sequence. Each record consists of: the value of the variable ( $X_j$ ), the source term (part of  $B_i$  that pertains to the source or sinks in physical units), the forcing function ( $B_i$ ), the diagonal component of the matrix coefficients ( $A_{ii}$ ) and, the negative of the matrix coefficient for each neighbor ( $-A_{ij}$ ,  $i \neq j$ ).

**NStep** The step number at which output is required. There is no default value; a value must be specified.

## EXAMPLES

---

DEBU MATRix information at step # 9 for each equation in turn

## 5.24 DECAy

**PURPOSE** To specify rate constants and mode of decay of a dependent variable due to physical, chemical or radioactive decay.

**MODE 1:** Direct Linear Decay

**SYNTAX** DECA { $\Phi$ } [RATE|LIFE] [FIEL] [ $\lambda$ ]

$\Phi$  A symbol that denotes the dependent variable for which the decay or reaction rate constant is specified. The valid symbols are listed in Table 2.7.1.

**RATE** The decay rate of  $\Phi$  is specified; the units are those of inverse of time. This is the default option.

**LIFE** The half-life of  $\Phi$  is specified; the units are those of time.

**FIEL** If the modifier **FIELD** is present, then the decay reaction is assumed to occur only inside the field and not at the boundaries. The boundary values are maintained at the specified conditions.

$\lambda$  The decay rate of  $\Phi$  in the presence of the **RATE** modifier or the half-life of decay in the presence of the **LIFE** modifier. The default value is set so that no decay occurs.

**EXAMPLES**

---

DECAy rate of C is 0.001

DECAy rate of FU is 0.001; only for FIELd values

DECAy half - LIFE of C is 1.632

DECAy half - LIFE of C is 15.00 in the FIELd only

**MODE 2:** Drag Type Of Decay Based On Flow Speed

**SYNTAX** DECA { $\Phi$ } [DRAG] [option] [subrgn] [ $C_D$ ] [ $\alpha$ ] [ $N_1, N_k$ ] [ $N_{k+1}$ ]

$\Phi$  See Mode 1 specification.

**DRAG** The decay of dependent variable  $\Phi$  is according to the drag law based on the flow speed. This adds a source,  $S_\Phi$ , in the transport equation for  $\Phi$  given by:

$$S_\Phi = 0.5 \rho \Phi C_D C_\Phi (U^2 + V^2 + W^2)^{\alpha/2} / \delta_V,$$

where  $C_D$  is a non-dimensional drag coefficient,  $C_\Phi$  is a scaling factor,  $U$ ,  $V$  and  $W$  are components of flow velocity and  $\delta_V$  is the volume of the element where the source is introduced. The units of  $C_\Phi$  are such that  $S_\Phi$  has units of  $[\rho\Phi/\text{Time}]$ . For example if  $\Phi$  is a velocity component, then the units of  $S_\Phi$   $[\text{M L}^{-2} \text{T}^{-2}]$  are those of force per unit volume. In this case if  $N=1$ , then  $C_\Phi$   $[\text{L}^2]$  is the area subject to drag.

**option** Options selected for implementation of the source.

option	Interpretation
<b>VOLU</b>	The coefficient $C_\Phi$ in the $S_\Phi$ term is set equal to the volume ( $\delta_V$ ) of the element.
<b>AREA</b>	The coefficient $C_\Phi$ in the $S_\Phi$ term is set equal to the area ( $\delta_A$ ) of the element face indicated by the <b>dir</b> modifier.
<b>dir</b>	The orientation index for the element boundary associated with the source. See Section 3.5 for available choices. There is no default value for this input.
<b>NORM</b>	The coefficient $C_\Phi$ in the $S_\Phi$ term is computed as: $C_\Phi = \sum_i A_i \cdot V_i$ where $A_i$ is the $i^{\text{th}}$ direction component of the element boundary area specified by <b>dir</b> . $V_i$ are the values specified by <b>N3</b> through <b>Nk</b> (2 for 2D, and 3 for 3D).
<b>DENS</b>	The computed source, $C_\Phi$ , is further multiplied by density. The density may be specified as the last value, <b>Nk+1</b> , on the command. If this value is omitted, then the boundary value in the <b>dir</b> direction is used if the <b>AREA</b> or <b>NORMAL</b> modifiers are present, otherwise the local density for the element is used.

**subrgn** The subregion for which the input is specified. See Section 3.4. If no subregion is specified, the entire computational domain is selected

**$C_D$**  The drag coefficient,  $C_D$ , in the drag law (see Comments). The default value is 0.

**$\alpha$**  The power exponent,  $N$ , in the drag law (see Comments). The default value is 1.

**$N_1, \dots, N_k$**  The normalizing vector components,  $V_i$ , if the **NORMAL** modifier is present. Two values must be specified for 2D and 3 for 3D flows. There are no default values for this input.

**$N_{k+1}$**  The density value that multiplies the computed source, if the **DENSITY** modifier is present. There is no default values for this input.

## EXAMPLES

**DECA**y of U: DRAG law: cf=0.001, for previously SELEcted subregion

**DECA**y of T: DRAG law: cf=0.002, N = 0.80 for subregion ID=OBSTruction

**DECA**y for T: DRAG type: cf=0.001, N=0.5 multiply by AREA in X- direction for SELEcted subregion

**DECA**y for T: DRAG type: cf=0.1, N=0.5 X- dir & multiply by VOLUme. DENSity for SELEcted subregion

**DECA**y for T: DRAG type: cf=0.1, N=0.5 X- direction NORMAlized 1. 0. DENSity for SELEcted subregion

## 5.25 DEFINE

**PURPOSE** To define the value of a symbolic variable as a numeric or character expression.

**MODE 1:** Assign Numeric Values to Symbolic Variables

**SYNTAX** **DEFI** {**X**<sub>1</sub>=**V**<sub>1</sub>} [**X**<sub>2</sub>=**V**<sub>2</sub>], ....., [**X**<sub>n</sub>=**V**<sub>n</sub>] [**OFF**]

**X**<sub>1</sub>, **X**<sub>2</sub>,...**X**<sub>n</sub> A list of symbolic variables that at run-time is replaced by the corresponding numerical value, **V**<sub>1</sub>, **V**<sub>2</sub>,...**V**<sub>n</sub> in the same order. Once variable has been defined, the corresponding numeric value will replace any occurrence of this variable in the user input.

The symbolic variable appearing on a **DEFINE** command must be distinct and must not be one of the defined field variables listed in Tables 2.7.1 or 2.8.1-3. Neither should it conflict with any command Keyword or modifier that appears on these commands. Only the first 8 characters in the symbolic variables are meaningful; any subsequent characters are ignored. The symbolic variable must start with an alphabetic character (A-Z) and the rest of the characters must be alphabetic (A-Z) or numeric (0-9). No case distinction is made and no other characters are allowed.

**V**<sub>1</sub>, **V**<sub>2</sub>,...**V**<sub>n</sub> The numerical values that replaces every occurrence of the **X**<sub>1</sub>, **X**<sub>2</sub>,...**X**<sub>n</sub> in the user input in the order of assignment.

**OFF** The previously defined variables are deactivated. New variables may be defined for input that follows.

**REMARKS**

This command provides a powerful means of performing symbolic math and/or creating 'prototype' data input files in which symbolic variables may be used to denote problem-specific input data. These symbolic variables may then be defined at the beginning of the input or they can be replaced by character or numeric data at run time by using the interactive input feature described in Section 3.2. Up to 256 symbolic variables may be defined at any time.

**EXAMPLES**

**DEFINE XXX** = 1.2345 in all input that follows

**DEFINE TWOPI** = 2.2830, **PI** = 3.1415, **PIBY2** = 1.57075. **PIBYFOUR** = 0.785375.

**DEFINE OFF** for all previous variables

**MODE 2:** Assign Symbolic Values to Symbolic Variables

**SYNTAX** **DEFI** { $X_1=Y_1$ } [ $X_2=Y_2$ ], ..., [ $X_n=Y_n$ ] [**OFF**]

$X_1, X_2, \dots, X_n$  Symbolic variables that at run-time are replaced by the corresponding character strings  $Y_1, Y_2, \dots, Y_n$  in the same order. Once variable has been defined, the corresponding symbol will replace any occurrence of this variable in the user input.

The symbolic variable appearing on a **DEFINE** command must be distinct and must not be one of the defined field variables listed in Tables 2.7.1 or 2.8.1-3. Neither should it conflict with any command Keyword or modifier that appears on these commands. Only the first 8 characters in the symbolic variables are meaningful; any subsequent characters are ignored. The symbolic variable must start with an alphabetic character (A-Z) and the rest of the characters must be alphabetic (A-Z) or numeric (0-9). No case distinction is made and no other characters are allowed.

$Y_1, Y_2, \dots, Y_n$  The symbols that replace every occurrence of the corresponding  $X_1, X_2, \dots, X_n$  in the user input in the order of assignment. Only the first 8 characters in the symbols are considered meaningful; any subsequent characters are ignored. Each symbol must start with an alphabetic character (A-Z) and the rest of the characters must be alphabetic (A-Z) or numeric (0-9). No case distinction is made and no other characters are allowed.

**OFF** The previously defined variables are deactivated. New variables may be defined for input that follows.

#### REMARKS

---

This command provides a powerful means of performing symbolic math and/or creating 'prototype' data input files in which symbolic variables may be used to denote problem-specific input data. These symbolic variables may then be defined at the beginning of the input or they can be replaced by character or numeric data at run time by using the interactive input feature described in Section 3.2. Up to 256 symbolic variables may be defined at any time. **DEFINE** Command is processed after the **INCLUDE** and **META** commands. It cannot be used to specify the name of a file that is specified on an **INCLUDE** or **META** command.

#### EXAMPLES

---

**DEFINE XXX** = **P** in all input that follows  
**DEFINE XXX** = **U234**, **YYY** = **U235** and **ZZZ** = **U238**.  
**DEFINE OFF** for all previous variables

**MODE 3:** Multiple Simulations of a Given Input for a Series of Assigned Values

**SYNTAX** **DEFI** {variable} {LIST} {V<sub>1</sub>, V<sub>2</sub>, ..., V<sub>n</sub>}

**variable** A symbolic variable, that at run-time, is replaced by user specified input value to create multiple sets of data (See **REMARKS** below). **The 'variable' must be the 1<sup>st</sup> modifier following the DEFINE command.** Only the first 8 characters in the symbolic variables are meaningful; any subsequent characters are ignored. The symbolic variable must start with an alphabetic character (A-Z) and the rest of the characters must be alphabetic (A-Z) or numeric (0-9). No case distinction is made and no other characters are allowed.

The symbolic variable for each **DEFINE** command must be distinct and must not be one of the defined field variables listed in Tables 2.7.1 or 2.8.1-3. Once a variable has been defined, a numeric value selected from the specified list replaces ALL occurrences of this **variable** in the user input.

**LIST** A list of values to generate a sequence of inputs with assigned values is specified. A **DEFINE LIST** command must not appear if a metafile for input is specified (see **META** command).

**V<sub>1</sub>, V<sub>2</sub>, ..., V<sub>n</sub>** The numerical values that sequentially replace the **variable** in each data set. See **REMARKS** below. A maximum of 32 values can be specified, excess values are discarded.

## REMARKS

This command provides a powerful means of performing multiple simulations with a given data set where one or more symbolic variables in the input are sequentially replaced with the specified values to generate multiple input data sets. **Up to 8 DEFINE LIST or DEFINE DO commands can occur in a data set and these can generate up to 256 simulations from a given data set.** The resulting “unfolded” data set is saved in a file that has the same name as the input file but with “\_DEFINE\_UNFOLD.DAT” appended to the file name. For example, if the input file is called “PROB.DAT”, then the unfolded input file will be called “PROB\_DEFINE\_UNFOLD.DAT”.

Output for all the simulations is directed to a single Standard Output Unit file) with the name as specified by the user or the default name generated automatically (see **OUTPUT** Command). A separate archive file is generated for each of the problems. If a **SAVE** command is present then the name of the archive file for each problem is the file name given in the command with the string “0nnn” appended to the name of the file where “nnn” represents the problem number in the sequence. For example if the file name on the **SAVE** command is “PROB.SAV”, then the names of the archive files will be “PROB\_0001.SAV”, “PROB\_0002.SAV”, etc. in order for each problem. If no file name is present then a default file name for the archive file is generated (see **SAVE** command). If no **SAVE** command is present, then a default **SAVE** command is automatically added.

Consider a set of statements in an input file, “PROB.DAT”:

```
TITLE: Test Problem for Parametric Analysis by creating Multiple sets of Input Data
GRID: 22 in x and 12 in y direction
DEFINE VAL_LEFT LIST 100, 120, 130
BOUNDARY T at X- value = VAL_LEFT
(Other input follows here)
SOLVE in STEADY mode for maximum of 1000 steps
END
```

The above input data will result in 3 problems being solved where the symbolic variable **VAL\_LEFT** will be successively assigned the value of 100, 120 and 130. The resulting unfolded data set is saved in file named “PROB\_DEFINE\_UNFOLD.DAT” with 3 problems as follows:

```

|*****
! Problem # 1 Created from DEFINE DO or LIST Command
|*****
TITLE: Test Problem for Parametric Analysis by creating Multiple sets of Input Data
GRID: 22 in x and 12 in y direction
DEFINE VAL_LEFT LIST 100
BOUNDARY T at X- value = VAL_LEFT
(Other input follows here)
SOLVE in STEADY mode for maximum of 1000 steps
SAVE inserted "PROB_0001.SAV"
END
!
|*****
! Problem # 2 Created from DEFINE DO or LIST Command
|*****
TITLE: Test Problem for Parametric Analysis by creating Multiple sets of Input Data
GRID: 22 in x and 12 in y direction
DEFINE VAL_LEFT LIST 120
BOUNDARY T at X- value = VAL_LEFT
(Other input follows here)
SOLVE in STEADY mode for maximum of 1000 steps
SAVE inserted "PROB_0002.SAV"
END
!
|*****
! Problem # 3 Created from DEFINE DO or LIST Command
|*****
TITLE: Test Problem for Parametric Analysis by creating Multiple sets of Input Data
GRID: 22 in x and 12 in y direction
DEFINE VAL_LEFT LIST 130
BOUNDARY T at X- value = VAL_LEFT
(Other input follows here)
SOLVE in STEADY mode for maximum of 1000 steps
SAVE inserted "PROB_0003.SAV"
END

```

## EXAMPLES

---

**DEFIne** XXX LIST of values: 1., 2., 3., 4

**MODE 4:** Multiple Simulations of a Given Input for a Series of Assigned Values

**SYNTAX** **DEFI** {variable} {DO} {V<sub>1</sub>, V<sub>2</sub>, V<sub>3</sub>}

**variable** A symbolic variable, that at run-time, is replaced by user specified input value to create multiple sets of data (See **REMARKS** below). **The 'variable' must be the 1<sup>st</sup> modifier following the DEFINE command. Only the first 8 characters are meaningful; any subsequent characters are ignored.** The symbolic variable for each **DEFINE** statement must be distinct and must not be one of the defined field variables listed in Tables 2.7.1 or 2.8.1-3. Once a variable has been defined, a numeric value selected from the specified list replaces ALL occurrences of this **variable** in the user input.

**DO** A DO Loop is specified to generate a sequence of inputs from the assigned values. A **DEFINE DO** command must not appear if a metafile for input is specified (see **META** command).

**V<sub>1</sub>, V<sub>2</sub>, V<sub>3</sub>** The numerical values that generate a sequence of values to replace the **variable** in each data set. **V<sub>1</sub>** is the starting value, **V<sub>2</sub>** the ending value, and **V<sub>3</sub>** the increment at which the values are generated. See **REMARKS** below. **All 3 numerical values must appear on the command; continuation records are not allowed. A maximum of 32 values are computed from the DO loop; any excess values are discarded.**

**REMARKS**

---

This command provides a powerful means of performing multiple simulations with a given data set where one or more symbolic variables in the input are sequentially replaced with the specified values to generate multiple input data sets. **Up to 8 DEFINE LIST or DEFINE DO commands can occur in a data set and these can generate up to 256 simulations from a given data set.** The resulting “unfolded” data set is saved in a file that has the same name as the input file but with “\_DEFINE\_UNFOLD.DAT” appended to the file name. For example, if the input file is called “PROB.DAT”, then the unfolded input file will be called “PROB\_DEFINE\_UNFOLD.DAT”.

Output for all the simulations is directed to a single Standard Output Unit file) with the name as specified by the user or the default name generated automatically (see **OUTPUT** Command). A separate archive file is generated for each of the problems. If a **SAVE** command is present then the name of the archive file for each problem is the file name given in the command with the string “0nnn” appended to the name of the file where “nnn” represents the problem number in the sequence. For example if the file name on the **SAVE** command is “PROB.SAV”, then the names of the archive files will be “PROB\_0001.SAV”, “PROB\_0002.SAV”, etc. in order for each problem. If no file name is present then a default file name for the archive file is generated (see **SAVE** command). **If no SAVE command is present, then a default SAVE command is automatically added.**

Consider a set of statements in an input file, “PROB.DAT”:

```
TITLE: Test Problem for Parametric Analysis by creating Multiple sets of Input Data
GRID: 22 in x and 12 in y direction
DEFINE VAL_LEFT DO 100, 125, 150
BOUNDARY T at X- value = VAL_LEFT
(Other input follows here)
SOLVE in STEADY mode for maximum of 1000 steps
END
```

The above input data will result in 3 problems being solved where the variable **VAL\_LEFT** will be successively assigned the value of 100, 125 and 150. The resulting unfolded data set is saved in file named “PROB\_DEFINE\_UNFOLD.DAT” with 3 problems as follows:

```
!*****
! Problem # 1 created from DEFINE DO or LIST Command
!*****
TITLE: Test Problem for Parametric Analysis by creating Multiple sets of Input Data
GRID: 22 in x and 12 in y direction
DEFINE VAL_LEFT LIST 100
BOUNDARY T at X- value = VAL_LEFT
(Other input follows here)
SOLVE in STEADY mode for maximum of 1000 steps
SAVE inserted “PROB_0001.SAV”
```

```

END
!
|*****
! Problem # 2 created from DEFINE DO or LIST Command
|*****
TITLE: Test Problem for Parametric Analysis by creating Multiple sets of Input Data
GRID: 22 in x and 12 in y direction
DEFINE VAL_LEFT LIST 125
BOUNDARY T at X- value = VAL_LEFT
(Other input follows here)
SOLVE in STEADY mode for maximum of 1000 steps
SAVE inserted "PROB_0002.SAV"
END
!
|*****
! Problem # 3 created from DEFINE DO or LIST Command
|*****
TITLE: Test Problem for Parametric Analysis by creating Multiple sets of Input Data
GRID: 22 in x and 12 in y direction
DEFINE VAL_LEFT LIST 150
BOUNDARY T at X- value = VAL_LEFT
(Other input follows here)
SOLVE in STEADY mode for maximum of 1000 steps
SAVE inserted "PROB_0003.SAV"
END

```

## EXAMPLES

---

**DEFIne** XXX DO start at 100, end at 200, interval =20 ! 5 sets (100 120,140,160, 200) generated

## 5.26 DENSITY

**PURPOSE** To specify the options and constants used to calculate fluid density.

**MODE 1:** Fluid Density as a Constant or General Function

**SYNTAX** **DENS** {func[ξ]} {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub>|fname} [phase] [subrgn] [STAT]

**subrgn** The subregion for which the input is specified. See Section 3.4. If no subregion is specified, the entire computational domain is selected.

**func** One of the modifiers listed in Table 4.2.1 that denotes the functional form of the input. For this input, the function specifies the value of the density for the corresponding phase. If no function is specified then the value is assumed to be constant.

**ξ** One of the independent variables listed in Table 4.2.2. If no variable is specified, then the independent variable is assumed to be time.

**N<sub>1</sub>,...,N<sub>n</sub>** The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.

**fname** The name of the file containing the numerical values N<sub>2</sub> through N<sub>n</sub>. This option is available only for selected functions. See Section 3.3 for additional information.

**phase** The phase for which the input is specified. See Section 3.6 for available options. By default the input pertains to the 1<sup>st</sup> phase of the fluid. This modifier is available only for the multi-phase versions of the **PORFLOW®** and **ANSWER®** Software Tools.

**subrgn** The subregion for which the input is specified. See Section 3.4. If no subregion is specified, the entire computational domain is selected.

**STAT** By default for transient flows, the volumetric change due to  $\partial\rho/\partial t$  term is automatically accounted for. If this modifier is present, then the effect of this term in the continuity equation is ignored. This modifier is effective only for the **ANSWER®** Software Tool. The use of this modifier is not recommended; it is being retained for backward compatibility.

**APPLICABILITY**

The general functional form of this command is currently available only for the **ANSWER®** and **PORFLOW®** Software Tools. Only the constant density option is available for **TIDAL®** Software Tool.

**EXAMPLES**

**DENSITY** of fluid = 1.18 Kg/m<sup>3</sup>

**DENSITY** of fluid is = 1000

Generic examples for this command are given in Section 4.4. The command keyword (**DENS**) must replace the keyword used in these examples.

**MODE 2:** Special Density Functions

**SYNTAX** DENS {POWE|POLY|LINE} [phase] [ρ\*] [α] [β] [γ]

**POWE** Density varies according to the power law:

$$\frac{\rho}{\rho^*} = \left[ \frac{\beta - T}{\beta - T^*} \right]^\alpha$$

**POLY** Density varies according to the polynomial Equation:

$$\frac{\rho}{\rho^*} = 1 + \alpha(T^* - T) + \beta(T^* - T)^2 + \gamma(T^* - T)^3$$

**LINE** Density varies according to the linear relation:

$$\frac{\rho}{\rho^*} = 1 + \alpha(T^* - T) + \gamma(\beta - C)$$

**phase** The fluid phase for which the input is specified. See Section 3.6 for available options. By default the input pertains to the 1<sup>st</sup> phase of the fluid. This modifier is available only for the multi-phase versions of the **PORFLOW®** and **ANSWER®** Software Tools.

**ρ\*** Reference density for the fluid, ρ\*

**α** The exponent for the **POWER** law or the coefficient for the **POLYNOMIAL** and **LINEAR** relations. The default value is 0.2 for the **POWER** law equation and 0 otherwise

**β** The critical temperature for the of the **POWER** law, the coefficient for the **POLYNOMIAL** relation or the reference concentration for the **LINEAR** relation. The default value is 374.15 for **POWER** law and 0 otherwise

**γ** The coefficient in the **POLYNOMIAL** and **LINEAR** relations. The default value is 0.

#### APPLICABILITY

---

This mode of the command is available only for the **ANSWER®** and **PORFLOW®** Software Tools.

#### EXAMPLES

---

**DENSITY** by **POWER** law: rho\* = 1., Exponent = 0.25, Tc = 374.15 K

**DENSITY POLYNOMIAL** law: rho\* = 1000, a=1000., b=0.05, c=0., d=3.E-5

**DENSITY LINEAR** function: 997, Beta=1.0E-4

**DENSITY LINEAR** function 789, Beta=1.0E-4 for **SECOND** phase

**MODE 3:**      **Density from Gas Law**

**SYNTAX**      **DENS** {**GAS**}[**phase**] [**INCO**|**COMP**] [**REAL**] [**A**, **B**]

**GAS**            Density varies according to the gas law Equation:

$$\rho = \frac{p + p^*}{R_u (T + T_a) \sum_j \frac{m_j}{M_j}}$$

In this equation, p is the gas pressure, p\* is a reference datum pressure, R<sub>u</sub> is the universal gas constant, T is the temperature, T<sub>a</sub> is the base to convert temperature to absolute units, m<sub>j</sub> is the mass fraction of the j<sup>th</sup> component of the gas species and M<sub>j</sub> is the corresponding molecular weight.

**phase**            The fluid phase for which the input is specified. See Section 3.6 for available options. **By default the input pertains to the 1<sup>st</sup> phase of the fluid. This modifier is available only for the multi-phase versions of the PORFLOW® and ANSWER® Software Tools.**

**INCO**            The incompressible form of the gas law is used where the local pressure, p, is ignored in comparison to p\* in the gas law equation. This modifier is effective only if the **GAS** modifier is present. **This is the default option for all ACRi Software Tools except PORFLOW®.**

**COMP**            The compressible form of the gas law equation is used. The local pressure, p, is added to the reference pressure, p\*, in computing density. This modifier is effective only if the **GAS** modifier is present. **This is the default option for the PORFLOW® Software Tool**

**REAL**            The gas is assumed to be real and not ideal; the density is given by:

$$\rho = \frac{(p + p^* + A)(1 - B \rho)}{R_u (T + T_a) \sum_j \frac{m_j}{M_j}}$$

**A**                The pressure coefficient if the **REAL** modifier is specified.

**B**                The density coefficient if the **REAL** modifier is specified.

#### APPLICABILITY

---

This mode of the command is available only for the **ANSWER®** and **PORFLOW®** Software Tools.

#### EXAMPLES

---

**DENSITY** from **GAS** law: reference value = **0.960** kg/m<sup>3</sup>

**DENSITY** from **GAS** law reference value computed from other input

**DENSITY GAS** law in **INCOMPRESSIBLE** mode

**MODE 4:**      **Density of Brine as a Function of Salt Concentration**

**SYNTAX**      **DENS {SAND} [ $\rho^*$ ] [ $\alpha$ ] [ $\rho_{max}$ ] [ $\rho_{salt}$ ] [ $C_{S\_Sat}$ ]**

**SAND**      Density varies according to the exponential law correlation:

$$\frac{\rho}{\rho^*} = \min\{ \exp(\alpha C_s), \rho_{max} \}$$

Where  $C_s$  is the mass concentration of salt ( $0 \leq C_s \leq 1$ ) in the brine mixture which is computed from a transport equation.; other symbols are defined below.

$\rho^*$       Reference density for the fluid,  $\rho^*$ . The default value is 998.2.

$\alpha$       The exponent for the function; the default value is 0.6995.

$\rho_{max}$       The maximum density of the mixture fluid; the default value is 1200.9.

$\rho_{salt}$       The density of solid salt; the default value is 2165 (see Comments below).

$C_{S\_Sat}$       The maximum saturation mass fraction of salt in solution; the default value is 0.2643 (see Comments below).

#### APPLICABILITY

---

This mode of the command is available only for the **ANSWER®** and **PORFLOW®** Software Tools.

#### REMARKS

---

This relationship for the density of saline mixture of water is based upon the correlations developed for Sandia national Laboratory. (Anthony J. Russo, 1983. A User's Manual for the Salt Solution Mining Code, SANSMIC, UC-94e, Sandia National Laboratories, Albuquerque, NM 87185).

The input of  $\rho_{salt}$  and  $C_{S\_Sat}$  is not used for computing fluid density. However, if the solution mining algorithm is activated (see **BLOCK SALT** command), then these values are used to compute the amount of salt that dissolves from the salt blocks into the water.

#### EXAMPLES

---

**DENSITY SANDIA** function

**DENSITY SANDIA** function; rho\* = 998.2, alfa=0.6995, rho\_max=1209

**DENSITY SANDIA** function; rho\* = 998.2, alfa=0.6995, rho\_max=1209, rho\_salt=2165, Csat=0.2643

<b>MODE 5:</b>	<b>Density of Solid or Particle Material</b>
<b>SYNTAX</b>	<b>DENS {SOLI PART} {func[ξ]} {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub>   fname} [ALWA] [subrgn]</b>
<b>SOLI PART</b>	The solid material or particulate density is specified. The density is assumed to be for the dry particulate material.
<b>func</b>	One of the modifiers listed in Table 4.2.1 that denotes the functional form of the input. For this input, the function specifies the value of the density of the solid phase. <b>If no function is specified then the value is assumed to be constant.</b>
<b>ξ</b>	One of the independent variables listed in Table 4.2.2. <b>If no variable is specified, then the independent variable is assumed to be time.</b>
<b>N<sub>1</sub>,...,N<sub>n</sub></b>	The numerical constants and coefficients for the selected function. See Section 4.4 for more details. <b>There are no default values for this input.</b>
<b>fname</b>	The name of the file containing the numerical values <b>N<sub>2</sub></b> through <b>N<sub>n</sub></b> . This option is available only for selected functions. See Section 3.3 for additional information.
<b>ALWA</b>	By default the <b>DENS</b> command is implemented <b>immediately and only once</b> – as soon as the command is encountered. If this modifier is present then the command is executed immediately as well as <b>repeatedly</b> at the beginning of each time step (or iterative step in steady state mode) of the solution procedure.
<b>subrgn</b>	The subregion for which the input is specified. See Section 3.4. <b>If no subregion is specified, the entire computational domain is selected.</b>

#### APPLICABILITY

---

This command is currently available only for the **ANSWER®** and **PORFLOW®** Software Tools.

#### REMARKS

---

If specification for a fracture subdomain (**FRACTURE** command) is missing, then the value is assumed to be that for the host matrix at the first element of the subdomain.

#### EXAMPLES

---

**DENSITY** of **SOLID** is = 1  
**DENSITY** of **SOLID** is **2200** kg/m<sup>3</sup>

Generic examples for this command are given in Section 4.4. The command keyword (**DENS**) must replace the keyword used in these examples.

## 5.27 DIAGNOSTIC

**PURPOSE** To specify the options for diagnostic output of the values of variables, flux balances or convergence residual values.

**SYNTAX** **DIAG** [ $\Phi$ ] [**COORD|ELEM**] { $N_1, \dots, N_n$ } [ [**STEP|TIME**]=[ $N_{Freq}$ ] ] [**NOW|OFF**] [**OPEN**] [**HIGH**] [**MATR**] [*fname*]  $n=1$  for unstructured,  $n = 2$  for 2D and  $n = 3$  for 3D input mode

$\Phi$  One or more symbolic character strings. Each string denotes a desired diagnostic output for a corresponding variable. The valid symbols are:

- 1) The symbols listed in Table 2.8.1-3,
- 2) The symbols listed in Table 2.8.1-3 preceded by 'D',
- 3) The symbols listed in Table 2.7.1 preceded by a 'B'
- 4) The symbols listed in Table 2.7.1 preceded by an 'R',
- 5) The modifiers TIME or DTIM.

The prefix 'D' denotes difference (change) of the variable from the previous value, 'B' denotes the normalized flux balance disparity over the whole domain, and 'R' denotes the matrix residue. The diagnostic variables appear in the output in the same order as on the command. A maximum of 9 variables are written to the output file. The first 5 of these also appear on the screen (see **SCREEN** command) by default. All 9 variables are directed to the screen if the **WIDE** modifier is present on the **SCREEN** command.

The default diagnostic output depends on the equations being solved and the nature of the problem. For transient problems, time is always printed as the first value. Then up to 8 (for transient) or 9 (for steady state) other diagnostic indicators are selected. The order of selection is: values of the active field variables, flux balances and matrix residue for the variables for which the governing equations are solved.

**COORD** By default the monitoring location for variables is in terms of the grid indices (I,J,K) for a structured grid) or the element number (for unstructured grid). If this modifier is present, then  $N_1, \dots, N_n$  specify the (x, y, z) grid coordinates of the monitoring location.

**ELEM** If this modifier is present, then **N1** specifies the element number at which diagnostic output is produced. This modifier has no effect for unstructured grids.

$N_1, \dots, N_n$  The grid index location for the element for which the output is obtained. For structured grids, by default, the input is assumed to consist of the grid indices (I, J, K) in the 3D or (I, J) in the 2D mode. For unstructured grids, the default is the element number. If the modifier **COORD** is specified then the input is assumed to be the grid coordinates (x, y, z) in 3D or (x, y) in the 2D mode. If the modifier **ELEM** is specified then the input is that of the element number.

**STEP** The frequency of output,  $N_{Freq}$ , is in terms of simulation steps. This is the default option.

**TIME** The frequency of output,  $N_{Freq}$ , is in terms of time interval.

$N_{Freq}$  The frequency of diagnostic output in terms of time steps. A value of 0 is treated as identical to 1. The default value is 1. If present, this must be the last numerical value on the command.

**OFF** Diagnostic output is suppressed.

**NOW** Diagnostic output is produced immediately.

**OPEN** If this modifier is present, and the user specifies an element for diagnostic output (see  $N_1, \dots, N_n$ ) that is located in a blocked region (**BLOCK** command), then the diagnostic element will be moved to a neighboring open element.

**HIGH** If this modifier is present then additional diagnostics is provided about the allocation of memory for the storage of variables.

**MATR** If this modifier is present then additional information pertaining to the time taken by the matrix solver is generated at the end of the simulation.

**fname** By default a file with extended diagnostic output is generated with the same name as the Standard Output file but with the extension as “DGN”. For example, if the Standard Output file is “PROBLEM.OUT”, then the diagnostic file is named “PROBLEM.DGN”. If a file name is specified, then the diagnostic output is directed to the named file. This file contains the mass balance residuals and fluxes for each variable for which a transport equation is solved.

## REMARKS

---

The 1<sup>st</sup> column of the output for the diagnostic command contains a “Convergence Index”. This index is representative of a measure of the residue of matrix equations and is controlled by the **CONVERGENCE** command. The residue in turn is related to the flux balance disparity in the solution of the differential equations. The flux balance disparity is composed of four components: The cumulative changes in storage, inflow, outflow and decay (see **FLUX** command).

The convergence index is normalized by a used specified error tolerance (**CONVERGENCE REFERENCE** command). A value of less than unity for this index means that the residue for the reference matrix equation(s) is smaller than the specified tolerance.

If the equations are solved in the steady state mode, then steady state is assumed to be reached when the index reaches a value of less than unity. However, caution should be exercised and the solution should also be examined for changes in the values of the key variables. The step-to-step values (or changes) of selected variables at a diagnostic node are also printed by this command. At steady state, there should be no further changes in the values of variables. In practice, often a true steady state (except for simple flows) can only be approached asymptotically. In this case a steady state is assumed to prevail provided the changes in the values of the variables are negligible compared to some norm of the solution.

If the equations are solved in the transient mode, then the index is a measure of the accuracy of the solution at each intermediate time step.

## EXAMPLES

---

**DIAG**nostic node (4,8) ! 2D input mode; diagnostic printout every step  
**DIAG**nostic node (4,8,7) ! 3D input mode; diagnostic printout every step  
**DIAG**nostic output at element number 25 ! Unstructured mode  
**DIAG**nostic output at element number 25 every 10 steps; also file 'Diagnostics.eqn'  
**DIAG**nostics at (7,2,5) every 10 steps  
**DIAG**nostic output for U, P and T at node (7,2) print every 10 steps; also 'EQUATIONS.DGN'  
**DIAG**nostic output for TIME, U, BP and RT at node (7,2) print every 10 steps  
**DIAG**nostic output HIGH level with TIME, U, BP and RT at node (7,2) print every 10 steps  
**DIAG**nostic output for TIME, DTIME, P, DP & RP at node (7,2) every 10 steps  
**DIAG**nostic output: U, BP, RT and TIME in that order at (7,2) every 10 steps  
**DIAG**nostic output: TIME, U, BP and RT at node (7,2) print every 10 steps  
**DIAG**nostic output: TIME, U, BP & RT at node (7,2) print every 10 steps but every 20 steps for file output  
**DIAG**nostic output OFF  
**DIAG**nostic output NOW with MATRIX timing information

**5.28 DIFFUSION**

**PURPOSE** To specify diffusivity and the treatment of interface diffusion.

**REMARKS** \_\_\_\_\_

This command is identical to the **CONDUCTION** command.

**EXAMPLES** \_\_\_\_\_

See **CONDUCTION** command

**5.29 DISABLE**

**PURPOSE** To disable built-in default options.

**MODE 1:** **Disable Global Options**

**SYNTAX** **DISA** [**FLOW**] [**ENTH**] [**VELO**] [**DPDX**] [**DENS**] [**ALL**]

**FLOW** By default, the equations for the velocity components U, V, W and the density/pressure correction variable are always solved. This modifier may be used to disable flow calculations. Flow computation may also be disabled by appropriate specification of variables on the **SOLVE** command.

**ENTH** By default if enthalpy is read from an archive file by a **READ** command, then any user input of temperature is ignored. Any new specification of temperature by the user is ignored. However, in some cases, it may be desirable to overwrite the specified enthalpy value by computing a new value from the temperature field. If this modifier is present, then enthalpy at the start of the solution process is recomputed from the temperature field. This modifier is active only for the **ANSWER®** Software Tool.

**VELO** By default velocity, pressure and density are corrected based on the continuity equation. If this modifier is present, then only the pressure and density are corrected; velocity is retained at its value computed from the momentum equations. This modifier is active only for the **ANSWER®** Software Tool.

**DPDX** The pressure gradient terms in the momentum equations for **ANSWER®** software are set to zero. This is primarily a diagnostic tool to check the relative importance of various terms in the governing momentum equations. This modifier is active only for the **ANSWER®** Software Tool.

**DENS** If this modifier is present then the effect of density in the pressure equation is ignored everywhere except in the buoyancy term. This is equivalent to the so-called Boussinesq assumption. This modifier is active only for the **PORFLOW®** Software Tool.

**ALL** If this modifier is present along with the **DENSity** modifier, then the effect of density changes is ignored in all the heat and mass transport equations; that is the fluid density appearing in these equations is set equal to its reference value. This modifier is active only for the **PORFLOW®** Software Tool.

**EXAMPLES**

---

**DISA**ble FLOW

**DISA**ble FLOW computations

**DISA**ble ENTHalpy values read from the file for **ANSWER** software

**DISA**ble FLOW

**DISA**ble VELOcity correction for ANSWER

**DISA**ble DENSity variations in flow equation (Boussinesq assumption)

**DISA**ble DENSity variations in ALL equations for PORFLOW

**DISA**ble FLOW and DENSity variation in ALL equation

**MODE 2:**      **Disable Options for Specific Variables**

**SYNTAX**      **DISA { $\Phi$ } [CONV] [OFF]**

**$\Phi$**             **One or more symbols** that denote the dependent variables for which the specified feature is disabled. **There is no default value.**

**CONV**        In the absence of the **OFF** modifier, the convection term in the transport equation for the specified variable is set to zero. If the **OFF** modifier is present, then the convection term is reinstated.

**OFF**            Any previously disabled **CONV** option is reinstated.

#### **REMARKS**

---

This command may be used to explore the effect of each individual component of the transport equation for a selected variable. **At any given time at least one of the three (storage, convection and diffusion) terms in the transport equation must stay active, otherwise the transport equation has no solution.**

#### **EXAMPLES**

---

**DISA**ble CONVection term for T and C

**DISA**ble CONVection term for T is OFF (reinstate convection)

5.30 DISPERSIVITY

**PURPOSE** To specify the dispersion coefficients for dispersion of thermal and species equations. This command is effective only for the **PORFLOW®** Software Tool.

**MODE 1:** Uniform Dispersion Coefficients

**SYNTAX** DISP {Φ} [LONG] [α<sub>L</sub>] [TRAN] [α<sub>T</sub>] [subrgn]

**Φ** The symbol for the dependent variable for which the distribution coefficient is specified. Valid symbols are those for thermal and species equations listed in Table 2.7.1. There is no default value; a symbol must be specified.

**LONG** The longitudinal dispersion coefficient is specified.

**α<sub>L</sub>** The longitudinal dispersion coefficient. The default value is 0.

**TRAN** The transverse dispersion coefficient is specified.

**α<sub>T</sub>** The transverse dispersion coefficient. The default value is 0.

**subrgn** The subregion for which the input is specified. See Section 3.4. If no subregion is specified, the entire computational domain is selected.

**REMARKS**

The dispersion coefficients are used to compute the effective diffusivity tensor for a species due to diffusion and dispersion (see e.g. Freeze & Cherry, 1979). Typical formulation due to Scheidegger (1961) is:

$$\Gamma_{ij} = \phi_D \tau_{ij} \Gamma_M + \phi_E D_{ij} \quad , \quad (1)$$

$$D_{ij} = \alpha_T \delta_{ij} \underline{V} + (\alpha_L - \alpha_T) \frac{|V_i V_j|}{V} \quad , \quad (2)$$

Where:

- $\Gamma_{ij}$  is the effective diffusivity tensor,
- $\phi_D$  saturated diffusional porosity,
- $\tau_{ij}$  is the tortuosity,
- $\Gamma_M$  is the molecular diffusivity,
- $\phi_E$  saturated effective porosity,
- $D_{ij}$  is the hydrodynamic dispersion,
- $\delta_{ij}$  is the Kronecker delta,
- $V_i$  the particle velocity vector, and
- $V$  the magnitude of the particle velocity.

**EXAMPLES**

**DISPERSION** for **C 5** and **1** meters

**DISPERSION** for **C LONGitudenal** coef **5** ; **TRANsverse** value = **1** meters for **ID=SAND**

**DISPERSIVITY** for **C LONGitudenal** coef **5** meters **ID=SAND**

**DISPERSION** for **C TRANsverse** value = **1** meters for **ID=SAND**

**MODE 2: Dispersion Coefficient as a General Function**

**SYNTAX** **DISP** { $\Phi$ } {LONG|TRAN|xyz} {func[ $\xi$ ]} {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub>|fname} [subrgn]

- $\Phi$**  The symbol for the dependent variable for which the distribution coefficient is specified. Valid symbols are those for thermal and species equations listed in Table 2.7.1. **There is no default value; a symbol must be specified.**
- LONG** The longitudinal dispersion coefficient is specified.
- TRAN** The transverse dispersion coefficient is specified.
- xyz** One of the character strings: **XX**, **YY**, **ZZ**, **XY**, **YX**, **XZ**, **ZX**, **YZ** and **ZY**. It denotes the component of the dispersion tensor to which the input is applied. **The first 3 denote the diagonal components of the tensor while the others denote the off-diagonal components. Since the tensor is symmetric, XY and YX, XZ and ZX, and YZ and ZY are equivalent.**
- func** One of the modifiers listed in Table 4.2.1 that denotes the functional form of the input. The function specifies the value of the appropriate distribution coefficient for the corresponding  **$\Phi$**  variable. **In this mode a function must be specified.**
- $\xi$**  One of the independent variables listed in Table 4.2.2. **If no variable is specified, then the independent variable is assumed to be time.**
- N<sub>1</sub>,...,N<sub>n</sub>** The numerical constants and coefficients for the selected function. See Section 4.4 for more details. **There are no default values for this input.**
- fname** The name of the file containing the numerical values **N<sub>2</sub>** through **N<sub>n</sub>**. **This option is available only for selected functions. See Section 3.3 for additional information.**
- subrgn** The subregion for which the input is specified. See Section 3.4. **If no subregion is specified, the entire computational domain is selected.**

**REMARKS**

The implementation proceeds in the manner above except that if the **xyz** modifier is present, then the corresponding component of the **D<sub>ij</sub>** in Equation 2 above is directly set to the values computed from the specified function

**EXAMPLES**

Generic examples for this command are given in Section 4.4. The command keyword (**DISP**) must replace the keyword used in these examples.

**MODE 3:** Dispersion Coefficient for Vertically Stratified Anisotropic Soils

**SYNTAX** DISP {Φ} {STRA} {α<sub>LH</sub>, α<sub>TH</sub>, α<sub>LV</sub>, α<sub>TV</sub>} [subrgn]

- Φ** The symbol for the dependent variable for which the distribution coefficient is specified. Valid symbols are those for thermal and species equations listed in Table 2.7.1. **There is no default value; a symbol must be specified.**
- STRA** The input is being specified for a stratified anisotropic media where the dispersion contribution will be computed as given by Equation 3 below. This modifier is effective only for 3D problems. For 2D problems this formulation collapses to that given by Equation (2); in this case **α<sub>LH</sub>= α<sub>L</sub>; α<sub>TH</sub> = α<sub>T</sub>** and **α<sub>LV</sub>** and **α<sub>TV</sub>** are ignored.
- α<sub>LH</sub>** The horizontal longitudinal dispersion coefficient. **The default value is 0.**
- α<sub>TH</sub>** The horizontal transverse dispersion coefficient. **The default value is 0.**
- α<sub>LV</sub>** The vertical longitudinal dispersion coefficient. **The default value is 0.**
- α<sub>TV</sub>** The vertical transverse dispersion coefficient. **The default value is 0.**
- subrgn** The subregion for which the input is specified. See Section 3.4. **If no subregion is specified, the entire computational domain is selected.**

**REMARKS**

The general formulation for anisotropic porous media requires 5 empirical coefficients (see, e.g., Bear and Bachmat, 1986). The mode described here is based upon the work of Burnett and Frind (1987) and employs 4 constants. It is suitable for vertically stratified (such a layered) porous media. The individual components of the **D<sub>ij</sub>** tensor, assuming that the vertical direction is z, are then given by:

$$D_{xx} = \alpha_{LH} \frac{V_x^2}{V} + \alpha_{TH} \frac{V_y^2}{V} + \alpha_{TV} \frac{V_z^2}{V}, \tag{3a}$$

$$D_{yy} = \alpha_{TH} \frac{V_x^2}{V} + \alpha_{LH} \frac{V_y^2}{V} + \alpha_{TV} \frac{V_z^2}{V}, \tag{3b}$$

$$D_{zz} = \alpha_{TV} \frac{V_x^2}{V} + \alpha_{TV} \frac{V_y^2}{V} + \alpha_{LV} \frac{V_z^2}{V}, \tag{3c}$$

$$D_{xy} = D_{yx} = (\alpha_{LH} - \alpha_{TH}) \frac{V_x V_y}{V}, \tag{3d}$$

$$D_{xz} = D_{zx} = \{ 0.5 (\alpha_{LH} + \alpha_{LV}) - \alpha_{TV} \} \frac{V_x V_z}{V}, \tag{3e}$$

$$D_{yz} = D_{zy} = \{ 0.5 (\alpha_{LH} + \alpha_{LV}) - \alpha_{TV} \} \frac{V_y V_z}{V}, \tag{3f}$$

This equation becomes identical to Equation (2) when:

$$\alpha_{LH} = \alpha_{LV} = \alpha_L \text{ and } \alpha_{TH} = \alpha_{TV} = \alpha_T \tag{3g}$$

**EXAMPLES**

**DISPERSION** for **C STRATIFIED** option: **5 , 1, 3 and 0.5** meters for **ID=SAND**

## 5.31 DISTRIBUTION COEFFICIENT

**PURPOSE** To specify the distribution coefficient for the governing differential equations. This command is effective only for the **PORFLOW®** Software Tool.

**MODE 1:** Distribution Coefficient as a General Linear Function

**SYNTAX** **DIST** { $\Phi$ } {func[ $\xi$ ]} { $N_1$ } [ $N_2$  ...,  $N_n$ ] **fname** [**subrgn**] [**TOTA**]

$\Phi$  The symbol for the dependent variable for which the distribution coefficient is specified. Valid symbols are listed in Table 2.7.1. There is no default value; a symbol must be specified.

**func** One of the modifiers listed in Table 4.2.1 that denotes the functional form of the input. The function specifies the value of the appropriate distribution coefficient for the corresponding  $\Phi$  variable. If no function is specified then the value is assumed to be constant.

$\xi$  One of the independent variables listed in Table 4.2.2. If no variable is specified, then the independent variable is assumed to be time.

$N_1, \dots, N_n$  The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.

**fname** The name of the file containing the numerical values  $N_2$  through  $N_n$ . This option is available only for selected functions. See Section 3.3 for additional information.

**subrgn** The subregion for which the input is specified. See Section 3.4. If no subregion is specified, the entire computational domain is selected.

**TOTA** By default any chemical species is assumed to adsorb only in the wetted part to the solid:

$$Q_s = \rho_s S (1 - \Theta_E) C_s ,$$

where,  $Q_s$  is the mass of species C in solid,  $\rho_s$  is the density of solid, S is the fluid phase saturation,  $\Theta_E$  is the effective porosity and  $C_s$  is the concentration of species C in solid.

If linear equilibrium of the species in the primary (usually liquid) and the solid phases is assumed then, with  $k_d$  as the distribution coefficient and  $R_d$ , retardation coefficient:

$$C_s = \rho_s k_d C$$

$$R_d = 1 + \frac{1 - \Theta_E}{\Theta_E} k_d .$$

However if the **TOTA** modifier is present, then the species is assumed to be stored in the total solid and the above relations are replaced by:

$$Q_s = \rho_s (1 - \Theta_E) C_s ,$$

$$R_d = 1 + \frac{1 - \Theta_E}{S \Theta_E} k_d .$$

**REMARKS**

**PORFLOW®** also provides for non-linear relations between the solid and liquid concentrations as well as separate computations of partitioning between solid and liquid phases. These options are implemented by other modes of the **DISTRIBUTION** command.

The term “distribution coefficient” is normally applied only to the transport of chemical species. However, this command may be used to specify the “storage coefficient” for any of the governing transport equations.

**EXAMPLES**

Generic examples for this command are given in Section 4.4. The command keyword (**DIST**) must replace the keyword used in these examples.

**MODE 2: Distribution Coefficient as a Special Function****SYNTAX** **DIST** {Φ} {FREU|LANG} [subrgn] [N1] [N2] [TOTA]

**Φ** The symbol for the dependent variable for which the distribution coefficient is specified. Valid symbols are listed in Table 2.7.1. **There is no default value; a symbol must be specified.**

**FREU** The concentration of the species in the solid phase,  $C_s$ , is given by the Freundlich Isotherm:

$$C_s = \rho_s k_d C^n,$$

where  $\rho_s$  is the density of solid,  $k_d$  is a distribution coefficient,  $C$  is the concentration in the liquid phase and  $n$  is an empirical exponent.

**LANG** The concentration of the species in the solid phase,  $C_s$ , is given by the Langmuir equation

$$C_s = \rho_s C_{smax} \frac{C}{C_{lang} + C},$$

where  $C_{smax}$  is the maximum concentration per unit mass in the solid phase and  $C_{lang}$  is the Langmuir constant. It is seen that when  $C \ll C_{lang}$ , the relation become equivalent to a linear isotherm with  $k_d = C_{smax} / C_{lang}$ .

**subrgn** See Mode 1 specification.

**N1** The distribution coefficient,  $k_d$ , for the Freundlich Isotherm or the  $C_{smax}$  for the Langmuir equation.

**N2** The power exponent,  $n$ , for the Freundlich Isotherm or the Langmuir constant,  $C_{lang}$ , for the Langmuir equation.

**TOTA** See Mode 1 specification.

**COMMENT**

This command mode is available only with the **PORFLOW®** Simulation Tool for all the species equations and for the first phase pressure equation. It is **one of the means** available in **PORFLOW®** to incorporate non-linear isotherms with geochemistry. Non-linear kinetics can also be implemented by functional forms of the distribution or retardation coefficients (see Mode 1 of **DISTRIBUTION** and **RETARDATION** commands) and by the **REACTION** command.

For the first phase pressure equation in **PORFLOW®**, the variable the species concentration,  $C$ , is replaced by  $P$ , the pressure. The units of  $k_d$  and  $C_{smax}$  are such that the units of  $C_s$  are those of volumetric unit of adsorbed gas in the solid phase divided by the mass density of solid.  $C_{lang}$  has units of pressure,  $P$ .

**EXAMPLES**

**DISTR**tribute C as FREUndlich Isotherm with  $k_d=0.22$ ,  $n = 0.8$  for ID=CLAY

**DISTR**tribute C as LANGmuir equation with  $C\_smax=1.89$ ,  $C\_langmuir = 0.2$  for ID=CLAY

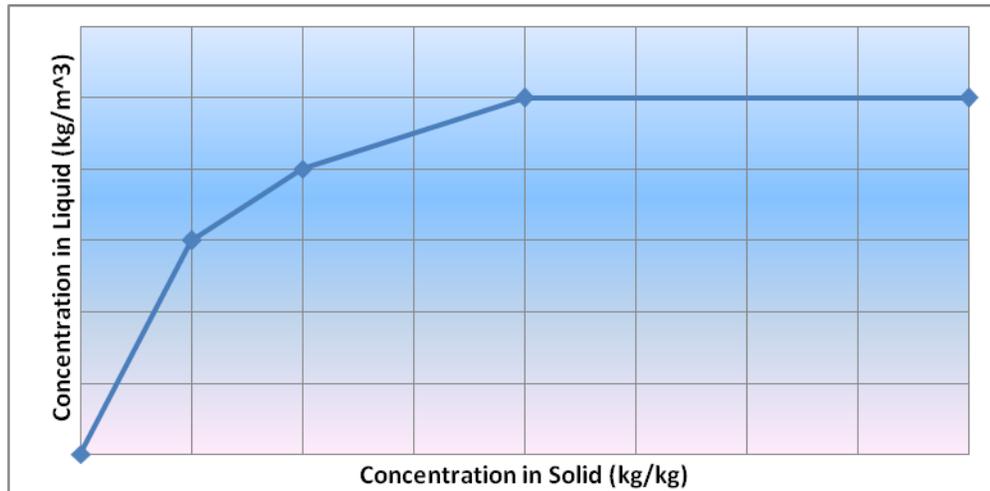
**MODE 3:** Distribution Coefficient from Fluid & Solid Phase Concentrations

**SYNTAX** **DIST** {**CONC**} {**Φ**} {**TABL**} [**subrgn**] {**N<sub>sets</sub>**} [**N<sub>1</sub>,..., N<sub>n</sub>**]**fname**] [**TOTA**]

**CONC** The distribution coefficient is computed from a specified table of concentration in the fluid phase as a function of the concentration in the solid. The total initial property can be specified by the **SET** command with **INVENTORY** modifier. If the initial inventory is not specified, then it is assumed that the initial amount in each element is equal to the initial concentration multiplied by the volume of the fluid in that element. The specified fluid concentration is then modified so that initial amount is distributed between the fluid and solid phases according to the specified tabular function.

**Φ** The symbol for the dependent variable for which the distribution coefficient is specified. Valid symbols are listed in Table 2.7.1. There is no default value; a symbol must be specified.

**TABL** The data is specified as a table of two values per data set. The first value is the concentration in the solid phase and the second is that in the liquid phase. A typical set of data is shown plotted below.



**subrgn** See Mode 1 specification.

**N<sub>sets</sub>** Number of sets of data in the table.

**N<sub>1</sub>, ..., N<sub>n</sub>** The values ( $2 * N_{sets}$ ) of solid and liquid concentration of species.

**fname** The name of the file containing the numerical values **N<sub>1</sub>** through **N<sub>n</sub>**.

**TOTA** See Mode 1 specification.

## COMMENT

---

This mode of the command is available only with the **PORFLOW®** Simulation Tool. It is one of the means available in **PORFLOW®** to incorporate non-linear isotherms with geochemistry. Non-linear kinetics can also be implemented by functional forms of the distribution or retardation coefficients (see Mode 1 of **DISTRIBUTION** and **STORAGE** commands) and by the **REACTION** command. More complex geochemistry can be implemented by separately solving the transport equations for the concentration of species in the liquid and solid phases.

## EXAMPLES

---

**DIST**ribute C as **CONC**entration **TABL**e of values: 6 sets

(0., 0.) (1., 0.) (1.0001, 1.) (2., 1) (2.0001, 0.5) (3., 0.2)

**DIST**ribute C2 in **CONC**entration mode for ID=ZN10 as **TABL**e of values: with 6 sets:

(0., 0.) (1., 0.) (1.0001, 1.) (2., 1) (2.0001, 0.5) (3., 0.2)

**DIST** C **CONC**entration as **TABL**e of 20 sets from file 'KD\_ALL' for ID=ZN10.

**MODE 4:** Distribution Coefficient from A Chain of Species or RadioNuclides

**SYNTAX** **DIST** {CHAIN} {Φ<sub>1</sub>, Φ<sub>2</sub>, .. Φ<sub>k</sub>} {TABL} [STEP|LINE|SECO|CUBI] [subrgn] {N<sub>sets</sub>} [N<sub>1</sub>,..., N<sub>n</sub> | fname] [TOTA]

**CHAIN**

The distribution coefficient is computed from a specified table of concentration in the fluid phase as a function of the concentration in the solid for the total mass of all species (or nuclides) in the chain. The total initial property for each species can be specified by the **SET** commands with **INVENTORY** modifier for each species. If the initial inventory is not specified, then it is assumed that the initial amount in each element is equal to the initial concentration multiplied by the volume of the fluid in that element. The specified fluid concentration is then modified so that initial amount is distributed between the fluid and solid phases according to the specified tabular function.

To implement this logic, the total solid concentration over all species is determined from the computed values of the species mass and liquid concentration as:

$$C_{solid} = \frac{\sum_k m_k - \sum_k \phi_k \theta S_f \delta V}{\rho_s (1-\theta) S_s \delta V}$$

Where, k denotes the species, m and Φ are, respectively, the computed mass and liquid concentration, θ is the porosity, S<sub>f</sub> is the saturation of the liquid phase, ρ<sub>s</sub> is the particle density of solid and δV is the volume of the element for which the computation is being performed. The quantity S<sub>s</sub> in the denominator is 1 if the **TOTAL** modifier is specified; otherwise it is equal to S<sub>f</sub>.

The effective total concentration in the liquid is then determined from the specified tabular function. The distribution coefficient, k<sub>d</sub> is determined from:

$$k_d = \frac{C_{solid}}{C_{liquid}}; \text{ if } C_{liquid} > 0, \quad k_d = \frac{\partial C_{solid}}{\partial C_{liquid}}; \text{ if } C_{liquid} = 0$$

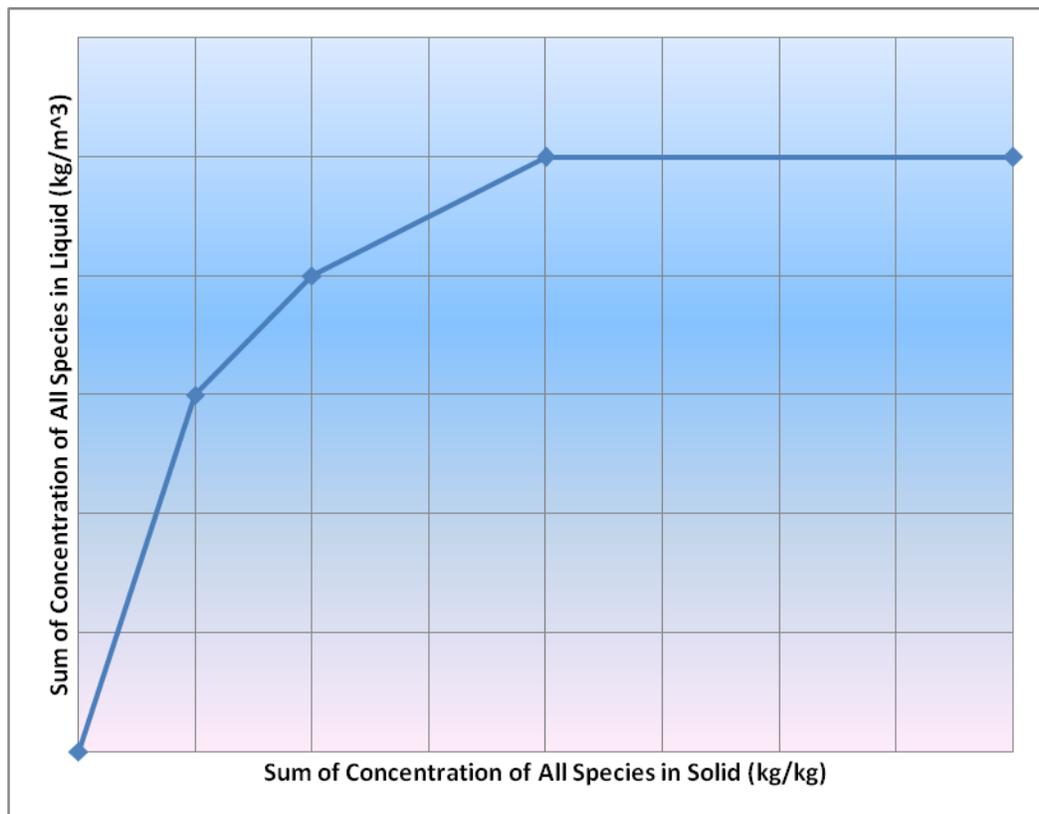
Finally a new value for each of the species in the chain is computed as:

$$\phi_k^{corrected} = \frac{m_k}{\theta S_f \delta V + \rho_s k_d (1-\theta) S_s \delta V}$$

Φ<sub>1</sub>, Φ<sub>2</sub>, .. Φ<sub>k</sub> The symbols for the dependent variable for which the distribution coefficient is specified. Valid symbols are listed in Table 2.7.1. There is no default value; a symbol must be specified

**TABL**

The data is specified as a table of two values per data set. The first value for each set is the sum of concentration in the solid phase and the second value is that in the liquid phase. A typical table is shown below.



- LINE** Linear interpolation is used for the tabular function. This is the default option.
- STEP** Table is interpreted in a step wise manner.
- SECO** Quadratic interpolation is used to evaluate the tabular function.
- CUBI** Cubic interpolation is used for the tabular function.
- subgrn** See Mode 1 specification.
- N<sub>sets</sub>** Number of sets of data in the table.
- N<sub>1</sub>, ..., N<sub>n</sub>** The values ( $2 * N_{sets}$ ) of solid and liquid concentration of species.
- fname** The name of the file containing the numerical values **N<sub>1</sub>** through **N<sub>n</sub>**.
- TOTA** See Mode 1 specification.

#### COMMENT

---

This mode of the command is available only with the **PORFLOW®** Simulation Tool. It is one of the means available in **PORFLOW®** to incorporate non-linear isotherms with geochemistry. Non-linear kinetics can also be implemented by functional forms of the distribution or retardation coefficients (see Mode 1 of **DISTRIBUTION** and **STORAGE** commands) and by the **REACTION** command. More complex geochemistry can be implemented by separately solving the transport equations for the concentration of species in the liquid and solid phases.

#### EXAMPLES

---

**DISTR**ibute for CHAIN of: U233, U234, U235, U236, U238 as TABLE of values: 6 sets  
 (0., 0.) (1., 0.), (1.0001, 1.), (2., 1) (2.0001, 0.5) (3., 0.2)

**MODE 5:** Distribution Coefficient for One Equation Concrete Degradation Model

**SYNTAX** **DIST** {CONCRETE} {CAL} {CAS} {TABL} [LINE|SECO|CUBI] {N<sub>set</sub>} {N<sub>1</sub> ..., N<sub>n</sub>|fname} {S<sub>si</sub>}, [subgrn]

**CONCRETE** This modifier triggers the solution of a CEA developed algorithm for degradation of concrete which is incorporated in the governing equation:

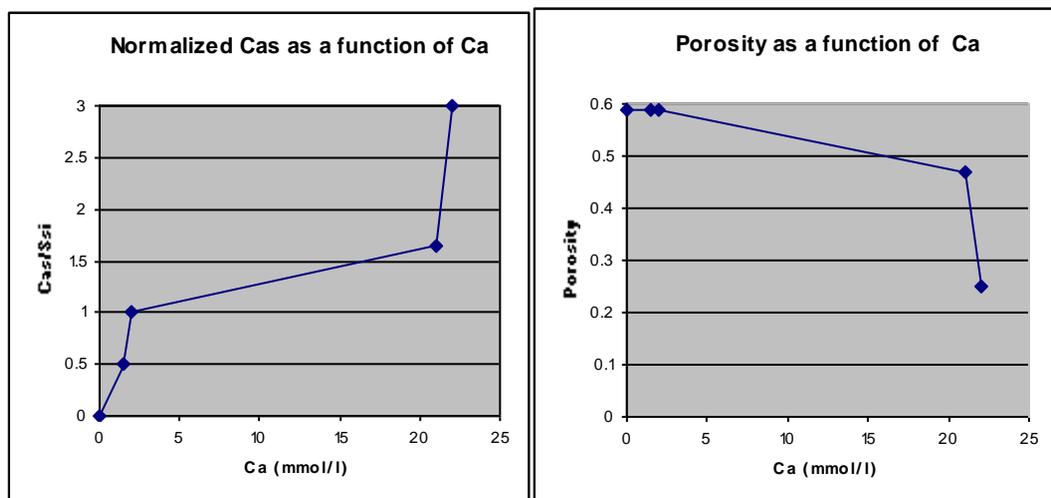
$$\frac{\partial}{\partial t} (\theta Ca + Ca_s) + \frac{\partial}{\partial x_i} (V_i Ca) = \frac{\partial}{\partial x_i} (\Gamma_{ca} \frac{\partial Ca}{\partial x_j}) + S_{ca} - Y_{ca}$$

Where Ca and Ca<sub>s</sub> are the concentrations of Calcium in the liquid and the solid phase respectively (The default symbolic name of Ca is CAL and that of Ca<sub>s</sub> is CAS). In this equation (x<sub>i</sub>,t) denote the space and time, V<sub>i</sub> is the Darcy velocity, θ is the porosity, Γ is the diffusivity, S and γ are the source and sink (reaction) terms and the subscript Ca denotes the Ca species. The diffusion coefficient for the governing concentration equations is typically an exponential function of porosity:

$$\Gamma = \Gamma_o \exp\left(\frac{\theta}{\theta_o}\right)$$

The diffusion coefficient must be specified by an appropriate DIFFUSION command. Tognazzi (1998) and Torrenti et al. (1999) quote Γ<sub>o</sub> = 2.3 x 10<sup>-13</sup> and θ<sub>o</sub> = 0.100503. Alternative diffusion coefficients may also be specified.

The degradation model is based on the assumption that decalcification of concrete occurs over a period of time due to multiple factors. The major parameters of degradation are leaching of Calcium (Ca) from the concrete components from the solid state into solution. During this process solid structure disintegrates with major changes in the volume of its constituents. The CEA model has been simplified so that both porosity and solid calcium concentration (Ca<sub>s</sub>) are tabulated functions of the calcium concentration in solution (Ca) (Please see References 1, 2 and 3 below for fuller details). The typical variation of Ca<sub>s</sub> and porosity with Ca is as shown in the figures below.



**CAL** The symbol for the dependent variable, Ca, for which the distribution and porosity functions are specified.

**CAS** The symbol for the solid phase concentration, Ca<sub>s</sub>.

**TABL** Specifies that a tabular function for the normalized Ca<sub>s</sub>/S<sub>si</sub> ratio and porosity as a function of Ca is specified. This modifier must be present; there is no default value.

**LINE** Linear interpolation is used for the tabular function. This is the default option.

**SECO** Quadratic interpolation is used for the tabular function.

**CUBI** Cubic interpolation is used for the tabular function.

- N<sub>Set</sub>** The number of sets of values that are specified for the characteristic function. There is no default value; a valid value (>0) must be specified.
- N<sub>1</sub>, ..., N<sub>n</sub>** **N<sub>Set</sub>** set of values for the characteristics. Each set consists of 3 values: Ca, normalized Ca<sub>s</sub>/S<sub>Si</sub> ratio, and porosity (θ), in that order.
- fname** See Mode 1 specification. The numerical values will be read from a file.
- S<sub>Si</sub>** Concentration of Silicium in the concrete paste that was used to normalize the specified Ca<sub>s</sub>/S<sub>Si</sub> ratio above. The specified values will be multiplied with this number to obtain the actual Ca<sub>s</sub> value. The default value is 4900.
- subrgn** See Mode 1 specification.

#### COMMENT

---

This mode of the command is available only with the **PORFLOW®** Simulation Tool. It is **one of the means** available in **PORFLOW®** to incorporate the concrete degradation. Other models can be incorporated by defining the phase variables (**ALLOCATE**) and appropriate distribution coefficients (see Mode 1 of **DISTRIBUTION** and **STORAGE** commands) and by the **REACTION** command.

#### EXAMPLES

---

**DISTRIBUTION** characteristics for CONCRETE given by 5 sets of values in a TABLE:

CAL	CAS	POR	
0.	0.	0.59	
1.5	0.5	0.59	
2.0	1.0	0.59	
21.0	1.65	0.47	
22.0	3.0	0.25	S <sub>Si</sub> = 4900

**MODE 6:** Distribution Coefficient for Two Equation Concrete Degradation Model

**SYNTAX** **DIST** {CONCRETE} {CARB} {CAL} {CAS} {TABL} [LINE|SECO|CUBI] {NSet} {N1 ..., Nn|fname} {Ssi}, [k], [Ca<sub>eq</sub>] [θ<sub>Lim</sub>] [V<sub>mol\_calcite</sub>] [subrgn] [INNE] [EXPL] [UNLI]

**CONCRETE** Together with the **CARB** modifier this triggers the solution of a two equation CEA developed algorithm for degradation of concrete which is incorporated is expressed in the governing equations:

$$\frac{\partial}{\partial t} (\theta Ca + Ca_s) + \frac{\partial}{\partial x_i} (V_i Ca) = \frac{\partial}{\partial x_i} (\Gamma_{ca} \frac{\partial Ca}{\partial x_j}) + S_{Ca} - Y_{Ca} - R_{Calcite}$$

$$\frac{\partial}{\partial t} (\theta CO_3) + \frac{\partial}{\partial x_i} (V_i CO_3 a) = \frac{\partial}{\partial x_i} (\Gamma_{co3} \frac{\partial CO_3}{\partial x_j}) + S_{CO3} - Y_{CO3} - R_{Calcite}$$

The reaction rate of Calcite formation,  $R_{Calcite}$  and the evolution of porosity are given by:

$$R_{Calcite} = k (\theta - \theta_{lim}) \max \{ \min [(Ca - Ca^*), (CO_3 - CO_3^*)], 0 \}$$

$$\frac{\partial \theta}{\partial t} = \frac{\partial W_{Ca}}{\partial t} - R_{Calcite} V_{mol\_Calcite}$$

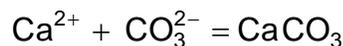
In these equations Ca and CO<sub>3</sub> are the concentrations of Calcium and the Carbonates in liquid and Ca<sub>s</sub> is the concentration of Calcium in solid phase (The default symbolic name of Ca, Ca<sub>s</sub> and CO<sub>3</sub> are CAL, CAS and CO3, respectively.). The constant k is the reaction rate of Calcite formation, θ<sub>lim</sub> is the minimum porosity, W<sub>Ca</sub> is the base porosity in the absence of any calcite formation and V<sub>mol\_Calcite</sub> is the molar volume of Calcite. Ca\* and CO<sub>3</sub>\* are the corresponding equilibrium concentration of the species. The rest o the notation is the same as in the previous mode of the command.

The diffusion coefficient for the governing concentration equations is typically an exponential function of porosity:

$$\Gamma = \Gamma_o \exp\left(\frac{\theta}{\theta_o}\right)$$

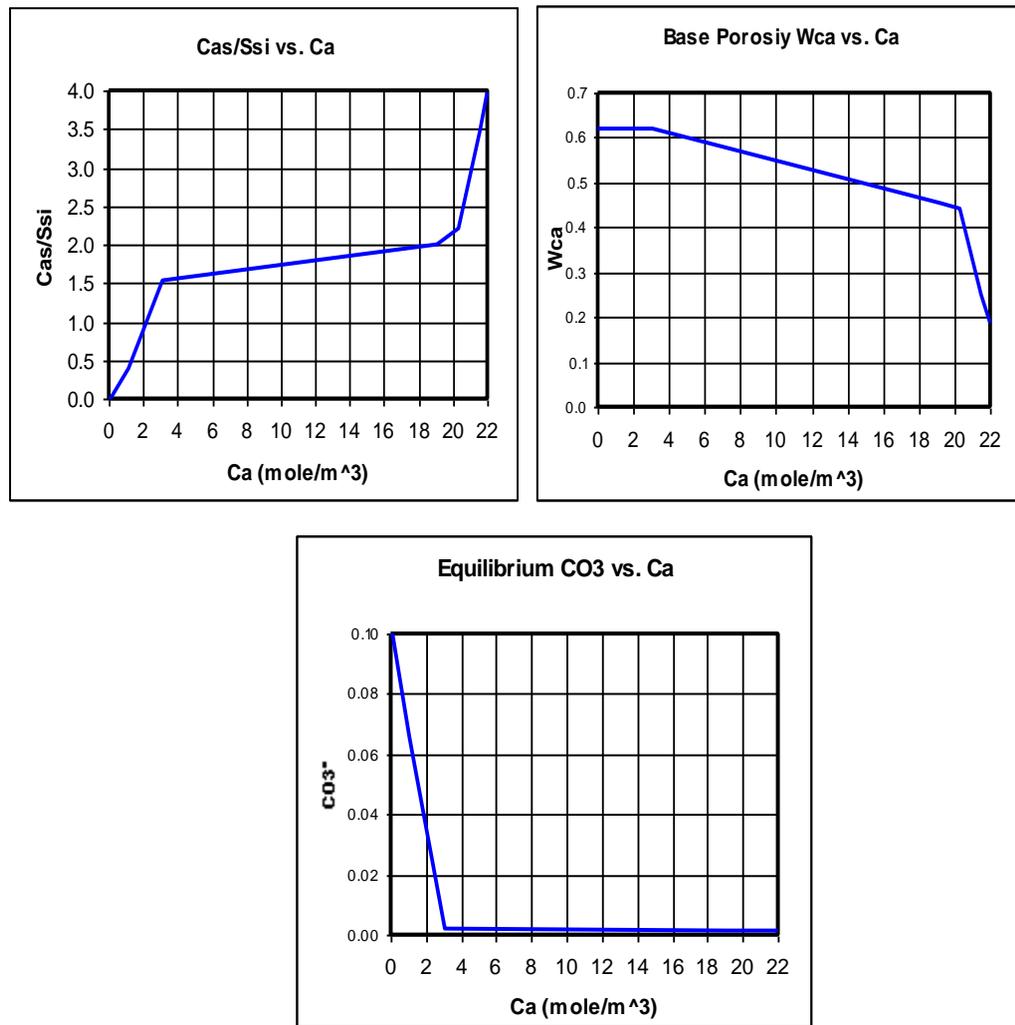
The diffusion coefficient must be specified by an appropriate **DIFFUSION** command. Tognazzi (1998) and Torrenti et al. (1999) quote  $\Gamma_o = 2.3 \times 10^{-13}$  and  $\theta_o = 0.100503$ . Alternative diffusion coefficients may also be specified.

This degradation model is based on the assumption that the major mechanism of degradation is leaching of Calcium (Ca) from the concrete components from the solid state into solution and that the primary calcification reaction is given by:



Further details and theoretical background of this model is described by Bary and Bregeon (2003) and, among others, is based on the work by Adenot and Buil (1992), Mainguy et al. (2000) and Richet et al. (2004).

The CEA model has been simplified so that solid calcium concentration, Ca<sub>s</sub>, the base porosity, W<sub>Ca</sub>, and the equilibrium CO<sub>3</sub> concentration, CO<sub>3</sub>\*, are tabulated functions of the calcium concentration in solution, Ca. The typical variations of these are shown in the figures below.



- CARB** The modifier to trigger the 2 equation concrete degradation model described above.
- CAL** The symbol for the dependent variable, Ca, for which the distribution and porosity functions are specified.
- CAS** The symbol for the solid phase concentration, Ca<sub>s</sub>.
- TABL** Specifies that a tabular function for the normalized Ca<sub>s</sub>/S<sub>Si</sub> ratio and porosity as a function of Ca is specified. **This modifier must be present; there is no default value.**
- LINE** Linear interpolation is used for the tabular function. **This is the default option.**
- SECO** Quadratic interpolation is used for the tabular function.
- CUBI** Cubic interpolation is used for the tabular function.
- N<sub>Set</sub>** The number of sets of values that are specified for the characteristic function. **There is no default value; a valid value (>0) must be specified.**
- N<sub>1</sub>, ..., N<sub>n</sub>** **N<sub>Set</sub>** set of values for the characteristics. Each set consists of 4 values: Ca, normalized Ca<sub>s</sub>/S<sub>Si</sub> ratio, W<sub>Ca</sub> porosity and CO<sub>3</sub><sup>\*</sup>, in that order.
- fname** See Mode 1 specification. The numerical values will be read from a file.
- S<sub>Si</sub>** Concentration of Silicon in the concrete paste that was used to normalize the specified Ca<sub>s</sub>/S<sub>Si</sub> ratio above. The specified values will be multiplied with this number to obtain the actual Ca<sub>s</sub> value. **The default value is 4900.**
- k** Reaction rate for the Ca and CO<sub>3</sub> reaction. Typical values are in units of [s<sup>-1</sup>] but any consistent units may be used. **The default value is 1**
- Ca<sub>eq</sub>** The equilibrium value of Ca. **The default value is 0.11**

- $\theta_{Lim}$**  The minimum value of porosity. The default value is 0.01
- $V_{mol\_calcite}$**  The molar volume of Calcite. The default value is  $3.69 \times 10^{-5}$  [m<sup>3</sup>/mol]
- subrgn** See Mode 1 specification.
- INNE** By default the carbonation reaction at any time step is computed before the species equations are solved. Once the new values of the species are computed at any time step, the reaction is not updated unless **CONVERGENCE COUPLED SPECIES** command is present. If the **INNER** modifier is present then the reaction is updated in the inner loop for the species which is controlled by the **CONVERGENCE** command. See **REMARKS** below.
- EXPL** By default at every time step, the new value of the **CO3** and **CAL** are computed on the assumptions that the reaction term can be treated implicitly. If the **EXPL** modifier is present, then the reaction term is treated explicitly. **This option is not recommended unless the reaction rate is very slow.**
- UNLI** By default it is assumed that the total reaction at any time step is limited by the amount of species present and the rate of diffusion and advection that can take place. If the **UNLI** modifier is present, then the reaction term is unbounded. **This option is not recommended unless the reaction rate is very slow.**

## REMARKS

---

This mode of the command is available only with the **PORFLOW®** Simulation Tool. It is one of the means available in **PORFLOW®** to incorporate concrete degradation. Other models can be incorporated by defining the phase variables (**ALLOCATE**) and appropriate distribution coefficients (see Mode 1 of **DISTRIBUTION** and **STORAGE** commands) and by the **REACTION** command.

Unless the reaction rate of **CO3** and **CAL** is very slow, one must use one of two different strategies for convergence. Because of the coupling due to the reaction rate, it may be necessary to perform an outer iteration on the species at every time step by specifying the **CONVERGENCE COUPLED SPECIES** command with appropriate number of outmost iterations. In most situations, it should be sufficient to perform 2 to 3 iterations on the coupled system. It is also important to ensure that each species is fully converged at each step. Because the reaction term in the equation is changing, it is important that a **CONVERGENCE** command be specified for each of **CO3** and **CAL** with adequate number of iterations. In most situations, it should be sufficient to perform 2 to 5 iterations on each species. Though both the coupled as well as the inner iteration may be required for stiff system of equations, it is generally more economical and sufficient to perform either the outer coupled iterations or the inner iterations.

## EXAMPLES

---

**DISTRIBUTION CONCRETE CARBONATION** given by 8 sets of values in a TABLE:

CAL	CAS	POR	CO3*
0.0	0.0000	0.6210	0.100
0.11	0.0000	0.6210	0.100
1.10	0.4000	0.6210	0.066
3.09	1.5200	0.6210	0.002
19.08	2.0100	0.4539	0.001
20.31	2.2100	0.4410	0.001
21.54	3.4900	0.2480	0.001
22.00	3.9687	0.1862	0.001
SSI	= 4900		
Reaction Rate	= 1		
Equilibrium Ca	= 0.11		
Limiting Porosity	= 0.01		
Calcite Molar Volume:	= 3.69E-5 !m^3/mol		

**MODE 7:**      **Distribution Coefficient for Gas Phase**

**SYNTAX**      **DIST {GAS} {Φ} {func[ξ]} {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub>} [fname] [subrgn]**

**GAS**      The species is partitioned in equilibrium between the liquid and gas phases. The partitioning concentration, C<sub>g</sub>, in the gas phase is given by:

$$C_g = \rho_g k_g C$$

Where  $\rho_g$  is the gas density,  $k_g$  is a distribution coefficient and C is the concentration in the liquid phase. This  $k_g$  is akin to the more usual Henry's Law constant except that the coefficient is defined in a manner that is consistent with that used for the solid phase. **In this mode the density of the gas phase must be separately defined by user input.**

**Φ**      The symbol for the dependent variable for which the distribution coefficient for the gas phase is specified. Valid symbols are listed in Table 2.7.1. **There is no default value; a symbol must be specified.**

**func**      One of the modifiers listed in Table 4.2.1 that denotes the functional form of the input. The function specifies the value of the appropriate distribution coefficient for the corresponding **Φ** variable. **If no function is specified then the value is assumed to be constant.**

**ξ**      One of the independent variables listed in Table 4.2.2. **If no variable is specified, then the independent variable is assumed to be time.**

**N<sub>1</sub>,...,N<sub>n</sub>**      The numerical constants and coefficients for the selected function. See Section 4.4 for more details. **There are no default values for this input.**

**fname**      The name of the file containing the numerical values **N<sub>2</sub>** through **N<sub>n</sub>**. **This option is available only for selected functions. See Section 3.3 for additional information.**

**subrgn**      See Mode 1 specification.

#### COMMENT

---

This mode of the command is available only with the **PORFLOW®** Simulation Tool. It is **one of the means** available in **PORFLOW®** to incorporate the Henry's Law for equilibrium phase partitioning. Non-linear equilibrium relations can also be implemented by functional forms of the distribution or retardation coefficients (see Mode 1 of **DISTRIBUTION** and **STORAGE** commands) and by the **REACTION** command.

#### EXAMPLES

---

Generic examples for this command are given in Section 4.4. The command keyword (**DIST**) must replace the keyword used in these examples and the modifier **GAS** must be specified.

**MODE 8:**      **Disable Storage or Transient Term Completely**

**SYNTAX**      **DIST { $\Phi$ } {OFF}**

**$\Phi$**               The symbol for the dependent variable for which the distribution coefficient is specified. Valid symbols are listed in Table 2.7.1. **There is no default value; a symbol must be specified.**

**OFF**             The storage term for the variable for whole of the flow field is completely omitted. In essence, the transient term in the governing differential equation is set to zero. A new Mode 1 specification may follow if storage term is to be reactivated.

#### **EXAMPLES**

---

**DIST** for C variable is now OFF

## 5.32 EBU

**PURPOSE** Specify the mixing limited rate and parameters for a given reaction using the eddy breakup model. This command is effective only for the ANSWER® Software Tool.

**MODE 1:** EBU Reaction Variables and Parameters

**SYNTAX** EBU {idreac} {Φ} [HARM] [C<sub>EBU</sub>, α, β] [γ]

**idreac** Character string identifying the **idreac** reaction on the **REACTION** command that is limited by the eddy breakup (EBU) limit. To activate this limit, the modifier EBU must also appear in the corresponding **REACTION** command. The EBU reaction rate is computed from:

$$R_{EBU} = C_{EBU} \rho \frac{\varepsilon}{k} \Phi_{ebu}^{rms}$$

$$\Phi_{ebu}^{rms} = \min(\Phi_1, \alpha \Phi_2, \beta \Phi_3).$$

If  $R_{KIN}$  is the kinetic reaction rate (see **REACTION** command), then the actual EBU-limited reaction rate for the **idreac** is defined by one of the following two options:

$$R_{Actual} = \min(R_{KIN}, R_{EBU}), \text{ or}$$

$$R_{Actual} = \frac{R_{KIN} R_{EBU}}{\gamma R_{KIN} + (1 - \gamma) R_{EBU}}.$$

The first equation is used for the default mode of the EBU-limiter whereas the second equation is used if the **HARMONIC** modifier is specified.

**Φ** One or more of the symbols in Table 2.8.1-3 which identify the EBU limiting species ( $\Phi_1$ ,  $\Phi_2$ , or  $\Phi_3$ ). If no symbol is specified then the species participating in the corresponding **REACTION** command are used as the limiting species.

**HARM** By default, the actual reaction rate is computed as the smaller of the kinetic and EBU reactions. If this modifier is present then the harmonic limiter is used.

**C<sub>EBU</sub>** The empirical constant,  $C_{EBU}$ , in the  $R_{EBU}$  above. The default value is 3.

**α, β** The constants,  $\alpha$  and  $\beta$ , in the EBU relation given above. The default value is unity.

**γ** The constant  $\gamma$  in the harmonic limiter as described below. It is the 2<sup>nd</sup>, 3<sup>rd</sup> or 4<sup>th</sup> numerical value on the command depending on whether 1, 2 or 3 symbols define the EBU model. The default value is 0.5.

**EXAMPLES**

EBU for R1 function of FU with constant = 3.0

EBU for R2 function of CH and O2, values = 3.0 and 0.3

EBU for R2 with CH and O2, 3.0, 0.3 with HARMonic limiter 0.75

EBU for R2 with FU CH and O2, 3.0, 0.5, 0.3 with HARMonic limiter 0.75

**MODE 2:** Global EBU Reaction Constants

**SYNTAX** EBU [C<sub>EBU</sub>] [I<sub>EBU</sub>]

**C<sub>EBU</sub>** The global empirical constant, C<sub>EBU</sub>, for the eddy breakup relations. This value is used if a value is not explicitly specified with a Mode 1 EBU command for a specific idreac reaction. The default value is 3.

**I<sub>EBU</sub>** The first step of the solution procedure at which the EBU limit is invoked. Often at the initial stages of a solution process, the flow and turbulence fields are not well developed. In this case, it is preferable to use the kinetic reaction rate (the maximum possible) during the initial stage. The default value is 10.

### EXAMPLES

---

EBU limit global rate constant = 3.2

EBU limit global rate constant is 3.2. Start at step number 20.

**5.33 ELSE**

**PURPOSE** To define an IF construct to control which, if any, of one or two blocks of user input statements are executed.

**SYNTAX** **ELSE**

**REMARKS** \_\_\_\_\_

This command is part of the **IF-ELSE-ENDIF** construct; please see the **IF** command.

**5.34 END****PURPOSE** To signify the end of a problem.**SYNTAX** **END****REMARKS**

---

This command signifies the end of a problem. The input for a new problem may be continued with a new problem specification after this command. For each problem, this command must be employed as the last command. Failure to do so may cause a loss of some or all of the data and output files, depending on the host operating system.

**EXAMPLES**

---

**END****END** of problem number 1

**5.35   ENDIF**

**PURPOSE**    To define an **IF** construct to control which, if any, of one or two blocks of user input statements are executed.

**SYNTAX**       **ENDI**

**REMARKS** \_\_\_\_\_

This command is part of the **IF-ELSE-ENDIF** construct; please see the **IF** command

## 5.36 FFT

**PURPOSE** To obtain Fast Fourier Transform of Given Discrete Data

**SYNTAX** **FFT** { $\Phi$ } [ $\xi$ ] [ $N_v$ ] [**INVE**]

$\Phi$  The symbol for the variable that contains the data to be analyzed by the FFT.

$\xi$  The symbol for the variable that will contain the output from the FFT analysis. If no symbol is specified, then the transform will be returned in the  $\Phi$  variable.

$N_v$  The number of values in the  $\Phi$  vector. The current algorithm is restricted to data that has  $2^N$  components where N is an integer. **If the specified number is larger, then the first  $2^N$  items will be included in the analysis.** If no value is specified then it is assumed that the vector length is equal to the total number of nodes specified by the **GRID** command.

**INVE** The inverse transform is computed.

**REMARKS**

---

This command computes the Fourier Transform of a set of N real data points and replaces data in the location of the  $\xi$  (or  $\Phi$  if  $\xi$  is not specified) variable by the positive frequency half of its complex Fourier Transform. The real-valued first and last components of the complex transform are returned in the 1<sup>st</sup> and 2<sup>nd</sup> locations.

**EXAMPLES**

---

**FFT** analyses of variable **T**

**FFT** analyses of variable **T** put output in variable **V\_FFT**

**FFT** analyses of variable **T** first **1024** values; output in variable **V\_FFT**

## 5.36 FILE

**PURPOSE** To open and close input/output devices.

**SYNTAX** **FILE** [**CLOS**|**OPEN**] [**fname**] [**attribute**] [**filetype**|**NUNIT**]

**CLOS** The specified file unit or I/O device is closed.

**OPEN** The specified file unit or I/O device is opened. If the same file unit was previously open, then it is closed and the new unit with the specified name is opened.

**fname** The name of the file to be opened or closed. See Section 3.3 for additional information. **If fname is specified, then the named file, if connected to a unit, will be closed.** If **filetype** or **NUNIT** refer to a different file unit, then that unit, if open, will also be closed. If the **OPEN** modifier is present, then a new file with the specified name will be opened and connected to the unit specified by **filetype** or **NUNIT**.

**attribute** The attributes of the file to be opened. This input is ignored if **CLOSE** modifier is present.

attribute	Interpretation
<b>FORM</b>	File is opened in the FORMATTED mode. <i>This is the default option.</i>
<b>UNFO</b>	File is opened in the UNFORMATTED mode
<b>UNKN</b>	File status is defined as UNKNOWN. <i>This is the default option.</i>
<b>NEW</b>	File status is defined as NEW. An error will occur if the file already exists.
<b>OLD</b>	File status is defined as OLD. An error will occur if the file does not exists..
<b>BOTH</b>	File is available for both READ & WRITE operations. <i>This is the default option.</i>
<b>READ</b>	File is available only for READ operations.
<b>WRIT</b>	File is available only for WRITE operations.

**filetype** **One** of the modifiers below that identify the file to be opened or closed.

filetype	Interpretation
<b>DEBU</b>	Debug output file ( <b>DEBUG</b> ), Unit 17, is selected for the operation.
<b>FLUX</b>	Flux output file ( <b>FLUX</b> ), Unit 14, is selected for the operation.
<b>HIST</b>	Time History file ( <b>HISTORY</b> ), Unit 13, is selected for the operation.
<b>SAVE</b>	The default Archive file ( <b>SAVE</b> ), Unit 11, is selected for the operation.
<b>TRAC</b>	Particle track file ( <b>TRACK</b> ), Unit 18, is selected for the operation.

**NUNIT** The file or I/O device unit number for the file for which the operation is performed. **The unit number is ignored if one of the filetype modifiers is present.**

**REMARKS**

In general FORTRAN language does not differentiate between lower and upper case characters. However that is not true of many operating systems (O/S). It is in fact the O/S that performs all file handling. O/S peculiarities have been reported where, for example, there was no case distinction for read/write operations but the O/S distinguished between upper and lower case for file closing. **It is therefore strongly recommended that the user should be consistent** in using the same case characters if a previous file is referred to during file operations.

**EXAMPLES**

**FILE OPEN** 'mynewfile.now' on 37  
**FILE OPEN OLD** file 'mydatafile.now' on unit 37 in UNFormatted READ only mode  
**FILE OPEN NEW SAVE** file 'mynewsavefile.now' on in FORMatted WRITe only mode  
**FILE CLOSE** SAVE HISTOrY file now  
**FILE CLOSE** file by name 'OLDFILE.TMP'

## 5.37 FIX

**PURPOSE** To fix the values or the matrix coefficients of selected variables for a subregion within the domain of calculation.

**MODE 1:** Fix Variable Values

**SYNTAX** FIX [ $\Phi$  | ALL SPEC] [ $V_{fix}$ ] [subrgn]

$\Phi$  One or more of symbols that denote the dependent variables for which the values are fixed for the selected subregion. Up to 100 symbols may be specified on one command. The valid symbols are listed in Table 2.7.1. If no symbol is specified, the values are fixed for all variables.

**ALL** If **ALL** and the **SPEC** modifiers are present together, then the input is applied to ALL species variables. That means the velocity and pressure (momentum) and thermal energy equation are excluded from its effect.

**SPEC** See above.

$V_{fix}$  The value that is to be applied to all variables that are to be fixed. If the value specified is for the Temperature (T) of Turbulence Length Scale (L), then no other variables can be present on the command. By default the variable values are fixed at their current state.

$V_{fix}$  subrgn The subregion for which the input is specified. If no subregion is specified, the entire computational domain is selected. See Section 3.4.

**REMARKS**

This command can only be used for those variables for which a transport equation is solved; that is the variables listed in Table 2.7.1. During the computation process the matrix is manipulated so that the current values stay in force and the transported equation is essentially deactivated for the specified region. The existing values may have been computed previously or set by a **SET** or other commands. A uniform value

This command can be used only to fix the values of a variable inside the domain of computation. The values at the domain boundary cannot be fixed by this command. The boundary values can be fixed by the **BOUNDARY** command. Multiple commands may be used to fix values in an arbitrary manner. The **INITIAL**, **READ** or **SET** commands may be used to define the variable values. Once specified, these values remain unchanged unless modified by a subsequent **INITIAL**, **READ** or **SET** command.

It should be noted that this command may creates internal sources and sinks since the values that are set may not be the values that satisfy the governing transport equation. The difference between the two essentially represents a source or sink.

**EXAMPLES**

**FIX** T for active subregion at previously defined value  
**FIX** P for subregion with ID=FREEstream  
**FIX** P, T and C for subregion identified as ID = ID01  
**FIX** values for all variables in ID=BLOCK

**MODE 2:** Fixed Matrix Coefficients for the Variable

**SYNTAX** **FIX** [ $\Phi$ ] [subrgn] { $N_1, \dots, N_n$ };  $n=6$  for 2D and 8 for 3D input mode

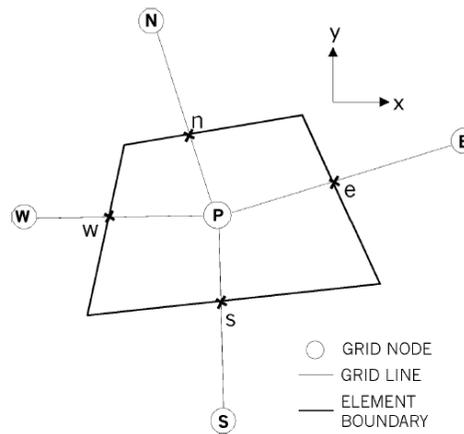
$\Phi$  A symbol that denotes the dependent variable for which input is specified. The valid symbols are listed in Table 2.7.1. There is no default value.

subrgn The subregion for which the input is specified. If no subregion is specified, the entire computational domain is selected. See Section 3.4.

$N_1, \dots, N_n$  The values of the matrix coefficients for the variable. For 2D simulation, with reference to the below figure, the actual value is computed according to the implicit formula:

$$\Phi_P^n = N1 + N2 \Phi_P^o + N3 \Phi_W^n + N4 \Phi_E^n + N5 \Phi_S^n + N6 \Phi_N^n$$

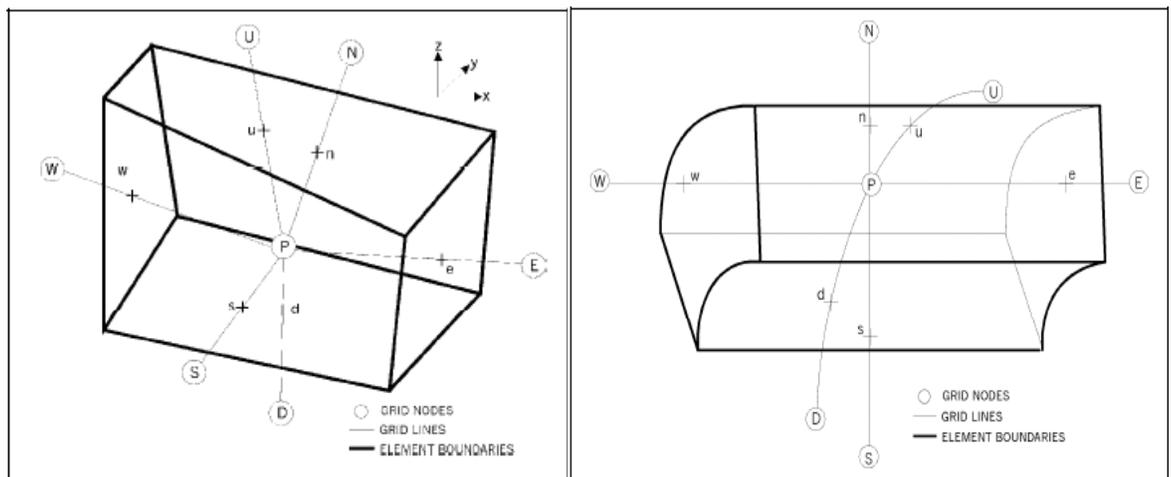
where superscript 'o' denotes the existing value and 'n' denotes the new computed value. The subscripts P, W, E, S and N denote the node being computed and the nearest neighboring nodes in the X-, X+, Y- and Y+ directions, respectively.



For 3D simulation, the formula is:

$$\Phi_P^n = N1 + N2 \Phi_P^o + N3 \Phi_W^n + N4 \Phi_E^n + N5 \Phi_S^n + N6 \Phi_N^n + N7 \Phi_D^n + N8 \Phi_U^n$$

where D and U denote the neighboring nodes in the Z- and Z+ directions, respectively, as shown in the figures below.



**MODE 3:**      **Disable Previously Specified Fixed Commands**

**SYNTAX**      **FIX**    {**OFF**} [**Φ**] [**subrgn**]

**OFF**            Previously specified **FIX** commands for **Φ**, for the identified subregion, are deactivated. A new specification for this subregion may follow. **This command can only be used for previous **FIX** commands that appeared with no **Φ** symbol or only a single **Φ** symbol.**

**Φ**                A symbol that denotes the dependent variable for which the input is specified. The valid symbols are listed in Table 2.7.1. **There is no default value.**

**subrgn**        The subregion for which the input is specified. **If no subregion is specified, the entire computational domain is selected.** See Section 3.4.

#### **EXAMPLES**

---

**FIX** T in ID=FIXD OFF

**FIX** T & P OFF for ID=FIX1

**FIX** all OFF in ID=FIXAll

## 5.38 FLOW

**PURPOSE** To specify the flow rate and other dependent variables at a boundary.

**MODE 1:** Flow Injection or Withdrawal with Fixed Variable Values

**SYNTAX** FLOW [TOTA [option] [func[ξ]] {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub>|fname} [Φ<sub>1</sub> =N<sub>n+1</sub>, ..., Φ<sub>m</sub> = N<sub>m</sub>] [N<sub>m+1</sub>, N<sub>k</sub>] [ρ<sub>B</sub>] [subrgn] {dir}

**TOTA** By default, the amount of flow specified, or computed from **func** (ξ), is applied to each element of the **subrgn**. If this modifier is present then the amount is assumed to be the total amount over whole of the **subrgn**. In this case, the amount is distributed equally to all the elements of the **subrgn** unless the **VOLUme** or **AREA** modifiers are present.

**option** Options selected for implementation of the source.

option	INTERPRETATION
<b>VOLU</b>	In the absence of the <b>TOTAL</b> modifier, the flow for each element is computed as: $Q = q \delta V$ . Here <b>q</b> is the amount specified by the user and $\delta V$ is the volume of the element. The <b>q</b> , in turn, is computed from <b>func</b> (ξ) and <b>N1</b> through <b>Nn</b> . If the <b>TOTAL</b> modifier is present, the amount for each element is computed as: $Q = q \delta V / V$ , where <b>V</b> is the volume of the total <b>subrgn</b> .
<b>AREA</b>	In the absence of the <b>TOTAL</b> modifier, the source for each element is computed as: $Q = q \delta A$ , where $\delta A$ is the area of the element boundary indicated by <b>dir</b> . If the <b>TOTAL</b> modifier is present, the source for each element is computed as: $Q = q \delta A / A$ , where <b>A</b> is the total area of the <b>subrgn</b> in the <b>dir</b> direction.
<b>INTE</b>	By default, if <b>dir</b> points to a wall, then any special treatment for that wall (such as wall function for turbulent flow) is deactivated and the diffusive flux at the wall is set to zero if the <b>INTERNAL</b> modifier is present then the wall treatment and wall diffusive flux are retained.
<b>NORM</b>	In the absence of the <b>TOTAL</b> modifier, the source, <b>Q</b> , is computed as: $Q = q \sum_i A_i \cdot V_i$ where $A_i$ is the $i^{\text{th}}$ direction component of the element boundary area specified by <b>dir</b> . $V_i$ are the values specified by <b>Nn+1</b> through <b>Nk</b> (2 for 2D, and 3 for 3D). In the presence of the <b>TOTAL</b> modifier, <b>Q</b> is computed in a manner identical to that for the <b>AREA</b> modifier.
<b>DENS</b>	The computed source, <b>Q</b> , is further multiplied by density. The density may be specified as the last value, <b>ρ<sub>B</sub></b> , on the command. The boundary value of density is overwritten by the specified value. If this value is omitted, then the existing boundary value at the node indicated by the <b>dir</b> direction is used.

**func** One of the modifiers listed in Table 4.2.1 that denotes the functional form of the flow rate. If no function is specified, the value is assumed to be constant.

**ξ** One of the independent variables listed in Table 4.2.2. If no independent variable is specified, the variable is assumed to be time.

**subrgn** The subregion for which the input is specified. If no subregion is specified, then entire computational domain is selected.

**N<sub>1</sub>,...,N<sub>n</sub>** The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.

**fname** The name of the file containing the numerical values **N<sub>2</sub>** through **N<sub>n</sub>**. This option is available only for selected functions. See Section 3.3 for additional information.

**Φ<sub>1</sub>, ..., Φ<sub>m</sub>** The dependent variables for the properties that are injected along with the flow such as heat(enthalpy or temperature) and concentration of species, etc.

**N<sub>n+1</sub>, ..., N<sub>m</sub>** The corresponding values of **Φ<sub>1</sub>, ..., Φ<sub>m</sub>** that are injected with the flow. Each of these values

	must follow the symbol for the corresponding $\Phi$ .
<b>N<sub>m+1</sub>, ..., N<sub>k</sub></b>	The normalizing vector components, $V_i$ , if the <b>NORMAL</b> modifier is present. These must be the last numerical values on the input command except for the value of the density (see below). Two values must be specified for 2D and 3 for 3D flows.
<b><math>\rho_B</math></b>	The density value that multiplies the computed source. It can only be specified if the <b>DENSITY</b> modifier is present and then it must be the last value on the command. <i>If no value is specified but the modifier <b>DENSITY</b> is present, then default value is the boundary value at the node indicated by the <b>dir</b> modifier.</i>
<b>dir</b>	The orientation index for the element boundary associated with the source. See Section 3.5 for available choices. <i>There is no default value for this input.</i>

## REMARKS

---

The **FLOW** command is primarily designed to allow injection of fluid through a solid boundary (or blocked region) inside the domain of computation. Its use is most appropriate when the scale of injection is comparable to the grid size. The command is implemented in terms of boundary “fluxes” entering the domain of computations. If the scale of injection is much smaller than the grid size, then the **SOURCE FLOW** command may be more appropriate. The injection at an exterior domain boundary is more appropriately accounted for by the **INLET** command.

In implementing this command it is assumed that at the point of injection, the boundary wall has essentially been removed. The computational treatment is akin to that for inflow such as through an **INLET** boundary. If the injection occurs through a blocked element or at an exterior boundary of the domain, then the injected values are also assumed to be the boundary values of the variables.

If the wall needs to be retained, such as for flow percolating through a solid matrix or through holes much smaller than the boundary area of the inflow element, then the **INTERNAL** modifier should be used.

## EXAMPLES

---

Generic examples for this command are given in Section 4.4. The command keyword (**FLOW**) must replace the keyword used in these examples. Some additional examples that illustrate the use of the attributes specific to this command are given below.

**FLOW** injection: amount = 0.001 with T=50, C=1.

**FLOW** per unit AREA of X- face: = 0.001 with T=50, C=1.

**FLOW** withdrawal: amount =-0.001 per second

**FLOW** TABLE 3 sets SELEcted (0, 0), (100, 1), (200, 0) U=1, V=0.1, W=0, T=10, K=0.001, L=1

**FLOW**: EXPOntial series with TIME 7 sets from 'SOURCE' T=100, C=0.

**FLOW** q=10 X- direction T=100, U=20. NORMAlized velocity 0., 1.5, 2.5 ID=VSOURce;

**FLOW** q= -10 in X- dir with NORMAlized vel 1., 1.5, 2.5 and DENSIty for ID=VSOURce

**FLOW** q=10 X- dir variable values: U=10, V=0, W=-20, T=100, K=0.03, L=0.5 INTERnal with NORMAlized vel 1., 1.5, 2.5 and DENSIty for ID=VSOURce:

**FLOW** q=10 per unit VOLUme injected variables: U=10, V=0, W=-20, T=100, K=0.03, L=0.5 multiply by DENSIty = 5 for SELEcted region

**MODE 2:** Flow Injection with Fixed Variable Values and Computed Momentum Components

**SYNTAX** **FLOW** {**MOME**} [**TOTA**] [**option**] [**func**( $\xi$ )] {**N**<sub>1</sub>, **N**<sub>2</sub>, ..., **N**<sub>n</sub>|**fname**} [ **$\Phi$** <sub>1</sub>=**N**<sub>n+1</sub>, ...,  **$\Phi$** <sub>m</sub> =**N**<sub>m</sub>] [**N**<sub>m+1</sub>, **N**<sub>k</sub>] [ **$\rho_B$** ] [**subrgn**] {**dir**}

**MOME** Velocity components,  $V_j$ , of the injected flow are computed from flow rate,  $Q$ , for the element:

$$V_j = \frac{Q}{\rho_B A} n_j,$$

where  $\rho_B$ , is the density,  $A$  is the area of the element boundary specified by the **dir** modifier, and  $n_j$  is a normalizing vector. If the **NORMAL** modifier is present, then  $n_j$  is obtained from the user input, otherwise the area unit vector ( $A_j/A$ ; where  $A_j$  is the component in the  $j^{\text{th}}$  direction) is used as the normalizing vector. By default the density is the fluid density at the boundary, unless the user specifies a value. Any velocity input given by the user is ignored.

**TOTA** See Mode 1 specification

**option** See Mode 1 specification. Unless the **NORMAL** modifier is present, it is assumed that the **AREA** modifier is in effect. Any specification of the **VOLUME** modifier is ignored. All other modifiers can be used as for Mode 1 Specification.

**func** See Mode 1 specification.

$\xi$  See Mode 1 specification.

**N**<sub>1</sub>, ..., **N**<sub>n</sub> See Mode 1 specification.

**fname** See Mode 1 specification.

**$\Phi$** <sub>1</sub>, ...,  **$\Phi$** <sub>m</sub> See Mode 1 specification.

**N**<sub>n+1</sub>, ..., **N**<sub>m</sub> See Mode 1 specification.

**N**<sub>m+1</sub>, ..., **N**<sub>k</sub> See Mode 1 specification.

**$\rho_B$**  See Mode 1 specification.

**subrgn** See Mode 1 specification.

**dir** See Mode 1 specification. This modifier must be present for this mode of command.

**EXAMPLES**

All the examples cited for Mode 2, except those with **VOLUME** modifier, are applicable provided that the modifier **MOMENTUM** is added. Some illustrative examples specific to this mode are given below.

**FLOW** with MOMEntum q=10 X- direction with ID=VSOURce; T=100, K=0.05, L=0.02

**FLOW** MOMEntum q=10 X- direction with ID=VSOURce; T=100, K=0.05, L=0.02, DENSity= 5

**FLOW** MOMEntum q=10 X- direction with ID=VSOURce; T=100, K=0.05, L=0.02, NORMal 1. -1, 0

**FLOW** MOMEntum q=10 X- ID=VSOURce; T=100, K=0.05, L=0.02, NORMal 1. -1, 0, DENSity=5.

**FLOW** MOMEntum: TABLe 2 sets SELEct (0, 0), (100, 1) T=10, K=0.001, L=1

**FLOW** MOMEntum EXPOnential SERIEs TIME 7 sets 'SOURCE' T=100, C=0. NORMal 1. -1 SELEcted

**FLOW** MOMEntum q=10 X- dir INTERnal for ID=VSOURce: injected variables: T=100, K=0.03, L=0.5  
NORMALized vel 1., 1.5, -0.7 and DENSity as exists

**MODE 3: Flow Injection at Fixed Spherical Angles with Computed Momentum Components**

**SYNTAX** **FLOW** {**ANGL**} [**TOTA**] [**func**{**ξ**}] {**N**<sub>1</sub>, **N**<sub>2</sub>, ..., **N**<sub>n</sub>|**fname**} [**Φ**<sub>1</sub>=**N**<sub>n+1</sub>, ..., **Φ**<sub>m</sub> =**N**<sub>m</sub>] [**N**<sub>m+1</sub>, ..., **N**<sub>m+5</sub>] [**DENS**|**SPEE**] [**ρ<sub>B</sub>**|**V<sub>S</sub>**] [**subrgn**] [**dir**]

**ANGL** Fluid is injected or withdrawn. The amount of the property of the injected (or withdrawn) fluid acts as the source or sink for each of the relevant properties. If the flow is injected, then the velocity component of the injected flow, U<sub>i</sub> in the i<sup>th</sup> direction, is either computed from the source flow rate, Q, or determined from a specified injection speed, V<sub>s</sub>, as:

$$U_i = \frac{Q}{\rho_B A} n_i \quad \text{or} \quad U_i = v_s n_i$$

where ρ<sub>B</sub> is the density at the neighboring element and A is the area of the element boundary specified by the **dir** modifier. The local direction vector, n<sub>i</sub>, for each segment of the surface is obtained from the user input of two spherical angles and a reference axis.

- TOTA** See Mode 1 specification
- func** See Mode 1 specification.
- ξ** See Mode 1 specification.
- N<sub>1</sub>, ..., N<sub>n</sub>** See Mode 1 specification.
- fname** See Mode 1 specification.
- Φ<sub>1</sub>, ..., Φ<sub>m</sub>** See Mode 1 specification.
- N<sub>n+1</sub>, ..., N<sub>m</sub>** See Mode 1 specification.
- N<sub>m+1</sub>** The spherical angle, **Φ** in degrees measured as the deflection of the injection vector from the face normal. The magnitude of the angle must be less than 180 degrees.
- N<sub>m+2</sub>** The spherical angle, **Φ** in degrees measured as the rotation of the injection vector with respect to the projection of the axis vector onto the face plane.
- N<sub>m+3</sub>, ..., N<sub>m+5</sub>** The direction cosines of the axis vector used to measure the angle **Φ**; 3 values must be specified since this option is only available for 3D flows.
- DENS** The density, **ρ<sub>B</sub>**, is specified as the last value on the command.
- ρ<sub>B</sub>** See Mode 1 specification.
- SPEE** The injection speed, **V<sub>s</sub>**, is specified as the last value on the command.
- V<sub>s</sub>** The value **V<sub>s</sub>** if the **SPEED** modifier is present. **There is no default value for this input.**
- subrgn** See Mode 1 specification.
- dir** See Mode 1 specification.

**EXAMPLES**

All examples cited for Mode 1 are applicable provided that the modifier **ANGLE** is added and appropriate input for spherical angles is appended. Some illustrative examples of the use of attributes specific to this mode are given below.

- FLOW** q=10, X- direction with ID=VSOURce; T=100, K=0.05, L=0.02, Theta=45, phi=20, axis=1., 0, 0
- FLOW** q=10, X- ID=VSOURce; T=100, K=0.05, L=0.02, Theta=45, phi=20, axis=1., 0. 0, SPEEd=120
- FLOW** q=10, X- ID=VSOURce; T=100, K=0.05, L=0.02, Theta=45, phi=20, axis=1., 0. 0, DENsity=2.5

**MODE 4:** Flow Injection with Multiple Tabular Functions**SYNTAX** FLOW { $\Phi$ } {TABL} {MULT} [option] {N<sub>sets</sub>} {N<sub>1</sub>,...,N<sub>n</sub>|fname}[subrgn] **$\Phi$**  One or more of the symbols that denote the dependent variables, which are specified as functions of time. There is no default value; at least one symbol must be specified.**TABL** The variables are specified as tabular functions of time. This is the only available option in this mode.**MULT** Multiple variables are tabular functions of time.**option** See Mode 1 specification.**N<sub>sets</sub>** The number of sets of data for the tabular functions.**N<sub>1</sub>, ..N<sub>n</sub>** The N<sub>sets</sub> sets of data for the flow rate and variables as tabular functions of time. Each set must consist of time, flow rate, and one value for each variable specified by the symbol  $\Phi$  on the command, in that order. Thus if 4 variables are selected, then each data set must consist of 6 values. There is no default value; the correct number of values must be specified. If the flow rate is negative (withdrawal of fluid), then specified value of the property is ignored since it is assumed that the fluid is being withdrawn with an amount of property equal to the local value of that property.**fname** See Mode 1 specification.**subrgn** See Mode 1 specification.**EXAMPLES****FLOW** with MULTiple TABLe functions: 4 sets

Time	Flow	U	V	T
0.	0.001	1.00	0.02	100
1.	0.002	0.50	0.01	200
2.	0.004	2.00	0.01	500
5.	0.010	5.00	0.02	600

**FLOW** with MULTiple TABLe: 4 sets per unit VOLUME

Time	Flow	U	V	T
0.	0.001	1.00	0.02	100
1.	0.002	0.50	0.01	200
2.	0.004	2.00	0.01	500
5.	0.010	5.00	0.02	600

**MODE 5:** Flow Injection or Withdrawal at a Boundary

**SYNTAX** FLOW {ONLY} [option] [func[ξ]] {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub>|fname} [subrgn]

**ONLY** By default the injected (or withdrawn) flow is assumed to be accompanied by a specification of all other variables (properties) that are assumed to be automatically injected with it (see Mode 1 specification). If this modifier is present, then it is assumed that only the flow rate is specified by this command. The boundary flux of other variables is computed by a product of the injected flow and the prevailing value of the variable at the boundary node.

**option** See Mode 1 specification.

**func** See Mode 1 specification.

**ξ** See Mode 1 specification.

**N<sub>1</sub>,...,N<sub>n</sub>** See Mode 1 specification.

**fname** See Mode 1 specification.

**subrgn** See Mode 1 specification.

### EXAMPLES

---

Generic examples for this command are given in Section 4.4. The command keyword (**FLOW**) must replace the keyword used in these examples. Some additional examples that illustrate the use of the attributes specific to this command are given below.

**FLOW** T is TABLE per unit AREA in X- direction: 3 sets (TIME, value) (0., 0.01), (100., 0.10 ), (200, -0.20)

**FLOW** for T 10 per unit AREA in X- direction for SELEcted region

**FLOW** for T 10 per unit area in X- direction with NORMAlized velocity of 0., 1.5, 2.5 ID=VFLOW

**FLOW** for T 10 in X- dir with NORMAlized vel 1., 1.5, 2.5 and DENSIty for ID=VFLOW

**MODE 6:**      **Disable Previously Specified FLOW commands**

**SYNTAX**      **FLOW { $\Phi$ } {OFF} {subrgn}**

**$\Phi$**               See Mode 1 specification.

**OFF**             Previous **FLOW** commands for the identified subregion are deactivated. A new specification may follow.

**subrgn**         See Mode 1 specification.

#### **EXAMPLES**

---

**FLOW OFF** for T for most recently SELEcted region

**FLOW OFF** for T for ID=MIDDLE

5.39 FLUID

**PURPOSE** To specify thermal and transport properties of the fluid. This command is effective only for the **PORFLOW®** and **TIDAL®** Software Tools.

**MODE 1:** Fluid Compressibility

**SYNTAX** FLUI {COMP} [phase|SATU] {α}

**COMP** The compressibility of fluid is specified. The exact interpretation depends on the presence of the **phase** or **SATU** modifier. The compressibility is used to compute the storage term in the continuity equation. **This mode of input is applicable only for the PORFLOW® Software Tool.**

**phase** The fluid phase for which the input is specified. See Section 3.6 for available options. **If no phase modifier exists, the input is assumed to be for the 1<sup>st</sup> phase of the fluid.**

$$\alpha_f^n = \frac{1}{\rho^n} \frac{\partial \rho^n}{\partial P^n}$$

In this equation, α<sub>f</sub> is the fluid compressibility, ρ is the density, P the pressure and superscript n denotes the phase of the fluid. If the fluid phase is gas, then the fluid compressibility can be computed from the gas law as: α<sub>f</sub><sup>n</sup> = 1/P<sup>n</sup>.

**SATU** This modifier is meaningful only for multi-phase flow simulation in **PORFLOW®** Software Tool. However, if this modifier is present, the storage term also accounts for the changes in volume due to compression or expansion of the phase immediately “above” the one being simulated. In this case a term equal to:

$$\alpha_s = \min\left(\frac{\partial S}{\partial p_c}, \alpha\right)$$

Where α is user input value to set a minimum threshold for the computed phase saturation compressibility, α<sub>s</sub>. S is the phase saturation and p<sub>c</sub> is the corresponding capillary pressure.

**α<sub>f</sub>** The compressibility of the fluid; α<sub>f</sub> or the input for the minimum phase compressibility. **The default value is zero.**

**EXAMPLES**

---

- FLUID COMPRESSIBILITY is 5.E-10 for water
- FLUID COMPRESSIBILITY for SECONd phase is 1.E-5
- FLUID COMP due to SATURATION is 0.001

**MODE 2:** Fluid Enthalpy or Latent Heat

**SYNTAX** FLUI {LATE} {L}

**LATE** Latent heat of freezing option in **PORFLOW®** is specified.

**L** The latent heat of phase change. The default value is 1000.

**REMARKS** \_\_\_\_\_

This mode of input is applicable only for the freezing option in **PORFLOW®** Software Tool. For evaporation option, the specific heat-enthalpy relation must be specified. See the **SPECIFIC HEAT** command.

**EXAMPLES** \_\_\_\_\_

**FLUID LATENT** heat of freezing is **1000**

**MODE 3:**      **Specific Heat of the Fluid**

**SYNTAX**      **FLUI {SPEC}**

**COND**            The input is for the specific heat of the fluid.

**REMARKS** \_\_\_\_\_

This mode of the command has been superseded by the **SPECIFIC HEAT** command. Please refer to that command.

**MODE 4:** Thermal Conductivity or Mass Diffusivity of the Fluid

**SYNTAX** FLUI {COND}

**COND** The input is for the conductivity or diffusivity of the fluid.

**REMARKS** \_\_\_\_\_

This mode of the command has been superseded by the **CONDUCTIVITY** command. Please refer to that command.

**MODE 5:**      **Fluid Density**

**SYNTAX**      **FLUI {DENS}**

**DENS**          Density of the fluid is specified.

**REMARKS** \_\_\_\_\_

This mode of the command has been superseded by the **DENSITY** command. Please refer to that command.

## 5.40 FLUX

**PURPOSE** To compute and obtain output of the flux balance for a dependent variable for a selected subregion within the flow domain.

**MODE 1:** Computation and Output of Flux Balance for a Variable

**SYNTAX** FLUX { $\Phi$ } [subrgn] [dir] [fname] [TIME] [V<sub>Frq-File</sub>, V<sub>Frq</sub>] [NOW] [OLD]

**$\Phi$**  One or more symbols for the dependent variables for which the flux-balance output is required. The valid symbols are listed in Table 2.7.1. There is no default value.

**subrgn** The subregion for computations. If no subregion is specified, the entire domain is selected.

**dir** The orientation index for the boundary for which the flux output is required. See Section 3.5 for available choices. There is no default value for this input. In this case the convective and diffusive fluxes at the selected boundary are written to both the flux file and the standard output file in a tabulated form at the end of computations.

**fname** The name of the output file. If no name is specified then the output is directed to a file with the same name as the Standard Output file but with the extension removed and replaced by “\_FLUX.TMP”. For example, if the Standard Output file is “PROBLEM.OUT”, then the file is name named “PROBLEM\_FLUX.TMP”. At any time only one flux file can be open. Output from all active commands is directed to this file. If a new file name is given, then the previous file is closed and output from all active FLUX commands is directed to the new file

**TIME** By default, V<sub>Frq-File</sub> and V<sub>Frq</sub> are interpreted as the frequency of output in terms of number of steps. If this modifier is present, then V<sub>Frq-File</sub> and V<sub>Frq</sub> are interpreted to be the time interval between successive outputs.

**V<sub>Frq-File</sub>** The frequency (step or time interval) at which the fluxes are written to the flux file specified by 'fname'. See Section 3.7 for further details. The default value is 1.

**V<sub>Frq</sub>** The frequency (step or time interval) at which a summary of flux balance is written to the standard output file (file unit 16; see Section 2.4) in a manner similar to V<sub>Frq-File</sub>. The default value is such that a summary of fluxes is obtained only at the end of simulations.

**NOW** A flux record is written immediately both to the flux output file and the standard output device.

**OLD** The flux record to the flux file is written in the old format which has now been replaced. This modifier affects only the output to the file named by the fname modifier. It does not affect the output of fluxes to the Standard Output Unit. The use of this modifier is not recommended; it is retained only for compatibility with archived data sets and output.

**REMARKS**

All fluxes in ACRi Software are defined so that a flux ENTERING the computational domain (or a subregion) is POSITIVE and that LEAVING the domain is NEGATIVE.

The flux for mass is reported under the flux balance for pressure (P) since the continuity equation is solved in terms of pressure.

By default, the flux balance for each relevant dependent variable is automatically computed for the whole domain. This command can be used to obtain fluxes for additional subregions, to control the output, and to deactivate the flux computations in a selective manner. If no FLUX command is specified, then the summary of the flux balances is still directed to the standard output file at the end of computations. The output to the flux file, on the other hand, is generated only if a FLUX command is explicitly specified.

The flux balance output produced by ACRi Software Tools consists of cumulative fluxes for the subdomain and instantaneous fluxes crossing the boundaries. The flux balance for a variable is composed of cumulative changes due to storage ( $Q_0$  and  $Q_N$ ), inflow ( $Q_{in}$ ), outflow ( $Q_{out}$ ), and decay ( $Q_{decay}$ ). The flux balance disparity,  $Q_d$ , is defined as:

$$Q_d = (Q_0 - Q_N) + Q_{in} - Q_{out} - Q_{decay} + Q_{move} ,$$

where  $Q_0$  is amount of property present originally,  $Q_N$  is the amount present now, and  $Q_{in}$ ,  $Q_{out}$  and  $Q_{decay}$  are the cumulative inflow, outflow, decay, respectively.  $Q_{move}$  is the net influx due to moving or evolving sub-regions.

The  $Q_{in}$  and  $Q_{out}$  are, in turn, each composed of three components:

$$Q = Q_{conv} + Q_{diff} + Q_{so}.$$

Here  $Q_{conv}$  and  $Q_{diff}$  denote the cumulative contributions due to convective and diffusive fluxes, respectively, and  $Q_{so}$  denotes the net input due to sources (inflow – outflow).

**All ACRi software allows arbitrary user specified units.** The actual units for the reported fluxes therefore depend on the units selected by the user. For an arbitrary property,  $\Phi$ , the cumulative flux units are the units of  $\Phi$  multiplied by the units of the density ( $\rho$ ) and the units of volume ( $L^3$ ). For **PORFLOW®** the fluxes for mass and transport species (but not heat) are further divided by the density of the fluid; therefore these are in volumetric rather than mass units. The units of the instantaneous fluxes are those of the cumulative fluxes divided by units of time. **For steady state simulations**, the cumulative fluxes are the same as the instantaneous fluxes since no time integration is carried out.

**A more detailed description of each of the components that are reported in the flux output is given in the table that follows on the next page.** In this table:

1. **For transient simulations**, flux Items 4 through 21 is in cumulative units. For flow (pressure) equation these are in terms of mass units (e.g. kg) for **ANSWER®** and **TIDAL®**, and in terms of volume units (e.g.  $m^3$ ) for **PORFLOW®**. For thermal transport, the flux is reported in units of heat (e.g. J). For transport equation of a species, if the property is expressed in units of  $kg/m^3$ , then the flux balance is in terms of kg.
2. **For steady state simulations**, flux Items 4 through 21 (see Table below) are in instantaneous units. For flow (pressure) equation these are in terms of mass per unit time (e.g. kg/s) for **ANSWER®** and **TIDAL®**, and in terms of volume per unit time (e.g.  $m^3/s$ ) for **PORFLOW®**. For thermal transport, the flux is reported in units of rate of heat (e.g. J/s). For transport equation of a species, if the property is expressed in units of  $kg/m^3$ , then the flux balance is in terms of kg/s.
2. The instantaneous fluxes in Item 22 through 24 are in rate units. The units of these are the units of the cumulative flux expressed per unit time. In fact, the cumulative flux is the time integral of the instantaneous flux.

## EXAMPLES

**FLUX** for U for the entire domain

**FLUX** for T for SELEcted region every 20 steps

**FLUX** for V in SELEcted region: file every 20 steps; printer every 4 steps

**FLUX** balance for T for subregion defined by ID=VAULt every 200 steps

**FLUX** for T at XY plane defined by ID=PLN1 print every 10 steps

**FLUX** balance for T NOW for whole of the domain

**FLUX** balance for T for ID=VAULt OFF

**FLUX** for U to 'FLUX.OUT' for SELEcted region at TIME interval of 0.4

## Terms and Notation Used to Report Flux Balance in Output

#	Term and Notation	Description
1.	Active Subdomain:	Particulars of the Sub-domain for which Flux Balance is given
2.	Subdomain ID:	ID assigned to the Sub-domain by the user or the default ID automatically assigned if none given by User
3.	Time:	The simulation Time at which the flux summary is computed
4.	Net flux disparity (Qin-Qout-Qa-Qdecay)	The total “disparity” or “error” in the subregion. It is a sum of the total inflow minus the total outflow for the system
5.	Total inflow (Qin = Qc_in+Qd_in+Qsor):	Total inflow into the subregion by convective flux (Item 11), diffusive flux (item 12) and incoming (or positive) Sources (Item 13)
6.	Total outflow (Qout = Qc_out+Qd_out+Qsnk):	Total outflow from the subregion by convective flux (Item 14), diffusive flux (Item 15) and Sinks or outgoing (or negative) Sources (Item 16)
7.	Net accumulated gain (Qa = Qn-Qo)	Net increase in the amount of property (storage) in the subregion from the start of simulations.
8.	Decay from start (Qdecay)	Net accumulated decay in the property in the subregion from the start of simulations.
9.	Total initial property in region (Qo)	Initial amount of property present in the subregion at start of simulations
10.	Total property in region now (Qn)	Amount of property currently present in the subregion.
11.	Convective influx (Qc_in):	Net inflow of property due to convection from all boundaries of the subregion from the start of simulations.
12.	Diffusive influx (Qd_in):	Net inflow of property due to diffusion or dispersion from all boundaries of the subregion from the start of simulations.
13.	Source influx (Qsor):	Net inflow of property due to sources in the subregion from the start of simulations.
14.	Convective outflux (Qc_out)	Net outflow of property due to convection from all boundaries of the subregion from the start of simulations.
15.	Diffusive outflux (Qd_out)	Net outflow of property due to diffusion or dispersion from all boundaries of the subregion from the start of simulations.
16.	Sink outflux (Qsnk):	Net outflow of property due to sources or sinks in the subregion from the start of simulations
17.	Flux disparity due to mass balance (div*F)	Error in flux balance due to the fact that the flow field itself may not be mass conservative. This is included in the total disparity reported in Item 4 above. The error in flux balance can be both due to the error in the flow and that in the solution of the transport equation.
18.	Convective Influx from Blocks	The inflow from the blocked areas of computational domain that were defined by the <b>BLOCK</b> or <b>LAND</b> command.
19.	Diffusive Loss to Blocks	The diffusive/conductive outflow to the blocked areas of computational domain that were defined by the <b>BLOCK</b> or <b>LAND</b> command.
20.	Influx due to Sub-domain Movement	The inflow due to movement of regions that evolve or move during simulation and change their nature of position.
21.	Change from last time step	Change in the amount for property from the last time step
22.	Instantaneous convective flux (in-out):	The current net inflow of convective flux at all boundaries of the subregion.
23.	Instantaneous diffusive flux (in-out):	The current net inflow of diffusive flux at all boundaries of the subregion.
24.	Instantaneous source-sink:	The current net inflow of property due to sources in the subregion.

**MODE 2:** Convective Flux and Flux-Average Value of a Variable

**SYNTAX** FLUX {AVER} [Φ] [option] [subrgn] [dir] [fname] [TIME] [V<sub>Frq</sub>] [NOW] [OFF]

**AVER** The convective flux and the flux-weighted mean of the variable, its average, minimum and maximum values, the flow rate and, area at the selected boundary (or boundaries) are printed to the output device or file.

**Φ** One or more symbols for the dependent variable for which output is required. Up to 10 symbols may be specified per command. The valid symbols are listed in Table 2.8.1-3. If no symbol is specified then output is obtained for each active variable.

**option** The boundary type for which output is required. More than one option may be selected. If an option is specified, then the subrgn and dir modifiers are ignored. If no option is specified, then the averages are computed for the specified subrgn and dir modifiers.

option	INTERPRETATION
<b>INLE</b>	Boundaries specified by the <b>INLET</b> command are selected.
<b>OUTL</b>	Boundaries specified by the <b>OUTLET</b> command are selected.
<b>OPEN</b>	Boundaries specified by the <b>OPEN</b> command are selected.
<b>IO</b>	All boundaries specified by <b>INLET</b> , <b>OUTLET</b> or <b>OPEN</b> command are selected.
<b>WALL</b>	Walls specified by <b>WALL</b> or <b>BLOCK</b> command are selected.
<b>EXTE</b>	All external (or outer) boundaries of the computational domain are selected.
<b>ALL</b>	All of the above boundaries are selected.

**subrgn** The subregion for computations. This specification is ignored if any of the option modifiers is present. If no subregion is specified, the entire domain is selected.

**dir** The orientation index for the boundary for which the output is required See Section 3.5 for available choices. If no input is given, then the output is obtained for all boundaries of the subrgn. This specification is ignored if any of the option modifiers is present.

**fname** The file name for output. If a file name is specified then the output is directed to the named file, otherwise the output is directed to the standard output device.

**TIME** By default V<sub>Frq</sub> is interpreted as the frequency of output in terms of number of steps. In the presence of TIME this is the time intervals between successive outputs.

**V<sub>Frq</sub>** The frequency (step or time interval) at which the output is written to the output device. See Section 3.7 for further details. By default the output is obtained only at the end of simulations.

**NOW** A record of fluxes is written to the output device at the next time step when the equation for the dependent variable is solved. This is in addition to the output from the V<sub>Frq</sub> specification.

**OFF** Any previously specified FLUX AVERAGE commands for the specified variables and subregion are disabled. New commands may be subsequently specified.

**EXAMPLES**

- FLUX AVERAge for ALL boundaries ! All active variables by default
- FLUX AVERAge for T (temperature) at IO boundaries ! All INLEt, OUTLEt & OPEN boundaries
- FLUX AVERAge for T (temperature) at INLEt and OPEN ! INLEt & OPEN boundaries
- FLUX AVERAge for T at INLEt and BOUNDaries ! INLEt & domain boundaries
- FLUX AVERAge for T at ALL on file 'FLUX.FIL'
- FLUX AVERAge for T at ALL every 5 steps to file 'FLUX.FIL'
- FLUX AVERAge for T at ALL at TIME=0.01 to file 'FLUX.FIL'
- FLUX AVERAge to file 'FLUX.FIL' every 1 step

**MODE 3:**      **Disable Previously Specified Mode 1 Flux Balance Computations**

**SYNTAX**      **FLUX {OFF} { $\Phi$ } {subrgn}**

**OFF**            Any previously specified flux balance command for the specified variable and subregion is disabled. New **FLUX** commands may be subsequently specified.

**$\Phi$**               The symbol for the dependent variable for which the previously specified flux-balance computation is disabled. The valid symbols are listed in Table 2.7.1. **One and only one character string must be specified for each command.**

**subrgn**        The subregion for flux computations. **If no subregion is specified, the entire domain is selected.**

#### **EXAMPLES**

---

**FLUX** balance computation OFF for T for whole domain

**FLUX** computations OFF for T for the currently active subregion

**FLUX** computations OFF for T for the ID=UPPEr

**MODE 4:**      **Disable Default Flux Balance Computations**

**SYNTAX**      **FLUX {OFF}**

**OFF**            By default the flux balance is computed automatically, for the whole of the computational domain, for each variable for which a transport equation is solved. This is in addition to any flux balance computation in response to **FLUX** commands for that variable. If this modifier is present, then the automatic flux computations are disabled with one exception. The exception is the mass balance equation, for which the flux balance is always computed irrespective of the user input.

#### **EXAMPLES**

---

**FLUX** balance computation OFF for all variables except mass

**FLUX** default computations OFF

**FLUX** OFF

**5.41 FOR**

**PURPOSE** To select the soil or rock material types or soil/rock zones to which the property information following the **FOR** specification is applicable.

**SYNTAX** **FOR** {N1}

**N1** The material type number to which the property specification applies. The maximum number of material type is set by default to be 10240.

**REMARKS**

---

**This command is being retained temporarily only for backward compatibility. Its function has been superseded by the specification of ID=Idsub on the input commands. The use of this command is not recommended.**

The property information to which this command applies is specified through the **HYDRAULIC**, **MULTIPHASE**, **ROCK**, **SOIL**, **THERMAL** and **TRANSPORT** commands. The input remains effective for subsequent property commands until another **FOR** command is encountered.

**EXAMPLES**

---

**FOR** material type **3** properties are specified by the following commands

## 5.42 FRACTURE

<b>PURPOSE</b>	Define embedded fractures in the porous matrix. This command is effective only for the <b>PORFLOW®</b> Software Tool.
<b>MODE 1:</b>	<b>Specification of a Fracture with Arbitrary Orientation</b>
<b>SYNTAX</b>	<b>FRAC</b> { <b>ID=</b> subrgn} { <b>W<sub>frac</sub></b> } [ <b>func</b> [ $\xi$ ]] [ <b>N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub></b>   <b>fname</b> ] [ <b>ADJU</b> ] [ <b>DETA</b> ]
<b>FRAC</b>	The material type designation is for a fracture. This feature is only available with the structured grid mode of <b>PORFLOW®</b> .
<b>ID</b>	The modifier to indicate that the input is for a previously defined subregion.
<b>subrgn</b>	The subregion in which the fracture is located. <b>The subregion must be previously specified by a LOCATE FRACTURE command.</b>
<b>W<sub>frac</sub></b>	Nominal width of the fracture which may be replaced by a computed value if the function is specified on the command. <b>There is no default value; a value &gt; 0 must be specified.</b>
<b>func</b>	One of the modifiers listed in Table 3.4 that denotes the functional form for fracture width. <b>If no function is specified, the value is assumed to be constant.</b>
$\xi$	One of the independent variables listed in Table 4.2.2.
<b>N<sub>1</sub>,...,N<sub>n</sub></b>	The numerical constants and coefficients for the selected function. See Section 4.4 for more details. <b>There are no default values for this input.</b>
<b>fname</b>	The name of the file containing the numerical values <b>N<sub>2</sub></b> through <b>N<sub>n</sub></b> . This option is available only for selected functions. See Section 3.3 for additional information.
<b>ADJU</b>	If this modifier is present, and a functional form for the width is specified, then the hydraulic conductivity of the fracture is modified as a cubic function of the width.
<b>DETA</b>	If this modifier is present, then a detailed table of the elements and faces that form the fracture is printed out to the Standard Output Device

**REMARKS**

**This mode of the command allows the specification of a fracture with arbitrary orientation.** See the **LOCATE FRAC** command for further details. This type of fracture can evolve with time in its own plane if the **MOVE** modifier is present on the corresponding **LOCATE FRAC** command and one or more **MOVE** commands are specified for the evolution of the fracture with time.

The fracture properties must be specified with the **POROSITY, HYDRAULIC, THERMAL, TRANSPORT** and **DENSITY SOLID**, commands. Functional options for corresponding properties that apply to the porous matrix are not available for Fractures. Specification of solid density is optional; other input must be specified if the corresponding transport equation is solved. If solid density is not specified, it is set equal to the porous matrix density at the 1<sup>st</sup> fracture element..

**EXAMPLES**

```
FRACture ID=FRACTURE_01 width = 0.1; print DETAIL
FRACture ID=UPPER_FRAC width = 0.1; varies LINEARLY 0.1 + 0.0001 * P; ADJUST K_hydraulic
```

<b>MODE 2:</b>	<b>Specification of a Fracture Aligned with Principal Axis</b>
<b>SYNTAX</b>	<b>FRAC {ID=subrgn} {W<sub>frac</sub>} [func[ξ]] [N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub> fname] [INCL] [dir ALL NONE] [ADJU] [DETA]</b>
<b>FRAC</b>	The material type designation is for a fracture. This feature is only available with the structured grid mode of <b>PORFLOW®</b> .
<b>ID</b>	The modifier to indicate that the input is for a previously defined subregion.
<b>subrgn</b>	The subregion in which the fracture is located. <b>The subregion must be previously specified by a suitable LOCATE command without the FRAC modifier. The subregion must be such that a continuous path exists for the flow. It is best to use an index (LOCATE IJK) or coordinate (LOCATE COOR) based window for this subregion.</b>
<b>W<sub>frac</sub></b>	Nominal width of the fracture which may be replaced by a computed value if the function is specified on the command. <b>There is no default value; a value &gt; 0 must be specified.</b>
<b>func</b>	One of the modifiers listed in Table 4.2.1 that denotes the functional form for fracture width. <b>If no function is specified, the value is assumed to be constant.</b>
<b>ξ</b>	One of the independent variables listed in Table 4.2.2.
<b>N<sub>1</sub>,...,N<sub>n</sub></b>	The numerical constants and coefficients for the selected function. See Section 4.4 for more details. <b>There are no default values for this input.</b>
<b>fname</b>	The name of the file containing the numerical values <b>N<sub>2</sub></b> through <b>N<sub>n</sub></b> . <b>This option is available only for selected functions. See Section 3.3 for additional information.</b>
<b>INCL</b>	By default only the boundary faces in the principal direction (determined from the <b>LOCATE</b> command) are included in the definition of the fracture faces. If this modifier is present, together then the boundary faces specified are included based on the dir, ALL or NONE modifier.
<b>dir</b>	One of the direction modifiers ( <b>X-, X+, Y-, Y+, Z-, Z+</b> ) which indicates the face to be included if the <b>INCL</b> modifier is present.
<b>ALL</b>	All boundary faces in the sub domain are included if the <b>INCL</b> modifier is present.
<b>NONE</b>	No boundary faces in the sub domain are included if the <b>INCL</b> modifier is present.
<b>ADJU</b>	See Mode 1.
<b>DETA</b>	See Mode 1.

**REMARKS**

This is an older mode of specifying fracture which is retained for backward compatibility. **Mode 1 should be used which allows arbitrary orientation.** This type of fracture can evolve with time in its own plane if the **MOVE** modifier is present on the corresponding **LOCATE** command and one or more **MOVE** commands are specified for the evolution of the fracture with time.

**EXAMPLES**

**FRAC**ture ID=FRACTURE\_X width = 0.1

**FRAC**ture ID=FRACTURE\_X width = 0.1; varies LINEARLY 0.1 + 0.0001 \* P; ADJUST K\_hydraulic

## 5.43 FRICTION

**PURPOSE** To specify the bottom friction or wind stress coefficients. This command is effective only for the TIDAL® Software Tool.

**MODE 1:** Bottom Friction Coefficients

**SYNTAX** FRIC [MANN|CHEZ], {C<sub>f1</sub>}, [C<sub>f2</sub>]

**MANN** By default the bottom friction coefficient is assumed to be constant. If this modifier is present then the bottom friction is calculated according to Manning's formula.

$$C_f = g \left\{ \frac{C_{f1}}{C_{f2}} \right\}^2 \frac{1}{H^{1/3}}$$

In the above relation, **g** is the constant of gravitational acceleration and **H** is the total depth of water.

**CHEZ** If this modifier is present then the bottom friction is calculated according to Chezy's formula.

$$C_f = \frac{g}{C_{f1}^2} \left( C_{f2} + \frac{1}{\sqrt{H}} \right)^2,$$

**C<sub>f1</sub>** When no modifier is present then this value is equal to the bottom friction coefficient that appears in the momentum equations. When either **MANN** or **CHEZ** is present it is equal to C<sub>f1</sub> of the respective formula. The default value is 0.02.

**C<sub>f2</sub>** When no modifier is present this input is ignored. When either **MANN** or **CHEZ** is present it is equal to C<sub>f2</sub> of the respective formula. The default value is 1.0.

**REMARKS**

This command is implemented only for the TIDAL® Software Tool.

**EXAMPLES**

**FRIC**tion coefficient on the bottom: 0.003

**FRIC**tion coefficient on the bottom: 0.003

**FRIC**tion coefficient CHEZy's formula: cf1=0.001, cf2 =1.

**FRIC**tion coefficient MANNing's formula: cf1=0.01, cf2 = 1.5.

**MODE 2:** Specification of Wind Stress Coefficients and Reference Wind Speed

**SYNTAX** **FRIC** {**WIND**} [**C<sub>w1</sub>**, **C<sub>w2</sub>**, **w<sub>0</sub>**]

**WIND** The input is for the wind stress coefficients at a given wind speed, W, according to the formula given below.

$$C_w = C_{w1} + C_{w2} \left\{ \max \left[ 0, \left( 1 - \frac{w_0}{W} \right)^2 \right] \right\},$$

**C<sub>w1</sub>** Linear component of wind stress coefficient, C<sub>w1</sub>, in the equation given above. The default value is 1.0E-6.

**C<sub>w2</sub>** Non-linear component of wind stress coefficient, C<sub>w2</sub> in the equation given above. The default value is 1.4E-6.

**w<sub>0</sub>** Reference wind speed, w<sub>0</sub> in the equation given above. The default value is 7.0.

### EXAMPLES

---

**FRIC**tion parameters for WIND: C<sub>w1</sub> = 2.4E-6, C<sub>w2</sub> = 0.00, w<sub>0</sub> = 0 m/s

**FRIC**tion - WIND parameters: all values are set to zero

**5.44 FUEL**

**PURPOSE** To specify the amount of carbon and hydrogen in the fuel, and the fuel enthalpy. This command is effective only for the **ANSWER®** Software Tool.

**SYNTAX** **FUEL** {**N<sub>C</sub>**, **N<sub>H</sub>**} [**H<sub>C</sub>**]

**N<sub>C</sub>** The fuel for a combustion process is assumed to have a chemical formula of the type C<sub>n</sub>H<sub>m</sub>. The N1 (>0) specifies the number of carbon atoms n. The default value is 10.

**N<sub>H</sub>** The number of hydrogen atoms (>0), the m in the fuel formula. The default value is 19.

**H<sub>C</sub>** The lower heat of combustion (>0), energy per unit mass; for fuel in J/kg. The default value is 4.50E7.

**EXAMPLES**

---

**FUEL** formula C=8 H=18

**FUEL** formula C=8 H=18: heat of combustion 5.788E7

**FUEL** is equivalent to C=10.25, H=20.5

## 5.45 GAS

**PURPOSE** To specify composition and properties of the gas phase. This command is effective only for the ANSWER® and PORFLOW® Software Tools.

**MODE 1:** Gas Molecular Weights for Gas Constituents

**SYNTAX** GAS { [  $\Phi_1=N1, \Phi_2=N2, \dots, \Phi_n=Nn$  ] }

$\Phi_n$  One or more of the symbols for gas constituents for which the molecular weights are specified. The valid symbols are problem dependent and may be selected by the user from the list of default dependent variables or those defined by the user. The user may define up to 64 gas constituents.

$N_1, N_2, \dots, N_n$  Molecular weights (>0) for the variables. If no value is specified, then any previous specification stays in effect. If only one value is specified, then it is applied to all specified variables. If more than 1 value is specified, then the number of values must equal the number of symbols on the command.

If only 1 numerical value is specified **without any symbol** then it is assumed that the value specifies the molecular weight of the default gas, which is assumed to be the 1<sup>st</sup> defined transport variable.

**REMARKS**

If this mode of the command is missing then by default the gas is defined to be single species gas with a molecular weight of 29. The one exception to this rule is the ANSWER® Software Tool if a REACTION command is given with default chemical reactions. In this case 8 gas species are automatically defined; these are FU, CH, O2, CO, CO2, H2, H2O and N2.

**EXAMPLES**

**GAS** molecular weights: H2O=18, CO2=44, N2 =28, O2=32

**GAS** molecular weight is 18! Gas species by default

**MODE 2:** Gas Datum and Reference Pressure

**SYNTAX** GAS {PRES|P} {V1} [V2]

**PRES** The reference datum gas pressure,  $p^*$ , is specified.

**P** Same as the modifier **PRES** above.

**V1** The datum pressure for the fluid. All pressure values, including the initial and boundary values, are relative to this pressure. This pressure will be added to the gas pressure to compute the absolute gas pressure if the gas density law is invoked. The default value for the datum pressure is 101325 for incompressible mode of computations but 0 for compressible mode of computations.

**V2** This value is read only if the compressible mode of computations is invoked. In this case, it is interpreted as the reference value of the gas pressure which is required for some of the computations (such as the droplet evaporation correlations) where the ratio of the gas pressure to the atmospheric pressure is required. If this value is not specified, then the reference gas pressure is computed as the average of the specified initial pressure.

For incompressible flow, the reference gas pressure is set equal to the datum pressure.

#### EXAMPLES

---

**GAS** datum PRESsure is 2.5E5

**GAS** datum P is 2.5E5

**GAS** P = 0, Datum=0., reference value = 1.E6

**MODE 3:** Gas Sonic Speed and Specific Heat Ratio

**SYNTAX** GAS {SONI|GAMM} {V1}

**SONI** The sonic speed for the gas is specified. This may be used to compute the Mach number of the flow (see the **OPTION** command).

**GAMM**  $\gamma$ , the ratio of specific heat at constant pressure to that at constant volume is specified. This constant may be used to compute the Mach number of the flow (see the **OPTION** command).

**V1** The reference sonic speed or  $\gamma$  for the gas. The default value is 300 for the sonic speed and 1.4 for the ratio of specific heats ( $\gamma$ ).

#### EXAMPLES

---

**GAS** SONIC speed is 275 m/s

**GAS** GAMMA is 1.2

<b>MODE 4:</b>	<b>Universal Gas Constant</b>
<b>SYNTAX</b>	<b>GAS</b> { <b>CONS</b> } [ <b>func</b> [ $\xi$ ]] [ <b>N</b> <sub>1</sub> , <b>N</b> <sub>2</sub> , ..., <b>N</b> <sub>n</sub>   <b>fname</b> ]
<b>CONS</b>	The universal gas constant, $R_u$ used to compute fluid density and other gas properties.
<b>func</b>	One of the modifiers listed in Table 4.2.1 which denotes the functional form of the gas constant. If no function is specified, then the gas constant is assumed to be fixed.
$\xi$	One of the independent variables listed in Table 4.2.2. If no variable is specified, the independent variable is assumed to be time.
<b>N</b> <sub>1</sub> ,..., <b>N</b> <sub>n</sub>	The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.
<b>fname</b>	The name of the file containing the numerical values <b>N</b> <sub>2</sub> through <b>N</b> <sub>n</sub> . This option is available only for selected functions. See Section 3.3 for additional information.

### EXAMPLES

---

Generic examples for this command are given in Section 4.4. The command keyword (**GAS**) must replace the keyword used in these examples. Some additional examples are given below.

**GAS** CONStant is 1545!!!! ft lbf/lb-mole R

**GAS** CONStant is LINEar function: 30. -0.015 \* P

**GAS** CONStant POLYnomial in P: (10., 0.5, 0., -0.1, 0.)

**GAS** CONStant is a TABLe in P: 3 sets: (0., 0.01), (100., 0.10 ), (200, -0.20)

## 5.46 GEOMETRY

**PURPOSE** To modify or specify the geometry for the computational domain.

**MODE 1:** Exchange Previously Specified Coordinates

**SYNTAX** **GEOM** {**EXCH**} {**dir<sub>1</sub>**} {**dir<sub>2</sub>**}

**EXCH** The coordinates in the two directions specified by **dir<sub>1</sub>** and **dir<sub>2</sub>** are exchanged with each other. For example, with **x** and **y** as the specified directions, the new **x** and **y** coordinates at a point will be equal to the old **y** and **x** coordinates, respectively, at that point. For an orthogonal grid this effectively results in a rotation of the coordinate system by 90 degrees.

**dir<sub>1</sub>, dir<sub>2</sub>** Two of the **X**, **Y**, **Z**, **R** or **THETA** modifiers that, respectively, denote the **x**, **y**, **z**, **r** or  $\theta$  coordinates to be exchanged with each other.

**EXAMPLES**

---

**GEOM**etry **EXCH**ange **X** and **Y**

**GEOM**etry **EXCH**ange **X** and **R**

**GEOM**etry **EXCH**ange **X** and **Z**

**MODE 2:** Rotate Previously Specified Coordinates

**SYNTAX** **GEOM** {**ROTA**} {**dir<sub>1</sub>**} {**dir<sub>2</sub>**} [**φ<sub>1</sub>**, **φ<sub>2</sub>**]

**ROTA** The coordinates in the two directions specified by **dir<sub>1</sub>** and **dir<sub>2</sub>** are rotated in the plane defined by **dir<sub>1</sub>** and **dir<sub>2</sub>** by **φ<sub>1</sub>** and **φ<sub>2</sub>** degrees, respectively, according to the following equations:

$$X_{\text{New}} = X_{\text{Old}} \cos \phi_1 + Y_{\text{Old}} \cos \phi_2$$

$$Y_{\text{New}} = X_{\text{Old}} \sin \phi_1 + Y_{\text{Old}} \sin \phi_2$$

**dir<sub>1</sub>, dir<sub>2</sub>** Two of the **X**, **Y**, **Z**, **R** or **THETA** modifiers that, respectively, denote the x, y, z, r or θ coordinates to be rotated in the plane defined by **dir<sub>1</sub>** and **dir<sub>2</sub>**.

**φ<sub>1</sub>** The angle **φ<sub>1</sub>** (in degrees) for the transformation defined above. For a rectangular system, it represents the angle between the old and the new x-direction coordinates. The default value is 0 degrees.

**φ<sub>2</sub>** The angle **φ<sub>2</sub>** (in degrees) for the transformation defined above. For a rectangular system, it represents the angle between the old x-direction and new y-direction coordinates. If no value is specified then **N2 = N1+90 degrees**.

## EXAMPLES

---

**GEOM**etry **ROTA**te X and Y by 45 degrees

**GEOM**etry **ROTA**te X by 45 and Y by 135 degrees!! same as above

**GEOM**etry **ROTA**te X and Y by -45 deg.

**GEOM**etry **ROTA**te X and Y 30 and Z by 115 degrees from old x-axis

**MODE 3:**      **Scale Previously Specified Coordinates**

**SYNTAX**      **GEOM {SCAL} {dir} { V<sub>Mult</sub> } [V<sub>Add</sub>]**

**SCAL**          The coordinates in the direction(s) specified by **dir** are scaled according to the following equation:

$$X_{\text{New}} = X_{\text{Old}} * V_{\text{Scale}} + V_{\text{Add}}$$

**dir**             One or more\_ of the **X, Y, Z, R** or **THETA** modifiers that, respectively, denote the x, y, z, r or  $\theta$  coordinates each of which is scaled according to the equation given above.

**V<sub>Mult</sub>**          The multiplication factor for the transformation defined above. **There is no default value; a value must be specified.**

**V<sub>Add</sub>**          The offset for the coordinate. **The default value is 0.**

### EXAMPLES

---

**GEOM**etry **SCAL**e X by 0.3048

**GEOM**etry **SCAL**e X by 0.3048 add 1.00

**GEOM**etry **SCAL**e X and R by 0.3048 add 1.00

**GEOM**etry **SCAL**e X, Y and Z by 0.3048 add 1.00

**MODE 4:**      **Generate Cylindrical Geometry in r - $\theta$  Plane**

**SYNTAX**      **GEOM {CYLI|CIRC} (R) [ $\theta_1$ ,  $\theta_0$ ]**

**CYLI**            Generates a cylindrical grid for the computational domain in the r - $\theta$  plane.

**CIRC**            Same as **CYLI** modifier.

**R**                The radius of the cylinder. **There is no default value; a value must be specified.**

**$\theta_1$ ,**            The angle of the cylinder circumference in degrees. **The default value is 180 degrees.**

**$\theta_0$**             The starting angle from the origin for the cylinder in degrees. **The default value is 0 degrees.**

**REMARKS**\_\_\_\_\_

This command is available only if the computational domain is located in the 1<sup>st</sup> or 2<sup>nd</sup> quadrant.

**EXAMPLES**\_\_\_\_\_

**GEOM**etry is **CYL**indrical with radius = 1.

**GEOM**etry is **CIRC**ular with r = 1, theta = 90

**GEOM**etry is **CYL**indrical with radius = 1., theta = 90

**GEOM**etry is **CYL**indrical with r = 1., theta = 90, alpha=90

**MODE 5:**      **Generate Annular Geometry in r -  $\theta$  Plane**

**SYNTAX**      **GEOM {ANNU} {R<sub>o</sub>} [R<sub>i</sub>, e,  $\theta_1$ ,  $\theta_0$ ]**

**ANNU**              Generates an annular geometry for the computational domain in the **r -  $\theta$**  plane.

**R<sub>o</sub>**                  The outer radius of the annulus. **There is no default value; a value must be specified.**

**R<sub>i</sub>**                  The inner radius of the annulus. **The default value is 0.**

**e**                    The eccentricity of the annulus. The inner cylinder of the annulus is displaced to the left of the outer cylinder by this value. **The default value is 0.**

**$\theta_1$**                  The total angle for the outer arc of the cylinder in degrees. **The default value is 180 degrees.**

**$\theta_0$**                  The starting angle from the origin for the cylinder in degrees. **The default value is 0 degrees.**

#### **REMARKS**

---

Currently this command is available only if the computational domain is located in the 1<sup>st</sup> or 2<sup>nd</sup> quadrant.

#### **EXAMPLES**

---

**GEOM**etry is ANNular with radius = 1.

**GEOM**etry is ANNular with rin=1, rout=0.5

**GEOM**etry is ANNular with rin=1, rout=0.5, epsilon=0.1

**GEOM**etry is ANNular with rin=1, rout=0.5, epsilon=0.1, theta = 90

**GEOM**etry is ANNular: rin=1, rout=0.5, epsilon=0.1, theta = 90, alpha=90

**MODE 6:** Transform from Cartesian to Cylindrical Geometry, or vice versa

**SYNTAX** **GEOM** {**CART** | **RADI**} {**dir**<sub>1</sub>} {**dir**<sub>2</sub>}

**CART** Convert existing cylindrical radial coordinates to Cartesian coordinates according to the transformation:

$$x_1 = r \cos \theta$$

$$x_2 = r \sin \theta$$

**RADI** Convert existing Cartesian coordinates to cylindrical radial coordinates according to the transformation:

$$r = \sqrt{x_1^2 + x_2^2}$$

$$\theta = \tan^{-1}(x_2/x_1)$$

**dir<sub>1</sub>, dir<sub>2</sub>** Two of the **X, Y, Z, R** or **THETA** modifiers that, respectively, denote the  $x_1$  and  $x_2$  directions in the Cartesian framework to be transformed to or from the **r** and **θ** directions of the cylindrical coordinate system. By default **r** direction is assumed to be aligned with the **y** direction and the **θ** with the **z** direction.

#### REMARKS

---

Transformation from Cartesian to radial framework is possible only for a 3-dimensional grid since in the 2-dimensional mode only (x, r) coordinate system is available. For the transformation to a cylindrical system the resulting **r** is stored at the same location as  $x_1$  and **θ** at  $x_2$ . For transformation to a Cartesian system the resulting  $x_1$  is stored at the same location as **r** and  $x_2$  at **θ**.

#### EXAMPLES

---

**GEOM**etry converts **R** and **THETA** to **CART**esian

**GEOM**etry convert **Y** and **Z** to **RAD**ial !Only with a 3D grid system

**GEOM**etry convert **X** and **Y** to **RAD**ial !Only with a 3D grid system

## 5.47 GRAVITY

**PURPOSE** To specify the gravitational acceleration vector and its components.

**SYNTAX** **GRAV** {**g<sub>x</sub>**, **g<sub>y</sub>**, **g<sub>z</sub>**}, [**g**] [ **ABSO**|**RELA**|**PASS**]

**g<sub>x</sub>**, **g<sub>y</sub>**, **g<sub>z</sub>** The components of the gravitational acceleration vector,  $g_j$ , in the three principal directions. A positive value implies that the acceleration vector is directed in the principal direction; a negative value implies that it is directed opposite to the principal direction.

For **ANSWER**® and **TIDAL**® Software Tools, the default values are all set to 0.

For **PORFLOW**® Software Tool, only the ratio  $g_j / g$  appears in the equations, therefore normalized values may be specified for this input. The default values are set so that, for multiphase or free-surface flow, **g<sub>y</sub>** is -9.81 for 2D flow and **g<sub>z</sub>** is -9.81 for 3D flow; otherwise, the values are 0

**g** The absolute value of the gravitational constant,  $g$ . If no value is specified, then the value is computed as the square root of the sum of squares of the 3 components. If no **GRAVITY** command is specified then the default value is set to 9.81

For **PORFLOW**® Software Tool, **g** is used for computing pressure head,  $P$ , from thermodynamic pressure,  $p$ , or vice versa. This input is necessary only if **g<sub>x</sub>**, **g<sub>y</sub>**, **g<sub>z</sub>** are specified in a normalized mode.

**ABSO** For the **ANSWER**® Software Tool, the gravitational body force term,  $B_i$  in the  $i^{\text{th}}$  direction momentum equation is given by:

$$B_i = \rho g_i$$

In this case the computed pressure represents the thermodynamic pressure,  $p$ . This is the default option for the **ANSWER**® Software Tool.

For the **PORFLW**® Software Tool the pressure equation is solved for the thermodynamic pressure head,  $P^n$ . In this case the gravitational body force,  $B_i^n$ , for the  $n^{\text{th}}$  phase in the  $i^{\text{th}}$  direction momentum equation is given by:

$$B_i^n = \frac{\rho^n g_i}{\rho^* g}$$

**RELA** For the **ANSWER**® Software Tool, the gravitational body force term,  $F_i$  in the  $i^{\text{th}}$  direction momentum equation is modified and defined as:

$$F_i = (\rho - \rho^*) g_i$$

Where  $\rho^*$  is the reference density. However, the computed pressure,  $P$ , is no longer equal to the thermodynamic pressure,  $p$ , but is related to it by:

$$P = p - \rho^* \{g_x x + g_y y + g_z z\}$$

For the **PORFLOW**® Software Tool the gravitational body force,  $B_i^n$  in the  $i^{\text{th}}$  direction of the momentum equation is modified as:

$$B_i^n = \frac{\rho^n - \rho^*}{\rho^*} \frac{g_i}{g}$$

The pressure equation is solved for the Total head,  $H^n$ . This is the default option for the **PORFLOW**® Software Tool.

This **RELATIVE** mode improves the stability of computations by decreasing the magnitude of the body force and hence the round-off errors. This should be the preferred mode of computations for most problems.

**PASS** The body force in the equations is ignored.

**APPLICABILITY NOTES**

For the **ANSWER®** Software Tool the absolute value (**g**) is used only if the **PROBLEM ATMOSpheric** or **REFERENCE HYDRaulic** command is present. In this case, this value is used to compute the hydrostatic pressure head from the thermodynamic pressure. Otherwise only the individual components in each direction appear in the momentum equations.

For the **PORFLOW®** Software Tool the **GRAVITY ABSOLUTE** command is specified, then the governing pressure equation is solved in terms of pressure head, P; otherwise the governing equation is solved in terms of the total head, H. The boundary and initial conditions may be specified in terms of either P or H. For unsaturated (multi-phase, Vadose or free surface) flow, the gravitational force is always included (**GRAVITY RELATIVE**) by default. For saturated flow, if no **GRAVITY** command is specified, then the body force is neglected and the P and H are equivalent.

For the **TIDAL®** Software Tool, only the absolute value appears in the governing equations since the equations are integrated in the vertical direction (assumed to be opposed to the direction of the gravity vector). Therefore only a single value (**g<sub>x</sub>**) may be specified; the absolute value (**g**) will be set equal to this value since the other values are assumed to be zero.

**REMARKS**

For the **ANSWER®** Software Tool, this command triggers the inclusion of the gravitational body force in computations. As such, **it must be present for computations where this body force is present**. The most common example is that of natural convection. Of course for natural convection to occur, the density of the fluid must vary due to thermal and/or other effects. Otherwise a uniform body force merely acts to alter the datum for pressure.

For the **PORFLOW®** Software Tool, if flow is single-phase and the density is uniform, then the buoyancy term,  $B_j$ , is uniform and constant. The net buoyancy contribution is then zero because it is the gradient of the buoyancy term that appears in the pressure equation. In this instance, the gravitational components may be set equal to zero. It should be noted that these conditions lead to the pressure head, P, being identical to the total head, H.

**EXAMPLES**

**GRAV**ity constants are: 0., -9.81 **RELATIVE** mode ! Orientation against y-axis  
**GRAV**ity constants are: -6.937, -6.937, 0. **ABSOLUTE** mode ! 45 deg to x & y  
**GRAV**ity constants are: 0., 0., -9.81 ! Orientation against z-axis  
**GRAV**ity constants are: 0., 0., -32.2, 32.2 ! Orientation against z-axis  
**GRAV**ity constant is 9.81

## 5.48 GRID

**PURPOSE** To specify the number of elements or grid nodes and the dimension of the problem.

**MODE 1:** **Structured Grid Specification**

**SYNTAX** **GRID** [**NODE|ELEM**] {**N<sub>x</sub>**} [**N<sub>y</sub>**, **N<sub>z</sub>**]

**NODE** The number of nodes is specified. For a structured grid the number of nodes in each direction is equal to the number of elements plus two additional nodes, one at each external boundary. **This is the default option**

**ELEM** **N<sub>x</sub>**, **N<sub>y</sub>** and **N<sub>z</sub>** specify the number of elements in each direction. The number of elements in each direction is 2 less than the number of grid nodes since there is one grid node per element and one boundary node at each end. **The use of this modifier is not recommended; it is being retained for compatibility with old legacy input file.** Some of the new **LOCATE** commands are not compatible with this input mode.

If this modifier is specified then all subregion input specification must be consistent with the indices in terms of elements rather than in terms of nodes. The element indices are numbered one less than the grid node indices. For example, for a 2D problem, the element grid index set (1,1) is equivalent to specifying the grid node (2,2) and so on. The boundary nodes cannot be explicitly specified in this mode; they are automatically captured by a **LOCATE** command if the element next to the boundary is specified.

**N<sub>x</sub>** The number of grid elements or grid nodes in the x direction. The number of nodes is denoted by **JMAX** in this document. **The default value is 1 element or 3 nodes.**

**N<sub>y</sub>** The number of grid elements or nodes in the y or r direction. The number of nodes is denoted by **JMAX**. **The default value is 1 element.**

**N<sub>z</sub>** The number of grid elements or nodes in the z or  $\theta$  direction. The specified value is denoted by **KMAX**. If the specified number of elements is 1 or more, then the 3D mode is activated. If a 0 or negative value is explicitly specified, then the 2D solution mode is invoked but it is assumed that all subregion input specification is in the 3D mode. If **N<sub>z</sub>** is not specified, then the 2D solution mode is invoked and it is assumed that all subregion input specification is also in the 2D mode. **There is no default value for this input.**

**EXAMPLES**

**GRID** is 32!

**GRID** is 32 by 27 by 1 NODEs!

**GRID** is 32 by 27 by 12 NODEs!

1D problem: **JMAX** automatically set to 3

2D problem: grid related input in 3D mode

3D problem: grid related input in 3D mode



**MODE 3:**      **Grid Less Computations**

**SYNTAX**      **GRID {NONE} {fname}**

**NONE**          The computations are performed in ACRi's revolutionary Virtual Finite Volume (VFM) method which consists of computation from a collection of arbitrary points in the computational domain. No grid is necessary.

**fname**         The name of the file that contains the locations of node points and other pertinent information. This file format is internal to ACRi and the file is generated by software provided by ACRi.

#### **EXAMPLES**

---

**GRID NONE** the data is on file "GRIDLESS.FIL"

**MODE 4:**      **Unstructured Grid Directly from Connectivity File**

**SYNTAX**      **GRID {CONN} [VERTEX|HYBRID] {fname} [THRE]**

**CONN**            Grid and connectivity is specified in an unstructured mode. In this mode, the connectivity data is read from the specified file and the number of field elements is automatically determined from the input. A separate **CONNECTIVITY** command is not required. The coordinates of the vertices must be specified on a file through the **COORDINATE** command.

**VERTEX**        The **VERTEX** mode of the **CONNECTIVITY** command is invoked. Please see the **CONNECTIVITY** command for further details. **This is the default option.**

**HYBRID**        The **HYBRID** mode of the **CONNECTIVITY** command is invoked. Please see the **CONNECTIVITY** command for further details.

**fname**           The name of the file containing the numerical input for the connectivity. **There is no default value; a file name must be specified.**

**THRE**            **By default the problem is assumed to be a 2D problem.** This modifier must be used to specify a 3D problem.

#### **EXAMPLES**

---

**GRID** CONNectivity of VERTices to elements on file "VERT2ELM.CNC"

**GRID** CONNectivity for HYBRID elements on file "MIXED\_ELEMENTS.CNC" THREE D problem

## 5.49 HISTORY

<b>PURPOSE</b>	To obtain output of the time history for dependent variables at selected nodes.
<b>MODE 1:</b>	<b>Specification of Time History For a Set of Variables in a Subregion</b>
<b>SYNTAX</b>	<b>HIST {subrgn} [<math>\Phi</math>] {fname} [TABL] [PLOT] [NOW   OFF   ON] [COMP] [<math>V_{Frq}</math>] [TIME]</b>
<b>subrgn</b>	The subregion for which the time history is required. See Section 3.4. For this mode to be invoked a subregion previously defined by a <b>LOCATE</b> command must be specified with an <b>ID=subrgn</b> or a <b>SELECT/LOCATE</b> modifier. There is no default value.
<b><math>\Phi</math></b>	One or more symbols that denote the variables for which the time history output is required. The valid symbols are listed in Table 2.8.1-3. By default, the output is obtained for all variables for which the equations are solved plus other variables that are of interest to the problem being solved. The variables can be specified only once for each time history command; a new command must be specified if a different set of variables is to be selected.
<b>fname</b>	The file name for time history output. See Section 3.3 for additional information. <b>Except for the 1<sup>st</sup> command, a file name must be specified with each command.</b> The default file name for the 1 <sup>st</sup> command is "xxx_HISTORY.TMP", where xxx is the Standard Output file without the extension. For example, if the Standard Output file is " <b>PROBLEM.OUT</b> ", then the file is named " <b>PROBLEM_HISTORY.TMP</b> ". To avoid confusion, it is recommended that a file name be specified with each command.
<b>TABL</b>	The time history data are printed in a tabular form at the end of simulations.
<b>PLOT</b>	A printer plot of time history of variables is generated at the end of simulations.
<b>NOW</b>	One record of time history output is produced immediately.
<b>OFF</b>	Time history output is discontinued.
<b>ON</b>	Time history output is resumed if it was previously suppressed.
<b>COMP</b>	By default the time history output file contains the project title and information about grid. If this modifier is specified then this information is omitted.
<b><math>V_{Frq}</math></b>	The output frequency (step or time interval) for writing to the output device. See Section 3.7 for further details. By default the time history is generated at every step.
<b>TIME</b>	By default, <b><math>V_{Frq}</math></b> is interpreted as the frequency of output in terms of number of steps. If this modifier is present, then <b><math>V_{Frq}</math></b> is interpreted as the time interval between successive outputs.

**REMARKS**

**Multiple Mode 1 commands may be specified.** Each command can have its own set of variables and other modifiers and frequency. This command mode is strongly recommended over the Mode 2 and 3 of the **HISTORY** command.

**EXAMPLES**

**HISTORY** for U, V, W only for ID=HIS region on "UVW.HIS" file.  
**HISTORY** for region ID=HISTORY every 20 steps on file 'ALL.HIS'  
**HISTORY** ID=HISTORY at TIME interval of 1.75 hours output to file 'ALL.HIS'  
**HISTORY** ID=HISTORY for U, P, T, K on file 'HISTORY.OUT' every 20 steps  
**HISTORY** ID=HISTORY for U, P on file 'HISTORY.OUT' TIME=2.50 hrs; print TABLEs also  
**HISTORY** ID=HISTORY for U, P, T, K at TIME interval 0.23: print TABLEs also 'UPTK\_HIS.TMP'  
**HISTORY** ID=HISTORY U & T on 'HIS.NEW' at every 20 steps; print TABLEs and PLOTS

**MODE 2:**      **Specification of Location of Time History Nodes****SYNTAX**      **HIST**    **{N<sub>1</sub>, ..., N<sub>n</sub>}** **[fname]**  
N=m in unstructured mode, = 2m in 2D and =3m in 3D mode

**N<sub>1</sub>, ..., N<sub>n</sub>**      The grid index locations for the elements for which the time history output is obtained. For unstructured grids, the input is that of the element numbers. For structured grids the input consists of the grid indices (I, J, K) in the 3D or (I, J) in the 2D mode. **If the grid is unstructured, then at least two elements must be specified; a single number will be interpreted as the frequency (See Mode 3 of command).** If history at a single element is desired in the unstructured mode, then it must be specified through Mode 1 of the command.

**fname**            See Mode 1 Specification.

**REMARKS**

---

This mode of the command works in conjunction with Mode 3. **This mode of HISTORY command is now obsolete and has been superseded by Mode 1 of the command, which provides the combined functionality of both Mode 2 and Mode 3.** It is being retained only temporarily to maintain compatibility with legacy files. It is strongly recommended that Mode 1 be used in preference to this mode of the command.

**If this mode of the HISTORY command is used, then it must precede any Mode 1 command.**

**EXAMPLES**

---

**HIST**ory at (2, 2), (2, 5), (5, 2), (11, 17) and (17, 11)! Input in 2D mode  
**HIST**ory at elements 57, 33, 165 and 915! Unstructured mode  
**HIST**ory output OFF at (2, 2) and (2, 5)  
**HIST**ory at node (7, 11) and (7, 15) to be added to previous ones  
**HIST**ory at (2, 2, 2), (2, 5, 7), (5, 2, 7), (11, 17, 19) & (17, 11, 12)! Input in 3D mode  
**HIST**ory OFF at (2, 2, 2) and (2, 5, 7)  
**HIST**ory output again at (2, 2, 2) and (2, 5, 7)

**MODE 3:**      **Specification of Variables and Output Options for Mode 2**

**SYNTAX**      **HIST**    **[Φ]** **[fname]** **[TABL]** **[PLOT]** **[NOW|OFF|ON]** **[COMP]** **[V<sub>Frq</sub>]** **[TIME]**

<b>Φ</b>	See Mode 1 Specification.
<b>fname</b>	See Mode 1 Specification.
<b>TABL</b>	See Mode 1 Specification.
<b>PLOT</b>	See Mode 1 Specification.
<b>NOW</b>	See Mode 1 Specification.
<b>OFF</b>	See Mode 1 Specification.
<b>ON</b>	See Mode 1 Specification.
<b>COMP</b>	See Mode 1 Specification.
<b>V<sub>Frq</sub></b>	See Mode 1 Specification.
<b>TIME</b>	See Mode 1 Specification.

#### REMARKS

---

This mode of the command works in conjunction with Mode 2. **This mode of HISTORY command is now obsolete and has been superceded by Mode 1 of the command, which provides the combined functionality of both Mode 2 and Mode 3.** It is being retained only temporarily to maintain compatibility with legacy files. It is strongly recommended that Mode 1 be used in preference to this mode of the command.

**If this mode of the HISTORY command is used, then it must precede any Mode 1 command.**

#### EXAMPLES

---

**HIST**ory on file 'HISTORY.TIM'

**HIST**ory for U, V, W only

**HIST**ory every 20 steps

**HIST**ory at TIME interval of 1.75 hours

**HIST**ory for U, P, T, K on file 'HISTORY.OUT' every 20 steps

**HIST**ory for U, P on file 'HISTORY.OUT' TIME=2.50 hrs; print TABLEs also

**HIST**ory for U, P, T, K at TIME interval 0.23: print TABLEs also

**HIST**ory U & T on 'HIS.NEW' at every 20 steps; print TABLEs

**HIST**ory U & T on 'HIS.NEW' at every 20 steps; print TABLEs and PLOTs

<b>MODE 4:</b>	<b>History at an Arbitrary Point in Space</b>
<b>SYNTAX</b>	<b>HIST {COORD} { <math>\Phi</math> } [fname] [TABL] [LINE SQUA] [N1,..Nn] [VFrq] [TIME]</b>
<b>COORD</b>	History is required at an arbitrary point in space specified by its (x, y, z) coordinates.
<b><math>\Phi</math></b>	<b>One or more</b> symbols for the variables for which the history is required. <b>No more than 10 symbols can be specified on one command.</b> The valid symbols are listed in Table 2.7.1. <b>There is no default value.</b>
<b>fname</b>	The file name for output. <b>A different output file may be specified for each command.</b> If no file name is specified then the output is directed to the most recent previously specified file. If no previous file was specified, then the output is directed to a file with the same name as the Standard Output file but with the extension removed and replaced by “ <b>_HISTORY_XYZ.TMP</b> ”. For example, if the Standard Output file is “ <b>PROBLEM.OUT</b> ”, then the file is named “ <b>PROBLEM_ HISTORY_XYZ.TMP</b> ”. A summary of output is also printed to the standard output device at the end of simulations.
<b>TABL</b>	The time history data are printed in a tabular form at the end of simulations.
<b>LINE</b>	The values at the specified (x, y, z) location are computed by linear interpolation from its nearest neighbors. <b>This option is active by default.</b>
<b>SQUA</b>	The values at the specified (x, y, z) location are computed by inverse distance squared interpolation from its nearest neighbors.
<b>N1, ..., Nn</b>	The grid coordinates (x, y, z) of the point. Two numerical values for 2D and three for 3D input modes must be provided. <b>There is no default value for this input; the required number of numerical values must be specified.</b>
<b>VFrq</b>	The frequency (step or time interval) at which the output is written to the output device. See Section 3.7 for further details. <b>If present, this must be last value on the command.</b> <b>By default the time history is generated at every step.</b>
<b>TIME</b>	<b>By default, VFrq is interpreted to be the frequency in terms of number of steps.</b> If this modifier is present, then <b>VFrq</b> is interpreted to be the time interval between successive outputs.

**REMARKS**

This command generates output of history of values at an arbitrary location in space, which may or may not coincide with the computational nodes. The history of values at computational nodes can be obtained by Mode 1 specification.

**EXAMPLES**

**HISTORY** of U at point with COORdinate x=1.35, y=0.796 every 3 steps  
**HISTORY** of U at point with COORdinate x=1.35, y=0.796 every 20 steps with inverse SQUAre interpolation  
**HISTORY** of U, V, W, T at COORdinate x=1.35, y=0.796, z=0.0975 at TIME interval of 1.75 units  
**HISTORY** of U, T, C at COORdinate (1.35, 0.796, 0.0975) TIME interval 0.235 on 'HISTORY.XYZ'  
**HISTORY** U, T, C COOR (1.35, 0.796, 0.0975) TIME interval 0.235; SQUARE method 'HISTORY.XYZ'

**MODE 5:** History of Source for a Variable**SYNTAX** HIST {SOUR} {Φ} [subrgn] [fname] [TABL] [OFF] [V<sub>Frq</sub>] [TIME]**SOUR** Output for the source term for Φ is required.

Φ The symbol for the variable for which the source inventory is required. Only those symbols may be specified for which a differential equation is solved. The valid symbols are listed in Table 2.7.1. There is no default value.

**subrgn** The subregion for computations. If no subregion is specified, the entire domain is selected.**fname** The file name for output. A different output file may be specified for each command. If no file name is specified then the output is directed to the most recent previously specified file. If no previous file was specified, then the output is directed to a file with the same name as the Standard Output file but with the extension removed and replaced by “\_SOURCE.TMP”. For example, if the Standard Output file is “PROBLEM.OUT”, then the file is named “PROBLEM\_SOURCE.TMP”. A summary of output is also printed to the standard output device at the end of simulations.**TABL** The time history data are printed in a tabular form at the end of simulations.**OFF** Any previously specified command for the specified Φ and subrgn is deactivated.**V<sub>Frq</sub>** The frequency (step or time interval) at which the output is written to the output device. See Section 3.7 for further details. By default the time history is generated at every step.**TIME** By default, V<sub>Frq</sub> is interpreted to be the frequency in terms of number of steps. If this modifier is present, then V<sub>Frq</sub> is interpreted to be the time interval between successive outputs.**REMARKS**

This command generates output of the history of “source” term for the variable. This includes all source (or sink) terms for the variable including the terms originally present in the governing equation (e.g. pressure gradient terms for momentum equations for ANSWER®) and those specified by the user through the various SOURce, DEcAY and REAcTion commands. The output is the integral of the source for the variable Φ over the volume of the subrgn.

The output is printed to the file specified by fname. In addition, at the end of simulations, tables of output are printed to the standard output file. These tables are sorted by variable and subregion.

**EXAMPLES****HISTory** SOUR or C for the entire domain**HISTory** for SOURce of C for the entire domain**HISTory** of SOURCE for T for SELEcted region every 20 steps**HISTory** of SOURce for T for subregion defined by ID=VAULt every 200 steps**HISTory** of SOURCE for T for ID=VAULt OFF**HISTory** of SOURCE for Ton 'SOURCE.OUT' for SELEcted region at TIME interval of 0.4

**MODE 6:** History of Mass or Property Inventory for a Variable

**SYNTAX** HIST {STOR} { $\Phi$ } [subrgn] [fname] [TABL] [OFF] [V<sub>Frq</sub>] [TIME]

**STOR** Output for the storage or accumulation term for  $\Phi$  is required.

**$\Phi$**  The symbol for the variable for which the inventory is required. Only those symbols may be specified for which a differential equation is solved. The valid symbols are listed in Table 2.7.1. There is no default value.

**subrgn** The subregion for computations. If no subregion is specified, the entire domain is selected.

**fname** The file name for output. A different output file may be specified for each command. If no file name is specified then the output is directed to the most recent previously specified file. If no previous file was specified, then the output is directed to a file with the same name as the Standard Output file but with the extension removed and replaced by “\_STORAGE.TMP”. For example, if the Standard Output file is “PROBLEM.OUT”, then the file is named “PROBLEM\_STORAGE.TMP”. A summary of output is also printed to the standard output device at the end of simulations.

**TABL** The time history data are printed in a tabular form at the end of simulations.

**OFF** Any previously specified command for the specified  $\Phi$  and subrgn is deactivated.

**V<sub>Frq</sub>** The frequency (step or time interval) at which the output is written to the output device. See Section 3.7 for further details. By default the time history is generated at every step.

**TIME** By default, V<sub>Frq</sub> is interpreted to be the frequency in terms of number of steps. If this modifier is present, then V<sub>Frq</sub> is interpreted to be the time interval between successive outputs.

#### REMARKS

---

This command generates output of the history of “accumulation” or “storage” term for the variable. The output is the integral of the property represented by the variable  $\Phi$  over the volume of the subrgn.

The output is printed to the file specified by fname. In addition, at the end of simulations, tables of output are printed to the standard output file. These tables are sorted by variable and subregion.

#### EXAMPLES

---

HIST STOR for C for the entire domain

HISTory of STORAge for T for SELEcted region every 20 steps

HISTory of STORAge for T for subregion defined by ID=VAULT every 200 steps

HISTory of STORAge balance for T for ID=VAULT OFF

HISTory of STORAge for T on “STORAGE.OUT” for ID=REGION1 at TIME interval of 0.4

<b>MODE 7:</b>	<b>History of Flux for a Variable</b>
<b>SYNTAX</b>	<b>HIST {FLUX} {Φ} [subrgn] [fname] [COMP] [OFF] [V<sub>Frq</sub>] [TIME]</b>
<b>SOUR</b>	Output for the flux term for $\Phi$ is required.
<b>Φ</b>	The symbol for the variable for which the flux history is required. Only those symbols may be specified for which a differential equation is solved. The valid symbols are listed in Table 2.7.1. There is no default value.
<b>subrgn</b>	The subregion for computations. The subregion specified must define a boundary. If no subregion is specified, the entire domain is selected.
<b>fname</b>	The file name for output. If no file name is specified then the output is directed to the standard output file. If a file name is specified, it must be distinct from any other previous file name.
<b>COMP</b>	By default a header is printed out for each file. This header contains the information about the variable and the subdomain that is output. If this modifier is present, then the header is omitted.
<b>OFF</b>	Any previously specified command for the specified $\Phi$ and <b>subrgn</b> is deactivated.
<b>V<sub>Frq</sub></b>	The frequency (step or time interval) at which the output is written to the output device. See Section 3.7 for further details. By default the time history is generated at every step.
<b>TIME</b>	By default, <b>V<sub>Frq</sub></b> is interpreted to be the frequency in terms of number of steps. If this modifier is present, then <b>V<sub>Frq</sub></b> is interpreted to be the time interval between successive outputs.

**REMARKS**

This command generates output of the history of the total (convective and diffusive) flux term for the variable entering or leaving the boundaries of the specified subdomain.

**EXAMPLES**

**HISTory** FLUX or C for the entire domain  
**HISTory** of FLUX for T for SELEcted region every 20 steps  
**HISTory** of FLUX for T for subregion defined by ID=VAULt every 200 steps  
**HISTory** of FLUX for T for ID=VAULt OFF  
**HISTory** of FLUX for Ton FLUX.OUT' for SELEcted region at TIME interval of 0.4

5.50 HYDRAULIC

**PURPOSE** To specify the reference hydraulic properties of the host porous matrix or of the planar or linear features. This command is effective only for the **PORFLOW®** Software Tool.

**SYNTAX** **HYDR** { $\alpha_s$ } [**K<sub>x</sub>**, **K<sub>y</sub>**, **K<sub>z</sub>**] [**subrgn**]

$\alpha_s$  The matrix compressibility ( $\geq 0$ ) with a default value of 0. This input is used to compute the effective storativity,  $S_e$  as defined by:

$$S_e^n = R^n S^n \alpha_s + R^n \theta_E (S^n \alpha_f^n + \alpha_{Sat}^n)$$

$\alpha_s$ , the compressibility of the matrix, is assumed to be a constant and obtained from empirical data. The superscript ‘n’ denotes the n<sup>th</sup> fluid phase, R is the density ratio, S is the saturation,  $\alpha_f$ , is the fluid compressibility and  $\alpha_{Sat}$  is the compressibility due to change of saturation with pressure. The  $\alpha_f$  can be specified with the **FLUID COMPRESSIBILITY** command and the  $\alpha_{sat}$  is computed automatically from the soil characteristic specified by the **MULTIphase** command. For a liquid,  $\alpha_f$  is generally small and is often ignored.

**K<sub>x</sub>, K<sub>y</sub>, K<sub>z</sub>** The reference values ( $\geq 0$ ) of the three diagonal components of the hydraulic conductivity tensor,  $K_{ij}$  in the three principal directions, (x, y, z) or (x, r,  $\theta$ ), respectively. All off-diagonal components ( $i \neq j$ ) are assumed to be zero. The default value is 0.

$$K_{ij} = k_{ij} \rho^* g / \mu^*$$

In this relation, k is the relative permeability, and  $K_j$  are the reference values of the hydraulic conductivity tensor for a reference fluid (usually water) of density  $\rho^*$  and viscosity  $\mu^*$ . The  $K_j$  is specified from field data for a given soil or rock formation,

**subrgn** The subregion for which the input is specified. See Section 3.4. If no subregion is specified, the entire computational domain is selected.

**REMARKS**

More complex functional relations for storativity can be specified by the **STORAGE** or **DISTRIBUTION** commands and those for hydraulic conductivity can be specified by the **CONDUCTIVITY** command.

**EXAMPLES**

**HYDRAULIC** properties: ss = 0.2,  $K_x^* = 2$ ,  $K_y^* = 0.2$ ,  $K_z^* = 0.2$  ft per day

**HYDRAULIC** properties: ss = 0.2,  $K_x^* = 2$ ,  $K_y^* = 0.2$ ,  $K_z^* = 0.2$  ft per day for **ID=Clayey\_Sand**

## 5.51 IF-ELSE-ENDIF

**PURPOSE** To define an IF construct to control which, if any, of one or two blocks of user input statements are executed.

**SYNTAX** **IF** {variable} = {Ψ|V} then  
           1<sup>st</sup> block of FREEFORM statements  
**ELSE**  
           2<sup>nd</sup> block of FREEFORM statements]  
**ENDIF**

**variable** A symbolic variable that at run-time is compared to the value assigned to Ψ or V. The 'variable' must be the 1<sup>st</sup> modifier following the IF command. Only the first 8 characters are meaningful; any subsequent characters are ignored. The DEFINE command provides a powerful way to define such variables and may be used to control the IF-construct. The input for the variable must be symbolic if Ψ is specified and numeric if V is specified.

**Ψ** The character string that is compared to the 'variable' to determine the status of the IF construct as true or false. If the status is true then the 1<sup>st</sup> block of statements is executed and the 2<sup>nd</sup> block, if present, is ignored. If the status is false then the 1<sup>st</sup> block of statements is ignored and the 2<sup>nd</sup> block, if present, is executed. This modifier, if it exists, must immediately follow the modifier 'variable' without any other intervening character strings. Only the first 8 characters are significant; any subsequent characters are ignored.

**V** The numerical value that is compared to the 'variable' to determine if the status of the IF construct is true or false. If the status is true then the 1<sup>st</sup> block of statements is executed and the 2<sup>nd</sup> block, if present, is ignored. If the status is false then the 1<sup>st</sup> block of statements is ignored and the 2<sup>nd</sup> block, if present, is executed. If both V and Ψ are specified, then V input takes precedence.

**REMARKS**

This command provides a powerful means of performing conditional branching with ACRi FREEFORM™ commands. The IF construct can be used to consolidate input data sets. This construct, together with the DEFINE commands can be used for creating 'prototype' data input files for parametric or multivariate studies.

Each block of FREEFORM™ statement is composed of zero or more statements. The presence of the 2<sup>nd</sup> block is optional. However, if the 2<sup>nd</sup> block is present, then it must be preceded by an ELSE command. If the ELSE statement is present, then exactly one of the two blocks is executed. The IF construct must be completed by an ENDIF command otherwise unforeseen consequences may arise. The ENDIF statement must appear as stated. For example, the statement END IF will be misinterpreted as an END statement.

Unless an explicit INDENT ON or INDENT OFF command has been previously given, the IF command automatically triggers the onset of the INDENT ON command and the ENDIF command automatically triggers the INDENT OFF command. Please see the INDENT command.

**EXAMPLES**

---

**DEFINE** GEOM\_3D = 3! This statement defines a numerical value for variable GEOM\_3D  
**IF** (GEOM\_3D = 2) THEN  
**GRID** = 20 BY 20 BY 20! This block will be ignored because the IF status is false  
**ELSE**  
**GRID** = 20 BY 20! This block will be executed  
**ENDIF**

**DEFINE** YESNO = OUTPUT  
**IF** (OUTPUT = YESNO) THEN  
**OUTPUT** U, V, W and T! This block will be executed because the IF status is true  
**ELSE**  
**OUTPUT** OFF! This block will be ignored  
**ENDIF**

**DEFINE** YESNO = NOOUTPUT  
**IF** (OUTPUT = YESNO) THEN  
**OUTPUT** OF U, V, W and T  
**ENDIF**

**5.52 INCLUDE**

**PURPOSE** To include a file at run time in the input data stream.

**SYNTAX** **INCL** [**fname**]

**fname** The name of the file which is to be included in the current input data file. The specified file must be compatible with the **FREEFORM™** language. It must consist of valid keyword commands and input data only. The **INCLUDE** command provides for unlimited nesting in that the file being incorporated may itself contain embedded **INCLUDE** commands. For this particular command the file name need not be included in single or double quotes.

If the user wants to specify a file name on-the-fly at run time, then a question mark (“?”) can be inserted anywhere on the command line. In this case, the user will be prompted for the file name at run time. The filename must be explicitly specified on the command; it cannot be specified through a **DEFINE** command.

**REMARKS**

---

In one important aspect, the **INCLUDE** keyword is treated differently than other **FREEFORM™** keywords commands. This keyword does not have to start in the 1<sup>st</sup> column (character) of a command line. That is **INCLUDE** command can be indented, if desired. **INCL** must, of course, be the 1<sup>st</sup> four non-blank characters on the command. This is different from all other keyword commands, which must start with the 1<sup>st</sup> character (column) unless an **INDENT** command was previously given.

**EXAMPLES**

---

**INCLUDE** file '**INCLUDE.NOW**'

**INCLUDE** commands from file **INCLUDE.NOW**

**INCLUDE** ? ! User will be prompted

5.53 INDENT

**PURPOSE** To allow **FREEFORM™** commands to be indented.

**SYNTAX** **INDE** [**ON|OFF**]

**ON** Ordinarily, the **FREEFORM™** input commands are assumed to start with the 1<sup>st</sup> character of the input command even it is a blank character. Any leading blanks are considered to be significant in processing the input (please see Appendix B for a general description of the **FREEFORM™** command syntax). If the **ON** modifier is present, then any leading blanks are ignored and the meaningful input is assumed to start with the 1<sup>st</sup> non-blank character of the input. **This is the default option; i.e. if no modifier is specified, then the ON modifier is assumed.**

**OFF** Any previous **INDENT** command is deactivated; new **INDENT** commands may follow.

**REMARKS**

An **INDENT ON** is automatically triggered if an **IF** command is encountered. If this is not desired, then **INDENT OFF** command must explicitly precede the **IF** command. Similarly an **INDENT OFF** command is automatically triggered by an **ENDIF** command unless previously an **INDENT** command was explicitly specified.

**When INDENT is ON care should be exercised in specifying commands with continuation lines.** In this case, a continuation line can not start with the first non-blank character as an alphabetic character (A through Z, in upper or lower case) otherwise the leading blanks will be ignored and the “continuation” command will be interpreted as a “Keyword” command. It is a safe practice to start the continuation lines with numeric or special non-alpha characters. **A recommended practice is to always start a continuation command with an ampersand (&) character.** For example, consider the sequence of commands:

```

INDENT ON from now
    LOCATE ID=ZONE1 COORdinate from (0.0, 0.0, 0.0)
    &
    3.0, 2.0) to (1.0,
INDENT OFF
    
```

Here, the “&” character in the 3<sup>rd</sup> line ensures that this input will be treated as a continuation line. If the “&” is removed, then the string “to” will be incorrectly interpreted as the keyword command “TO”. Since this keyword is invalid, the execution will be terminated.

**EXAMPLES**

```

INDEnt from now on ! Any leading blanks in commands will be ignored
    IF (GEOM_3D = 2) THEN
        GRID = 20 BY 20 BY 20
    ELSE
        GRID = 20 BY 20
    ENDIF
INDEnt OFF starting with next command ! Any leading blanks will not be ignored from now on
    
```

## 5.54 INITIAL

**PURPOSE** To specify the initial conditions for dependent variables for structured grids

**SYNTAX** **INIT** { $\Phi = \Phi_o$ } [ $N_1, \dots, N_n$ ]

$\Phi$  The symbol that denotes the field variable for which input is provided. The valid symbols are listed in Table 2.8.1-3. **One, and only one, symbol must be specified. There is no default value.**

$\Phi_o$  The initial value for the variable.

$N_1, \dots, N_n$  These values define the subregion to which the input is applied. In the default mode, the subregion is defined by 4 values for the 2D and 6 for the 3D input mode. **If only one value, that is  $N_1$ , is specified, then it is assumed that  $N_1$  is the material or zone number to which the input applies. If this input is completely omitted, then the subregion is defined to be the entire domain.**

**COMMENT**

This command is available only for structured grids. **SET** command is a much more powerful option for setting field values of the variables for structured or unstructured grids.

**EXAMPLES**

**INIT**ial U is 0.1 everywhere

**INIT**ial T is 1.E-3 from (2, 2) to (7, 9) ! 2D input mode

**INIT**ial T is 1.E-3 from (2, 2, 2) to (7, 9, 4) ! 3D input mode

**INIT**ial T is 1.E-3 for zone 5

## 5.55 INJECTION

**PURPOSE** To specify properties and parameters relating to injection of water films curtains or to condensation. This command is effective only for the ANSWER® Software Tool.

**MODE 1** Location and Properties for Water Films

**SYNTAX** **INJE** {**FILM**} [**subrgn**] {**TINJ**= $V_{Tinj}$ } {**QINJ**= $V_{Qinj}$ } [**TIME**] [**TEMP**] [**COOR**= $V_x$   $V_y$  [ $V_z$ ]  
[**STAR**= $V_{Start}$ |**STAR NOW**][**STOP**= $V_{Stop}$ |**STOP NOW**] [**HIST**= $V_{Hist}$  [**fname**]]

**subrgn** The subregion for which the input is specified. This subregion must be defined with **LOCATE INJECTION** command. If this subregion is a boundary of the domain, then the temperature must be fixed or the boundary must be adiabatic. If this subregion is a boundary of a solid body, then conduction heat exchange may be taken into account. See Section 3.4 for additional details.

**TINJ** Introduces water film temperature.

$V_{Tinj}$  Water film temperature (in Kelvin). There is no default value.

**QINJ** Introduces the mass rate of water in the film.

$V_{Qinj}$  Mass rate of water in the film. There is no default value; a positive value must be specified.

**TIME** By defaults, values of  $V_{Start}$ ,  $V_{Elaps}$ ,  $V_{Stop}$  and  $V_{Hist}$  (see below) are in time step number. If this modifier is present, then values are in time units.

**TEMP** By default, the value of  $V_{Start}$ . (see below) is a step or a time value. If this modifier is present,  $V_{Start}$  is a temperature for which the water film starts to be active.

**COOR** If modifier **TEMP** is present then this keyword must be added to define coordinates of the point at which film temperature monitored.

**NOW** By default, keywords **START** and **STOP** must be followed by values. If **NOW** is set with **START/STOP**, the water film is activated/stopped at the current step.

**STAR** Introduces the start time of the film. By default this is defined by **INJE FREQ** for all injections.

$V_{Start}$  Step, Time (modifier **TIME**) or Temperature (modifier **TEMPERATURE**) value.

**STOP** Introduce the film end time. By default this is defined by **INJE FREQ** for all injections.

$V_{Stop}$  Step or Time (if modifier **TIME** is present) value.

$V_x, V_y, V_z$  Coordinate values at which temperature is monitored. There is no default value.

**HIST** By default no history file of the water film is written. In the presence of this keyword, an history file is computed.

$V_{Hist}$  Step or Time (if modifier **TIME** is present) frequency of history computation.

**fname** The file name for history output. A different output file may be specified for each command. If no file name is specified then the output is directed to the most recent previously specified file. If no previous file was specified, then the output is directed to a file with the same name as the Standard Output file but with the extension removed and replaced by “\_INJ\_FILM.TMP”. For example, if the Standard Output file is “PROBLEM.OUT”, then the file is named “PROBLEM\_INJ\_FILM.TMP”.

**EXAMPLES**

**INJECTION** **FILM** ID=FILM, **TINJECTION**=280. Kelvin, **QINJ**=10 kg/s, **HISTORY** every 1 step, ‘FILM.HIS’  
**INJECTION** **FILM** ID=FILM **START** **TEMP** is 340 Kelvin with **COORDINATES** (2.00,3.00,2.00)

**MODE 2**      **Location and Properties for Water Curtains**

**SYNTAX**      **INJE** {CURTAIN} [ID=subrgn] {QINJ=V<sub>Qinj</sub>} [DIAM=V<sub>Diam</sub>] [TIME] [TEMP]  
 [STAR=V<sub>Start</sub>STAR NOW] [COOR=V<sub>x</sub>, V<sub>y</sub> [V<sub>z</sub>]] [STOP=V<sub>Stop</sub>STOP NOW]  
 [HIST=V<sub>Hist</sub> [fname]]

<b>subrgn</b>	The subregion for which the input is specified. <b>This subregion must be defined with LOCATE INJECTION command.</b> See Section 3.4 for additional details.
<b>QINJ</b>	Introduces the mass rate of water in the film. <b>There is no default value; a positive value must be specified.</b>
<b>V<sub>Qinj</sub></b>	Mass rate of water in the film
<b>DIAM</b>	Introduces droplet diameter of water curtain
<b>V<sub>Diam</sub></b>	Droplet diameter. <b>There is no default value; a positive value must be specified.</b>
<b>TIME</b>	By defaults, values of <b>V<sub>Start</sub></b> , <b>V<sub>Elaps</sub></b> , <b>V<sub>Stop</sub></b> and <b>V<sub>Hist</sub></b> (see below) are in time step number. If this modifier is present, then values are in time units.
<b>TEMP</b>	By default, the value of <b>V<sub>Start</sub></b> , (see below) is a step or a time value. If this modifier is present, <b>V<sub>Start</sub></b> is a temperature for which the water film starts to be active.
<b>NOW</b>	By default, keywords <b>START</b> and <b>STOP</b> must be followed by values. If <b>NOW</b> is set with <b>START /STOP</b> , the water film is activated/stopped at the current step.
<b>STAR</b>	Introduces the start time of the film. <b>By default this is defined by INJE FREQ for all injections.</b>
<b>V<sub>Start</sub></b>	Step or Time (if modifier <b>TIME</b> is present) or Temperature (if modifier <b>TEMP</b> is present) value.
<b>STOP</b>	Introduces the film end time. <b>By default this is defined by INJE FREQ for all injections</b>
<b>V<sub>Stop</sub></b>	Step or Time (if modifier <b>TIME</b> is present) value.
<b>COOR</b>	If modifier <b>TEMP</b> is present then this keyword must be added to define coordinates of the point at which film temperature monitored.
<b>V<sub>x</sub>, V<sub>y</sub>, V<sub>z</sub></b>	Coordinate values at which temperature is monitored. <b>There is no default value.</b>
<b>HIST</b>	<b>By default no history file of the water film is written.</b> In the presence of this keyword, a history file is written.
<b>V<sub>Hist</sub></b>	Step or Time (if modifier <b>TIME</b> is present) frequency of history computation.
<b>fname</b>	The file name for history output. <b>A different output file may be specified for each command.</b> If no file name is specified then the output is directed to the most recent previously specified file. If no previous file was specified, then the output is directed to a file with the same name as the Standard Output file but with the extension removed and replaced by <b>"_INJ_CURTAIN.TMP"</b> . For example, if the Standard Output file is <b>"PROBLEM.OUT"</b> , then the file is named <b>"PROBLEM_INJ_CURTAIN.TMP"</b> .

**EXAMPLES**

**INJE** CURTain ID=ZCURT QINJection=10. kg/s DIAMeter=0.001 m  
 START NOW and STOP at TIME 180 seconds  
 HISTory every 0 second, file is 'CURTain.his'

<b>MODE 3</b>	<b>Location and Properties for Condensation</b>
<b>SYNTAX</b>	<b>INJE</b> {COND} {subrgn} [dir] [DROP   FILM] [HIST=V <sub>Hist</sub> ] [TIME] [fname]
<b>COND</b>	The condensation of the injected films or droplets is allowed to take place.
<b>subrgn</b>	The subregion for which the input is specified. See Section 3.4.
<b>dir</b>	The orientation index for the boundary at which the condensation occurs. See Section 3.5 for available choices.
<b>DROP</b>	The droplet condensation model is activated. <b>This is the default option.</b>
<b>FILM</b>	The film condensation model is activated.
<b>HIST</b>	By default no history file of the condensation is given. In the presence of this keyword, a history file is generated.
<b>V<sub>Hist</sub></b>	Step or Time (if modifier <b>TIME</b> is present) frequency of history computation.
<b>TIME</b>	By default <b>V<sub>Hist</sub></b> is a frequency in terms of number of steps. If this modifier is present, then the value is taken to be the frequency in time units.
<b>fname</b>	The file name for history output. A different output file may be specified for each command. If no file name is specified then the output is directed to the most recent previously specified file. If no previous file was specified, then the output is directed to a file with the same name as the Standard Output file but with the extension removed and replaced by “_INJ_CONDENSE.TMP”. For example, if the Standard Output file is “PROBLEM.OUT”, then the file is named “PROBLEM_INJ_CONDENSE.TMP”.

#### REMARKS

---

If **FILM** model is activated, the water film falling direction is defined by the direction of gravity. See the **GRAVITY** command for further information.

#### EXAMPLES

---

**INJE**ction CONDensation ID=NORD HISTory every 1 step, file is 'COND\_NORD.his'  
**INJE**ction COND ID=NORD HISTory TIME every 0.1 second, file is 'COND\_NORD.his'

**MODE 4:** Injection Computations Parameters**SYNTAX** INJE {BALA} {V<sub>frq</sub>} [TIME] [FILM, CURT, COND, NOZZ] [fname]**BALA** The balance summary of the instantaneous and cumulative amount in liquid and vapor phases due to liquid injection is output to a file. See **REMARKS** below.**V<sub>frq</sub>** The frequency (step or time interval) at which the balance is written to the file. The default is 1.**TIME** By default, **V<sub>frq</sub>** is the frequency of output in terms of number of steps. If this modifier is present, then **V<sub>frq</sub>** is interpreted to be the time interval.**FILM** The amount due to **INJECTION FILM** commands is included.**CURT** The amount due to **INJECTION CURTAIN** commands is included.**COND** The amount due to **INJECTION CONDENSE** commands is included.**NOZZ** The amount due to **NOZZLE** commands is included.**fname** The file name for balance output. A different output file may be specified for each command. If no file name is specified then the output is directed to the most recent previously specified file. If no previous file was specified, then the output is directed to a file with the same name as the Standard Output file but with the extension removed and replaced by “\_INJ\_BALANCE.TMP”. For example, if the Standard Output file is “**PROBLEM.OUT**”, then the file is named “**PROBLEM\_INJ\_BALANCE.TMP**”.**REMARKS**

---

If none of the modifiers, **FILM**, **CURT**, **COND** or **NOZZ**, are present, then the balance is output for all **INJECTION** and **NOZZLE** commands.

**EXAMPLES**

---

**INJECTION BALANCE** at step interval of **5** on default file**INJECTION BALANCE** at **TIME** interval of **0.2** on 'PHASE-BALANCE.OUT' file**INJECTION BALANCE** **TIME** interval of **0.2** for **FILM** and **CURTAIN** commands

**MODE 5:** Injection Computations Parameters**SYNTAX** INJE {FREQ} {V<sub>frq</sub>} [TIME] [N<sub>Begin</sub>, N<sub>End</sub>]**FREQ** The spray computation frequency parameters are specified.**V<sub>frq</sub>** The frequency (step or time interval) at which the spray computations are performed. The default value is 10.**TIME** By default, V<sub>frq</sub> is the frequency of computations in terms of number of steps. If this modifier is present, then V<sub>frq</sub> is interpreted to be the time interval.**N<sub>Begin</sub>** The first computational step at which the droplet computations are performed. The default value is 1.**N<sub>End</sub>** The last computational step at which droplet computations is performed. The default frequency is set to 9999999.**REMARKS**

---

Spray droplet computations are automatically triggered if a **INJECTION** command in Mode 1 through 3 is encountered. This input is required only if the user wants to modify the default values.

This command is equivalent to the **NOZZLE FREQUENCY** command and the input on either command will govern both the **NOZZLE** and the **INJECTION** command computations.

**EXAMPLES**

---

**INJECTION FREQUENCY** of calculations: step = 1 start at step number 10

**INJECTION FREQUENCY** of calculations: 10 start at step number 10 and stop at step 500

**INJECTION FREQUENCY TIME** units 2.5 , start at t=0 and stop at t=15.5

**5.56 INLET**

**PURPOSE** To specify an inflow boundary for the domain of computation. This command is effective only for the ANSWER® Software Tools.

**SYNTAX** **INLE** {**dir**} [**subrgn**] [**DIAG**] [**OFF**]

**dir** The orientation index for the inlet boundary. See Section 3.5 for available choices. There is no default value, a value must be specified.

**subrgn** The subregion to be identified as an inlet. If no subregion is specified, the outermost "dir" oriented boundary of the entire computational domain is selected.

**DIAG** By default only summary statistics of total boundary area, average density and velocity, and flow rate are printed to the standard output device. If this modifier is present then a complete detail of area, velocity components, density and flow rate for each segment of the boundary is printed.

**OFF** Previously specified inlet boundary for the identified subregion is deactivated. A new specification for this subregion may follow.

**REMARKS**

---

An inlet boundary, by definition, is assumed to be a boundary where the values of all dependent variables are known. This command provides a compact way to specify fixed boundary conditions for all variables at a given boundary. The actual values at the inlet boundary may be specified by **INITIAL**, **READ** or **SET** commands. These values may be changed by a subsequent command during simulations. The only effect of this command is to keep the values at the inlet boundary fixed at the value set by the user.

**EXAMPLES**

---

**INLE**t at X- boundary at left (minimum x)

**INLE**t at Y- boundary for the most recently SELEcted subregion

**INLE**t at X+ boundary for subregion ID = INFLow

**INLE**t OFF at X+ boundary for subregion ID = INFLow

## 5.57 INTEGRATION

<b>PURPOSE</b>	Choice of discretization scheme for integration of flow, heat and mass transport equations.
<b>SYNTAX</b>	<b>INTE</b> [ $\Phi=V_1, \Phi=V_2, \dots, \Phi=V_n$ ] { <b>HYBR</b>   <b>BOUN</b>   <b>CENT</b>   <b>QUIC</b>   <b>ACRI</b>   <b>UPWI</b> }
$\Phi$	One or more of the symbols that denote the variables for which the integration profile specification is effective. The valid symbols are listed in Table 2.7.1. If no symbol is specified, the input is assumed to be effective for all variables.
$V_1, V_2, \dots, V_n$	The input for the gradient blending factor ( $\beta$ , in <b>BOUNDED</b> scheme). The recommended values are between 0.1 and 1.0. The default value is 0.2. If no value is specified, then any previous specification stays in effect. If only 1 value is specified, then it is applied to all specified variables. If more than 1 value is specified, then the number of values must equal the number of symbols on the command.
<b>HYBR</b>	The hybrid scheme (Runchal, 1972) is used for integration. This is the default option.
<b>BOUN</b>	A bounded NVD central difference scheme is used.
<b>CENT</b>	The 2 <sup>nd</sup> order central-difference scheme is used for integration. The scheme may generate over- and under-shoots if the local grid Peclet number exceeds 2. It is known to be unstable in the linear sense if the local grid Peclet number exceeds 2.
<b>QUIC</b>	A generalized flux-conservative form of the QUICK scheme (Leonard, 1979) is used for integration. This scheme is recommended for use only with hex grids; it should not be used for general polyhedral grids.
<b>ACRI</b>	The stabilized 2 <sup>nd</sup> order central-difference scheme is used for integration. The scheme uses 2 <sup>nd</sup> order central differences for fluxes but the matrix coefficients are controlled to ensure positive-definite property. The scheme may still show over- and under-shoots if the local grid Peclet number is greater than 2; however it generally extends the range of applicability of the central difference scheme.
<b>UPWI</b>	The 1 <sup>st</sup> order upwind scheme is used computing convective fluxes whereas the diffusive fluxes are still computed by the 2 <sup>nd</sup> order central differences. The scheme is unconditionally stable in the linear sense but may increase numerical diffusion under certain conditions.

**REMARKS**

The default option should be adequate for most applications. However, if the local grid Peclet number as given below is significantly larger than 10 and the direction of flow is not (approximately) aligned with any of the coordinate directions, one of the other schemes, except UPWIND, may be desirable. If robustness is required, then UPWIND is the most robust scheme.

$$Pe = U \delta L / \Gamma$$

where  $U$  is the flow speed,  $\delta L$  is the grid interval and  $\Gamma$  is the diffusivity

**EXAMPLES**

**INTEGRATION** for all by **QUICK** scheme  
**INTEGRATION** for **T, C, C2** by **CENTRAL** difference scheme  
**INTEGRATION** for **T, C, C2** by **ACRI** central difference scheme  
**INTEGRATION** for **U** by **BOUNDED** second order scheme  
**INTEGRATION** for **U=0.3, V=0.25, T=0.2** by **BOUNDED** scheme  
**INTEGRATION** for **U, V, T**, all beta = **0.1** by **BOUNDED** scheme

## 5.58 LAGRANGIAN TRANSPORT

**PURPOSE** To select the Lagrangian Random-Walk algorithm for transport of species.

**MODE 1:** Specification of Maximum Number of Particles

**SYNTAX** **LAGR** {**PART**} {**N<sub>Part</sub>**} [**q<sub>Part</sub>**] [**MASS**[[**LOCA**[[**DENS**= $\rho$ ]]**RADI**= $\beta$ ]]] [**SURF**] [**OFF**]

**PART** The number of Lagrangian particles to be tracked for transport of a species.

**N<sub>Part</sub>** The maximum number of Lagrangian particles that can be present in the computational domain at any given time. During computations, new particles can be released, the particles can move out of the computational domain, and they can be eliminated and consolidated based on specified criteria. There is no default value.

**q<sub>Part</sub>** The mass of the Lagrangian particle at time of release in to the computational domain. The default value is 0.001.

**MASS** The species concentration at a grid cell is computed by assuming that the mass of a particle is distributed in the immediate neighborhood of the particle according to the inverse of square distance of the particle from the element node. See Section C.2 in Appendix C for further details. This is the default option.

**LOCA** The species concentration is computed by assuming that the mass of a particle is totally contained in the element of its location. See Section C.2 in Appendix C for further details.

**DENS** The species concentration is computed by assuming that a particle has a volume that is computed from its density. The mass of the particle is then distributed in the immediate neighborhood of the particle according to the intersection of the particle volume with the cell volume. See Section C.2 in Appendix C for further details. The default value is 2.

$\rho$  The mass density of the particles if **DENS** modifier is present. The default value is 1.

**RADI** The species concentration is computed by assuming that a particle has a radius of influence. The mass of the particle is then distributed in the immediate neighborhood according to the ratio of the distance of the particle to its radius of influence. The radius of influence is set equal to  $\beta$  times the characteristic half length of the element in which the particle is located. See Section C.2 in Appendix C for further details.

$\beta$  A factor that multiplies the cell characteristic half length to compute the radius of influence of the particle. The default value is 1.

**SURF** By default the particles are assumed to be distributed through the volume of the fluid and the concentration of the particles for any grid element is the mass in the element divided by its volume. For TIDAL® Software Tool, if depth-averaged assumption is used, and this modifier is present, then it is assumed that the particles are present above the water surface and the quantity computed is the mass divided by the surface area for the element. If instead of the mass of the particles (**LAGRANGIAN RELEASE** mode of command), the volume released is specified, then the computed quantity is the equivalent thickness of the layer of particles. This mode may be used to compute the spread of an oil spill. This then is the stochastic method to compute oil spill as opposed to the deterministic approach of the **SPILL** command.

**OFF** By default the concentration of species is computed at every time step. If this modifier is present, then the concentration is computed only in response to an **OUTPUT** or **SAVE** command specified with the symbol for the species. The computing time spent in concentration calculations can be saved if the concentration is not required except for purposes of output. This modifier must not be used if the species concentration is used as an active variable for other purpose.

**REMARKS**

A **LAGRANGIAN PARTICLE** command must be present to initiate the solution of the transport equation by the stochastic **RADM** (**RA**ndom **W**alk **D**ispersion **M**ethod) algorithm. Details of the algorithm and its applications and verification are given in a number of publications by Runchal(1980), Goodin et al. (1980), Goodin et al. (1982), and others. A brief description of RADM is given in Appendix C.

For each computational element (or cell) the concentration of the species is equal to the sum of the mass of particles in the cell divided by the volume of the cell. However in the macro scale simulations, each particle represents a large collection of “molecules” that are distributed around the mean computed position of the particle. Different methods for computing the concentration are described in [Section C.2 in Appendix C](#).

**EXAMPLES**

---

**LAGRANGIAN** transport with **10000 PARTICLES**.

**LAGRANGIAN** transport with **10000 PARTICLES** and mass of each as **0.002** kg with RADius factor as **10**.

**LAGRANGIAN** transport with **10000 PARTICLES** and mass of each as **0.002** kg concentration **OFF**.

**MODE 2:** Specification of Particle Release Rate

**SYNTAX** **LAGR** {**RELE**} [**Φ**] {**func**[**ξ**]} {**N**<sub>1</sub>, **N**<sub>2</sub>, ..., **N**<sub>n</sub>|**fname**} [**NUMB**=**N**<sub>part</sub>] [**INIT**|**INST**] [**subrgn**]

**RELE** A source for release of particles into the computational domain is specified.

**Φ** **One and only one** symbol for the dependent variable for which the Lagrangian transport is computed. Valid symbols are those for the species equations listed in Table 2.7.1. **There is no default value; a symbol must be specified.**

**func** One of the modifiers listed in Table 4.2.1 that denotes the functional form for computing the mass rate of release of particles. The rate of release has units of [M t<sup>-1</sup>] such as [kg/s].

**ξ** One of the independent variables listed in Table 4.2.2. **If no variable is specified, then the independent variable is assumed to be time.**

**N<sub>1</sub>,...,N<sub>n</sub>** The numerical constants and coefficients for the selected function. See Section 4.4 for more details. **There are no default values for this input.**

**fname** The name of the file containing the numerical values **N**<sub>2</sub> through **N**<sub>n</sub>. **This option is available only for selected functions. See Section 3.3 for additional information.**

**NUMB** **By default the mass of each particle is assumed to be that specified as **q**<sub>part</sub> with mode 1 of this command.** If this modifier is present, then it is assumed that the total number of particles for the release rate are given as **N**<sub>part</sub> and the mass of each is equal to the total release mass divided by the number of particles. If the mass release rate is function of time, then the actual mass of each released particle will vary with time.

**N<sub>part</sub>** The number of particles to be released at each time step, if the modifier **NUMB** is present.

**INIT|INST** **By default the release is assumed to be continuous and the specified mass is released at every time step.** If either of these modifiers is present, then the source is instantaneous and only once.

**subrgn** The subregion for which the input is specified. See Section 3.4. **If no subregion is specified, the entire computational domain is selected.**

**REMARKS**

A number of **LAGRANGIAN RELEASE** commands may be given for a simulation to accommodate the diversity of sources. **However currently only one variable may be selected for Lagrangian transport.**

**EXAMPLES**

**LAGRANGIAN** transport with **20000 RANDOM** numbers.

**MODE 3:** Detailed Output for Lagrangian Particles

**SYNTAX** LAGR {DETA} [TIME] [V<sub>frq</sub>] [NOW|ONLY|OFF]

**DETA** If this modifier is present, then a table of output is produced that includes the complete detail of the current location, mass and status of each particle. As the number of particles increases, this output can become very large especially if the frequency of output is not judiciously chosen.

**TIME** By default, V<sub>frq</sub> is the frequency of output in terms of number of steps. If this modifier is present, then V<sub>frq</sub> is interpreted to be time interval between successive outputs.

**V<sub>frq</sub>** The frequency (step or time interval) at which the output is written to the output device. See Section 3.7 for further details. The specified value is ignored if it is zero or negative. The default value is set so that output is obtained only at the end of simulations.

**NOW** The output is produced immediately after the command is encountered.

**ONLY** If this modifier is specified then the output is produced only once in response to a **NOW** modifier or at the frequency determined by **V<sub>frq</sub>**.

**OFF** Output for any previous command(s) for the same **subrgn** and type is subsequently suppressed.

#### REMARKS

---

This command only results in the computation of the species concentrations for the specified subdomain. A separate **OUTPUT** command must be given to obtain the output of the concentration field.

#### EXAMPLES

---

**LAGRANGIAN DETAIL** at end of simulations.

**LAGRANGIAN DETAIL** at every **TIME=100** units.

**MODE 4:**      **Specification of Size of The Random Number Distribution**

**SYNTAX**      **LAGR {RAND} {N<sub>random</sub>}**

**RAND**              The number of random numbers are being specified.

**N<sub>random</sub>**            The number of random numbers that comprise a uniform random number distribution with a mean value of zero and a range of unity that is used to simulate the diffusive and dispersive transport of a species. **The default value is 10,007.**

#### **REMARKS**

---

These random numbers are sampled during diffusive transport of each Lagrangian particle. The sampling occurs sequentially during transport of these particles. For most applications, 10000 random numbers provide a sufficiently large sample. However the diffusive resolution improves as the number is increased.

#### **EXAMPLES**

---

**LAGRANGIAN** transport with **20000 RANDOM** numbers.

**5.59 LAMINAR**

**PURPOSE** To select laminar flow option. This command is effective only for the **ANSWER®** Software Tool.

**SYNTAX** **LAMI**

**REMARKS**

---

By default the **ANSWER®** Software Tool selects the laminar or turbulent mode of computations automatically. If any of the turbulence related quantities ( $k$ ,  $\epsilon$ , or  $l$ ) are specified then the turbulence mode is initiated, otherwise flow is assumed to be laminar. This command may be employed to force laminar flow computations. This command is not available with **PORFLOW®** and **TIDAL®** Software Tools.

**EXAMPLES**

---

**LAMI**nar flow for this problem.

**LAMI**nar flow! **By default**

## 5.60 LAND

**PURPOSE** To define a land or solid object within the flow domain. This command is effective only for the **ANSWER®** and **TIDAL®** Software Tools.

**REMARKS** \_\_\_\_\_

This command is identical to the **BLOCK** command. Please see the **BLOCK** command for details.

## 5.61 LIMIT

**PURPOSE** To specify the limiting values for field variables

**SYNTAX** **LIMI** { $\Phi$ } { $\Phi_{Min}$  ,  $\Phi_{Max}$  | **OFF**} [**MATR**]

$\Phi$  A symbol that denotes the variable for which the limits are being set.

**OFF** All previously set limits for the selected variable are deactivated.

$\Phi_{Min}$  The minimum permissible value for the variable. Any computed value that is less than  $\Phi_{Min}$  is set to  $\Phi_{Min}$ .

$\Phi_{Max}$  The maximum permissible value for the variable. Any computed value that is greater than  $\Phi_{Max}$  is set to  $\Phi_{Max}$ .

**MATR** By default when the limit for a variable is specified, the solution computed by the matrix solver is bounded and truncated by the specified bounds. The matrix itself is not modified in any manner to prevent the variable from exceeding the specified limits. If this modifier is present, then the forcing function of the matrix is modified to ensure that the matrix is unlikely to generate values below the specified minimum value. This option can only be used if transient mode of the **SOLVE** command is invoked.

**REMARKS**

The default values for all field variables are set to  $-10^{30}$  as the minimum and  $10^{30}$  as the maximum. However, these values may be internally revised for some of the variables (such as mass concentration) based on the nature of the problem and that of the variable. This command may be used to override the internally set values, if so desired.

**EXAMPLES**

**LIMI**t for T minimum = 300

**LIMI**t for T minimum = 0 maximum = 100

**LIMI**t for T **OFF**! Disable any previously specified or internally set limits

## 5.62 LOCATE

**PURPOSE** To locate and identify a subregion in the domain of computation for later reference by other input commands.

**MODE 1:** Subregion Specification by Grid Indices for a Structured Grid

**SYNTAX** **LOCA** [**WIND**] [**ID=idsub**] [**FIEL | BOUN**] [**FLUI | SOLI**] [**INTE**] [**NOT**] [**DIFF**] [**EXCL**] {**N<sub>1</sub>, ..., N<sub>n</sub>**} [**ISK,JSK,KSK**] [**fname**]

**WIND** One or more “windows” (with a maximum of 8000) are specified by their grid (I, J, K) indices. Each window is defined by 2 sets of grid indices. The 1<sup>st</sup> set defines the “lower-left” and the 2<sup>nd</sup> set the “upper-right” corner of the window. The subsequent windows are defined in an identical manner. There is no restriction on the coordinate values. But each window must be specified by 4 numerical values for 2D and 6 for 3D geometry. This modifier is the default option for the **LOCATE** command.

**idsub** A unique identifier for the subregion consisting of up to 32 characters. If there are more than 32 characters, then the subsequent characters are ignored. The 1<sup>st</sup> character must be an alphabetic (A-Z) character. No distinction is made between the lower and upper case characters. The allowable set of characters includes the alphabetic (A-Z), the numeric (0-9), the underscore (\_), the minus (-), the plus (+) and the period (.) characters. The identifier is terminated by the 1<sup>st</sup> separator character. A list of separator characters is given in Chapter 1. This identifier may be subsequently used in an input command to provide selective input for that subregion. If the identifier is omitted, then the modifier **LOCATE** or **SELECT** on the command can make a generic reference to the specified subregion until the next **LOCATE** or **SELECT** command. See Section 3.4 for additional details.

**FIEL** If the **FIELD** modifier is present, then only the interior field nodes (or elements) of the subdomain are included in the subregion. Any exterior boundary nodes are excluded.

**BOUN** If the **BOUNDARY** modifier is present, then only the exterior domain boundary nodes are selected. Any elements interior to the computational domain are excluded.

**FLUI** If this modifier is present then only the fluid elements are retained; any elements that are solid or blocked (**BLOCK** command) are excluded from the selected set of elements.

**SOLI** If this modifier is present then only the solid or blocked (**BLOCK** command) elements are included in the selected set of elements.

**INTE** If multiple windows are defined, then by default the union (elements that belong to any) is selected. If this modifier is present, then the intersection (elements that belong to more than one window) are selected.

**NOT** This modifier is effective only if the **INTERSECTION** modifier is also specified. It selects the union of multiple windows minus their intersection (elements that belong to only one and, not more than one, window)

**DIFF** If multiple windows are defined, then this modifier selects the difference of the 2<sup>nd</sup> through the n<sup>th</sup> window from the 1<sup>st</sup> window. Only those elements from the 1<sup>st</sup> window are retained that are not present in any other window. Any elements that are present in the 2<sup>nd</sup> through the n<sup>th</sup> window but are not in the 1<sup>st</sup> window are ignored. If only two windows are specified, then this is equivalent to the presence of the **INTE** and the **NOT** modifiers.

**EXCL** If the modifier is present, then the elements otherwise selected from the windows are excluded and the complimentary set in the computational domain is selected. For example, if the union is selected, and this modifier is present, then all the elements that belong in the union are excluded and rest of the computational domain is selected.

**N<sub>1</sub>, ..., N<sub>n</sub>** The grid index coordinates (I, J, K) for the “lower-left” and “upper-right” corners of the window (s). The number of values specified must be in sets of 4 for 2D and 6 for 3D for each window. If the values specified lie outside the range specified on the **GRID** command then the input values are clipped to lie in the valid range. See Section 4.4 for additional information. There is no default value; a valid set of values must be specified.

**ISK,JSK, KSK** If only 1 window is specified, then a skip frequency for the window indices in the I, J and K

(x, y, z) direction may be specified. In this case, only those nodes occurring at this interval, starting with  $N_1$  specification are considered to be part of the subregion. For example, a specification of  $I_{sk}=3$  will result in the nodes located at  $I=N_1$ ,  $I=N_1+3$ ,  $I=N_1+6$ , etc. to be included in the selected subregion. The y and z direction node intervals have a similar effect. The default value for all is 1.

**fname** If a valid file name is present (see Section 3.3) then an ordered list of element numbers selected by the command is written to the file. For structured grids a list of corresponding (I,J,K) grid indices is also additionally written to the file. The file has a header that consists of a **LOCATE** command with the **ID=idsub** and a comment field that has the number of elements in the subdomain. This file can be used for input (See Mode 10 of the **LOCATE** command).

## EXAMPLES

---

**LOCAt**e subregion from (6,10) to (31,10)

**LOCAt**e subregion (6,10) to (31,10) with ID=DMN1

**LOCAt**e region ID=DMN2 as: (6,10,1) to (31,10,15) with interval (2,3,1)

**LOCAt**e region (6,10) to (31,10) with interval (2,3)

**LOCAt**e subregion (1,1) to (22,22) only FIELd nodes as ID=INNR

**LOCAt**e subregion (1,1,3) to (22,22,7) only FIELd nodes as ID=INNR output on file 'INNR.IJK'

**MODE 2:** Subregion Specification by Grid Coordinates of Rectangular Windows

**SYNTAX** **LOCA** {**COORD**} [**ID=idsub**] [**FIEL** | **BOUN**] [**FLUI** | **SOLI**] [**INTE**] [**NOT**] [**DIFF**] [**EXCL**] {**N<sub>1</sub>, ..., N<sub>n</sub>**} [**fname**]

**COORD** One or more “windows” (with a maximum of 8000) are specified by their grid (x, y, z) coordinates. Each window is defined by 2 sets of coordinates. The 1<sup>st</sup> set defines the “lower-left” and the 2<sup>nd</sup> set the “upper-right” corner of the window. The subsequent windows are defined in an identical manner. There is no restriction on the coordinate values. But each window must be specified by 4 numerical values for 2D and 6 for 3D geometry.

The elements are selected based on the coordinate value of the element node. All elements with the node located inside or on the boundary of the window are selected by this command. An element is considered either in or out. No account is taken of the fact that the selected elements may be partially in or out. The elements selected are further affected by the presence or absence of **FIELD** and **BOUNDARY** modifiers.

**idsub** See Mode 1 Specification.

**FIEL** See Mode 1 Specification.

**BOUN** See Mode 1 Specification.

**FLUI** See Mode 1 Specification.

**SOLI** See Mode 1 Specification.

**INTE** See Mode 1 Specification..

**NOT** See Mode 1 Specification.

**DIFF** See Mode 1 Specification.

**EXCL** See Mode 1 Specification.

**N<sub>1</sub>, ..., N<sub>n</sub>** The grid coordinates (x,y,z) for the “lower-left” and “upper-right” corners of the window (s). The number of values specified must be in sets of 4 for 2D and 6 for 3D for each window. See Section 4.4 for additional information. **There is no default value; a valid set of values must be specified.**

**fname** See Mode 1 Specification.

**REMARKS**

If a **SCALE** command is specified immediately preceding this command, then the input coordinate values are scaled as described in the **SCALE** command.

**EXAMPLES**

**LOCA**te region with **COORD**inates (0., -20.) to (11571.5, 80.)

**LOCA**te subregion **ID** = DMN2 for grid **COORD**inates (0., 0.) to (100, 1500)

**LOCA**te subregion **ID** = DMN2 for **COORD**inates (0., 0.) to (100, 1500) **EXCL**ude selection

**LOCA**te **ID** = DMN2 for grid **COORD**inates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2)

**LOCA**te grid **COORD**inates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2) and **EXCL**ude selected

**LOCA**te **COORD**inates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2) select **INTER**section & **EXCL**ude

**LOCA**te **COORD**inates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2) select **NOT INTER**section & **EXCL**ude

**LOCA**te **COORD**inates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2) select output on “DMN2.LOC”

**LOCA**te **COORD**inates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2) select only **BOUND**ary

**MODE 3: Subregion Specification by a Polygonal Window****SYNTAX** **LOCA** {**POLY**} [**ID=idsub**] [**EXCL**] [**FIEL|BOUN**] [**FLUI|SOLI**] {**N<sub>1</sub>, ..., N<sub>n</sub>**} [**fname**]**POLY** A polygonal region is specified by the coordinates of its vertices. **The polygon must be convex; concave polygons may lead to inaccurate selection.**

For 2D geometry, the pairs of (x, y) coordinates for vertices of the polygon must be specified in a counterclockwise sense.

For 3D geometry, current formulation only allows for the specification of 8-vertex hexahedral windows. First the base of the hex must be defined by 4 sets of (x, y, z) coordinates of its vertices in a counterclockwise sense. The last 4 sets of (x, y, z) coordinates define the top surface of the hex in a counterclockwise sense.

The elements are selected based on the coordinate value of the element node. All elements with the node located inside or on the boundary of the polygon are selected by this command. An element is considered either in or out. No account is taken of the fact that the selected elements may be partially in or out. The elements selected are further affected by the presence or absence of **FIELD** and **BOUNDARY** modifiers.**idsub** See Mode 1 Specification.**EXCL** If the modifier is present, **then the specified elements are excluded** and the complimentary set in the computational domain is selected.**FIEL** See Mode 1 Specification.**BOUN** If the **BOUNDARY** modifier is present, then only the external boundary nodes are selected. Any elements interior to the computational domain are excluded.**FLUI** See Mode 1 Specification.**SOLI** See Mode 1 Specification.**N<sub>1</sub>, ..., N<sub>n</sub>** The grid coordinates (x, y) or (x, y, z) for the vertices of the polygon. The number of values must equal  $N_{23D} * N_{vertex}$  where  $N_{23D}$  is 2 for 2D and 3 for 3D geometry and  $* N_{vertex}$  is the number of vertices. **There is no default value; a valid set of values must be specified.****fname** See Mode 1 Specification.**REMARKS**If a **SCALE** command is specified immediately preceding this command, then the input coordinate values are scaled as described in the **SCALE** command.**EXAMPLES****LOCA**te POLYgonal region (x, y) coordinates are: (0, 0) (1,0) (1,1) (0,1) **!2D rectangle****LOCA**te POLYgon (x, y) coordinates are: (0, 0) (1,-1) (2,0) (1,1) **!Diamond shaped region****LOCA**te POLYgon (x, y): (0, 0) (0.5,-0.866), (1.5,-0.866) (2,0) (1.5,0.866) (0.5,0.866) **!Hexagon****LOCA**te POLYgon (x, y, z): Base: (0, 0,0) (1,-1,0) (2,0,0) (1,1,0)Top: (0, 0,1) (1,-1,1) (2,0,1) (1,1,1) **!Hexahedral with diamond base****LOCA**te region ID=DIAMOND POLYgon (0, 0) (1,-1) (2,0) (1,1) output on "FILE.LOC"**LOCA**te ID=NOT\_DIAMOND EXCLUDE POLYgon (0, 0) (1,-1) (2,0) (1,1) output on "FILE.LOC"

**MODE 4: Specification of a Circular or Cylindrical Subregion**

**SYNTAX** **LOCA** {**CYLI**|**CIRC**} [**ID=**idsub] [**FIEL**|**BOUN**] [**FLUI**|**SOLI**] {**d<sub>cyl</sub>**} [**N<sub>1</sub>,...** **N<sub>n</sub>**] [**IJK**|**ELEM**] [**N<sub>n+1</sub>,...N<sub>m</sub>**], [**N<sub>m+1</sub>**] [**fname**]

**CYLI | CIRC** A cylindrical (circular in 2D) subregion is specified. Only the internal elements can be specified. Associated boundary nodes are automatically included unless the **FIELD** modifier is present.

**idsub** See Mode 1 specification.

**FIEL** See Mode 1 Specification.

**BOUN** If the **BOUNDARY** modifier is present, then only the external boundary nodes are selected. Any elements interior to the computational domain are excluded.

**FLUI** See Mode 1 Specification.

**SOLI** See Mode 1 Specification.

**d<sub>cyl</sub>** The diameter of the cylinder of the identified subregion.

**N<sub>1</sub>,...N<sub>n</sub>** In the absence of the **IJK** or **ELEM** modifier, these specify the (x,y) or the (x,y,z) coordinates of the center of the cylinder. In the presence of **IJK** or **ELEM** modifier these are interpreted as given below.

**IJK** The numerical input [**N<sub>1</sub>, . N<sub>n</sub>**] specifies the grid indices (I,J) or (I,J,K) of the element. The center of the circle or cylinder is assumed to coincide with the center (node) of the element. Two values must be specified for 2D and three for 3D geometry

**ELEM** The numerical input [**N<sub>1</sub>**] specifies the element which is at the center of the cylinder. Only 1 value must be specified.

**N<sub>n+1</sub>,...N<sub>m</sub>** The components of a vector (2 values in 2D and 3 in 3D) normal to the plane in which the center of the cylinder is located. **An elliptic region can be selected by appropriate choice of the normal vector.** The values may be in arbitrary units and are internally converted to unit normals. **At least one value must be specified.**

**N<sub>m+1</sub>** The half-length of the cylinder or the tolerance in the direction normal to the plane of the circle. The logic employed computes the normal and tangential distance of the existing element centers (nodes) from the center of circle in reference to the specified plane. If the tangential distance is  $\leq$  the radius of the cylinder, then the element is included **provided its normal distance from the plane is less than the tolerance.** **By default the tolerance is set to a large number (1.E+30) so that all elements in the projected plane of the circle are captured.** This input may be used to specify a different tolerance to capture only elements within a certain distance from the plane. **If specified, the intervening values for all components of the unit normal (even if zero) must be specified**

**fname** See Mode 1 Specification.

**REMARKS**

If a **SCALE** command is specified immediately preceding this command, then **d<sub>cyl</sub>** and **N<sub>1</sub>,...N<sub>n</sub>** (in the absence of **IJK** or **ELEM**) are scaled as described in the **SCALE** command.

**EXAMPLES**

**LOCATE CIRCLE** dia=0.4, center coordinates (2., 0.5) I2D  
**LOCATE CIRCLE** dia=0.4, center coordinates (2., 0.5) external BOUNdary only I2D  
**LOCATE CIRCLE** dia=0.4, center coordinates (2., 0.5, 0.5)  
**LOCATE CIRCLE** dia=0.4, center coordinates (2., 0.5, 0.5) normal (1.,1.0.) !45 degree in xy  
**LOCATE CYLINDER** dia=0.4, center coordinates (2., 0.5, 0.5) normal (1.,1.0.) half length=0.5  
**LOCATE CYLINDER** dia=0.4, center (2., 0.5, 0.5) normal (1.,1.0.) half length=0.5 file 'Cylinder.loc'  
**LOCATE CYLINDER** dia=0.4, center IJK at (11,5,7) normal (1.,1.0.) half length=0.5  
**LOCATE CYLINDER** dia=0.4, center ELEMENT at (147) normal (1.,1.0) half length=0.5

**MODE 5:** Specification of a Spherical Subregion**SYNTAX** **LOCA** {**SPHE**} [**ID=idsub**] [**FIEL**] [**FLUI|SOLI**] {**R<sub>sphere</sub>**} [**N<sub>1</sub>, . N<sub>n</sub>**] [**fname**]**SPHE** A spherical (circular in 2D) subregion is specified. Only the internal elements can be specified. Associated boundary nodes are automatically included unless the **FIELD** modifier is present.**idsub** See Mode 1 specification.**FIEL** See Mode 1 Specification.**FLUI** See Mode 1 Specification.**SOLI** See Mode 1 Specification.**R<sub>sphere</sub>** The radius of the sphere for the identified subregion.**N<sub>1</sub>, . N<sub>n</sub>** The (x,y) or the (x,y,z) coordinates of the center of the sphere.**fname** See Mode 1 Specification.**REMARKS**

If a **SCALE** command is specified immediately preceding this command, then **R<sub>sphere</sub>** and **N<sub>1</sub>,...N<sub>n</sub>** are scaled as described in the **SCALE** command.

**EXAMPLES**

**LOCA**te SPHEre radius=0.4, center coordinates (2., 0.5) !2D  
**LOCA**te SPHEre radius=0.4, center coordinates (2., 0.5, 0.5 )  
**LOCA**te SPHEre radius =0.4, center coordinates (2., 0.5, 0.5 ) **FIEL** node only  
**LOCA**te SPHEre radius =0.4, center coordinates (2., 0.5, 0.5 ) output to file 'SPHERE.LOC'

**MODE 6: Specification of an Annular Subregion**

**SYNTAX** **LOCA** {**ANNU**} [**ID=idsub**] [**FIEL**] [**FLUI|SOLI**] {**D<sub>o</sub>**, **D<sub>i</sub>**} [**N<sub>1</sub>**, . **N<sub>n</sub>**] [**IJK|ELEM**] [**N<sub>n+1</sub>**,..**N<sub>m</sub>**],  
**[N<sub>m+1</sub>]** [**fname**]

**ANNU** An annular subregion is specified. Only the internal elements can be specified. Associated boundary nodes are automatically included unless the **FIELD** modifier is present.

**idsub** See Mode 1 specification.

**FIEL** See Mode 1 Specification.

**FLUI** See Mode 1 Specification.

**SOLI** See Mode 1 Specification.

**D<sub>o</sub>** The outer diameter of the annulus (circular or cylindrical) of the identified subregion.

**D<sub>i</sub>** The inner diameter of the annulus (circular or cylindrical) of the identified subregion.

**N<sub>1</sub>,. N<sub>n</sub>** In the absence of the **IJK** or **ELEM** modifier, these specify the (x,y) or the (x,y,z) coordinates of the center of the circle or the cylinder. In the presence of **IJK** or **ELEM** modifier these are interpreted as given below.

**IJK** The numerical input [**N<sub>1</sub>**, ... **N<sub>n</sub>**] specifies the grid indices (I,J) or (I,J,K) of the element. The center of the circle or cylinder is assumed to coincide with the center (node) of the element. 2 values must be specified for 2D and 3 for 3D geometry

**ELEM** The numerical input **N<sub>1</sub>** specifies the element which is at the center of the cylinder. Only 1 value must be specified.

**N<sub>n+1</sub>, ... N<sub>m</sub>** See Mode 4 Specification.

**N<sub>m+1</sub>** See Mode 4 Specification.

**fname** See Mode 1 Specification.

**REMARKS**

If a **SCALE** command is specified immediately preceding this command, then **D<sub>o</sub>**, **D<sub>i</sub>** and **N<sub>1</sub>,...N<sub>n</sub>** (in the absence of **IJK** or **ELEM** ) are scaled as described in the **SCALE** command.

**EXAMPLES**

**LOCAtE** ANNULus outer dia=0.4, inner dia = 0.2 center coordinates (2., 0.5) !2D

**LOCAtE** ANNULar Cylinder dia=0.4, d\_inner=0.2 center (2., 0.5, 0.5 ) normal (1.,1.0.) half length=0.5

All Examples from Mode 4 apply except that the inner diameter must follow as the second numerical value. All subsequent input is in the same order.

**MODE 7:**      **Subregion Specification by Material Type****SYNTAX**      **LOCA** {**MATE|ZONE**} [**ID=idsub**] {**N<sub>mat</sub>**} [**FIEL**] [**FLUI|SOLI**] [**fname**]**MATE**            The subregion is identified by a Material type of Zone number.**ZONE**            Same as MATE modifier.**idsub**            See Mode 1 specification.**N<sub>mat</sub>**            The material type or zone number that previously appeared in a **MATERIAL** (or an equivalent) command. The material type may denote a non-contiguous and non-rectangular subregion.**FIEL**            See Mode 1 Specification.**FLUI**            See Mode 1 Specification.**SOLI**            See Mode 1 Specification.**fname**           See Mode 1 Specification.**REMARKS** \_\_\_\_\_

This mode of the **LOCATE** command is currently implemented in a limited manner; please consult **ACRi** before its use.

**EXAMPLES** \_\_\_\_\_**LOCAtE** **MATER**ial type 3 as the active subregion**LOCAtE** **ZONE** number 5 as subregion with ID=TYP5**LOCAtE** **ZONE** number 5 as subregion with ID=TYP5 **FIELd** nodes only

**MODE 8: Subregion Specification by a Random List of Coordinates****SYNTAX** **LOCA** {**COORD**} {**LIST**} [**ID=idsub**] [**FIEL**] [**FLUI** | **SOLI**] [**fname**] {**N<sub>1</sub>** | **N<sub>1</sub>, ..., N<sub>n</sub>**}

**COORD** Explicit list of coordinates that specify a number of points are specified. For each point, the nearest element (by location of its node) is selected and included in the subdomain. Only the internal elements are selected. If a point is located at or outside the domain boundary, then the nearest element inside the domain is selected. Associated boundary nodes are automatically included unless the **FIELD** modifier is present.

**LIST** This modifier denotes that the **COORD** modifier applies to a random list. **It must be present to invoke this mode of the command.**

**idsub** See Mode 1 specification.

**FIEL** See Mode 1 Specification.

**FLUI** See Mode 1 Specification.

**SOLI** See Mode 1 Specification.

**fname** If a file name is present then the file may be used for input or output. The **input mode** is activated if a file name is specified with **only one** numerical value (**N<sub>1</sub>**). The file may contain a header field with up to 100 lines (records) of information. **The header field, if present, must end with the record "END HEADER" and the numerical values must immediately follow this record. In the output mode,** a file is generated that lists the element and surface number pairs and, for structured grids, a list of (I,J,K) grid indices. In the file input or output mode, a valid name (see Section 3.3) must be specified.

**N<sub>1</sub>** If the file input mode with a valid **fname** is activated, then **N<sub>1</sub>** denotes the number of elements to be read from the file. If the file input mode is not activated, then **N<sub>1</sub>, ..., N<sub>n</sub>** are described below.

**N<sub>1</sub>,...,N<sub>n</sub>** The (x, y) or (x, y, z) coordinates for the points that comprise the list. For each point, 2 values must be specified for the 2D and 3 for 3D geometry. **There is no default value.**

**REMARKS**

If a **SCALE** command is specified immediately preceding this command, then the input coordinate values are scaled as described in the **SCALE** command.

**EXAMPLES**

**LOCA**te COORdinate LIST: (0.,0.), (2,2), (1.53,1.37), (23.1,27.2) ! List for 4 points in  
2D

**LOCA**te COORdinate LIST: (0.,0,0), (2,2,2), (1.53,1.37,1), (23.1,27.2,2) ! List for 4 points in  
3D

**LOCA**te COORdinate LIST of 500 elements as ID=BIG from file 'XYZ.LST' ! Read from file

**LOCA**te COOR LIST 500 as ID=BIG from file 'XYZ.LST' only FIELD ! Read file; only field

**LOCA**te COOR LIST: (0.,0.), (2,2), (1.53,1.37), (23.1,27.2) output to 'FILE.rgn' ! List with output file

**MODE 9: Subregion Specification by Grid Indices for Structured Grid****SYNTAX** **LOCA** {**IJK**|**IJ**} [**ID=**idsub] [**FIEL** | **BOUN**] [**FLUI**|**SOLI**] [**fname**] {**N<sub>1</sub>**|**N<sub>1</sub>, ..., N<sub>n</sub>**}**IJK** An explicit list of grid indices of elements (I,J) for 2D or (I,J,K) for 3D geometry is specified. This option can be used only for structured grids. Both the interior and the boundary nodes of the computational domain can be specified.**IJ** Same as **IJK**.**idsub** See Mode 1 specification.**FIEL** See Mode 1 Specification.**BOUN** See Mode 1 Specification.**FLUI** See Mode 1 Specification.**SOLI** See Mode 1 Specification.**fname** If a file name is present then the file may be used for input or output. The **input mode** is activated if a file name is specified with **only one** numerical value (**N<sub>1</sub>**). The file may contain a header field with up to 100 lines (records) of information. **The header field, if present, must end with the record "END HEADER" and the numerical values must immediately follow this record. In the output mode, a file is generated that lists the element and surface number pairs and, for structured grids, a list of (I,J,K) grid indices. In the file input or output mode, a valid name (see Section 3.3) must be specified.****N<sub>1</sub>** If the file input mode with a valid **fname** is activated, then **N<sub>1</sub>** denotes the number of elements to be read from the file. If the file input mode is not activated, then **N<sub>1</sub>, ..., N<sub>n</sub>** are described below.**N<sub>1</sub>,...,N<sub>n</sub>** These specify the grid indices for the list of elements. For each element, 2 values must be specified for the 2D and 3 for 3D geometry.**EXAMPLES****LOCAtE** ID=LISTIJK by IJK indices: (2,5), (3,3), (5,7), (9,2), (3,4) for a two-dimensional structured grid**LOCAtE** ID=LIST by IJ indices: (2,5), (3,3), (5,7), (9,2), (3,4) only FIELD elements**LOCAtE** IJK indices: (2,5,2), (3,3,5), (5,7,2), (9,2,3), (3,4,5) for three-dimensional structured grid**LOCAtE** IJK indices: (1,1), (1,22), (22,1), (22,22), EXTERior corner points of the grid**LOCAtE** IJK indices: 50 values from the file 'GETLIST.IJK'

**MODE 10: Subregion Specification by a List of Elements****SYNTAX** **LOCA** {**LIST**} [**ID=idsub**] [**FIEL** | **BOUN**] [**FLUI**|**SOLI**] [**fname**] {**N<sub>1</sub>**|**N<sub>1</sub>, ..., N<sub>n</sub>**}

**LIST** Explicit list of element numbers is specified. Only the internal elements can be specified. Associated boundary nodes, if any, are automatically included unless the **FIELD** or **BOUNDARY** modifiers are present. For unstructured grids this is the default mode of the command; however the modifier must be specified for a structured grid.

**idsub** See Mode 1 specification.

**FIEL** See Mode 1 Specification.

**BOUN** See Mode 1 Specification.

**FLUI** See Mode 1 Specification.

**SOLI** See Mode 1 Specification.

**fname** If a file name is present then the file may be used for input or output. The **input mode** is activated if a file name is specified with **only one** numerical value (**N<sub>1</sub>**). The file may contain a header field with up to 100 lines (records) of information. The **header field, if present, must end with the record "END HEADER"** and the numerical values must immediately follow this record. In the **output mode**, a file is generated that lists the element and surface number pairs and, for structured grids, a list of (I,J,K) grid indices. In the file input or output mode, a valid name (see Section 3.3) must be specified.

**N<sub>1</sub>** If the file input mode with a valid **fname** is activated, then **N<sub>1</sub>** denotes the number of elements to be read from the file. If the file input mode is not activated, then **N<sub>1</sub>, ..., N<sub>n</sub>** are described below.

**N<sub>1</sub>,...,N<sub>n</sub>** The element numbers that define the subregion. .

**EXAMPLES**

**LOCAt**e element numbers LIST: 1, 2, 3, 7, 17, 29

! Structured Grid

**LOCAt**e element numbers: 1, 2, 3, 7, 17, 29

! Default for Unstructured Grid

**LOCAt**e LIST of 500 elements from file 'ZONE.BIG'

! Read from file

**LOCAt**e 5000 elements from file 'ZONE.BIG'

! Read from file for Unstructured Grid

**LOCAt**e LIST 79, 22, 33, 34, 89, 2, 7 with ID=LST1 and output on file "LST1.LOC" ! Write to file

**MODE 11:**      **Subregion Specification Defined by Sequential Elements**

**SYNTAX**      **LOCA {SEQU} [ID=idsub] [FIEL|BOUN] [FLUI|SOLI] {N<sub>1</sub>, N<sub>2</sub>} [N<sub>3</sub>]**

**SEQU**      A sequence of elements is specified. Only the internal elements can be specified. Associated boundary nodes are automatically included unless the **FIELD** modifier is present. For unstructured grids this is the default mode of the command; however the modifier must be specified for a structured grid.

**idsub**      See Mode 1 specification.

**FIEL**      See Mode 1 Specification.

**BOUN**      See Mode 1 Specification.

**FLUI**      See Mode 1 Specification.

**SOLI**      See Mode 1 Specification.

**N<sub>1</sub>, N<sub>2</sub>, N<sub>3</sub>**      These three numbers specify the starting, the ending and the interval index for the element numbers in the sense of a FORTRAN DO loop. If **N<sub>3</sub>** is not specified, it is assumed to be 1.

#### **EXAMPLES**

---

**LOCAt**e ID=ILISTSEQ of SEQUential elements from 1 through 51 interval=10

**LOCAt**e SEQUence from 31 to 51

! All elements from 31 to 51,inclusive

**MODE 12:** Subregion Specification Defined by Sequential Elements**SYNTAX** **LOCA** {**RANG**} {**Φ**} {**Φ<sub>Min</sub>**, **Φ<sub>Max</sub>**} [**ID=idsub**] [**FIEL|BOUN**] [**FLUI|SOLI**]**RANG** A sequence of elements that have a value of the variable that satisfies the given constraints is included in the specified subdomain identified with **ID=idsub**.**Φ** A symbol for the variable the value of which is used as the criterion for selecting the elements that fall within the specified range. The valid symbols are listed in Table 2.8.1-3.**Φ<sub>Min</sub>**, **Φ<sub>Max</sub>** All elements that satisfy the criterion  $\Phi_{Min} \leq \Phi \leq \Phi_{Max}$  are selected to be included in the subdomain.**idsub** See Mode 1 specification.**FIEL** See Mode 1 Specification.**BOUN** See Mode 1 Specification.**FLUI** See Mode 1 Specification.**SOLI** See Mode 1 Specification.**EXAMPLES****LOCA**te RANGE of P between 10 and 100 as ID=P\_RANGE**LOCA**te RANGE of P between 10 and 100 for FLUID elements as ID=P\_FLUID

**MODE 13:**      **Specification of a Given Location as a Station**

**SYNTAX**      **LOCA {STAT} {ID=idsub} {N<sub>1</sub>, ..., N<sub>n</sub>} [fname]**

**STAT**            A single location is identified as a station for subsequent input. The field values are then obtained by interpolation of values at the specified stations.

**idsub**            See Mode 1 specification. **In this mode, the identifier must be specified.**

**N<sub>1</sub>, ..., N<sub>n</sub>**      The grid coordinates (x,y,z) for the station. Two numerical values for 2D and three for 3D input modes must be provided. **There is no default value for this input; the required number of numerical values must be specified.**

**fname**            See Mode 1 Specification.

#### **EXAMPLES**

---

**LOCA**te STAT ID=LOC2 at (0., -20.)

**LOCA**te STATION ID=LOC2 at coordinates (x=50., y=0., z=0.20)

**LOCA**te STATION ID=LOC2 at coordinates (x=50., y=0., z=0.20) output on file 'Station.Loc'

**MODE 14:**      **Subregion of Matched Elements**

**SYNTAX**      **LOCA {CORR} {ID=idsub1} {ID=idsub2} [ID=idsub] [fname]**

**CORR**      A subregion of correlated elements is defined for special purposes. The subregion consists of two sets of elements which are paired with each other. The pairing takes place in a parallel sequential mode where the 1<sup>st</sup> element in the 1<sup>st</sup> subregion is paired with the 1<sup>st</sup> element of the 2<sup>nd</sup> subregion, and so on. This type of subregion can only be used with certain commands (e.g. **CORRELATION**) where the values at the elements in the 1<sup>st</sup> set are paired with, or dependent on, the values of the elements in the 2<sup>nd</sup> set.

**idsub1**      The name of the 1<sup>st</sup> subregion to be operated upon. **There is no default value; a valid name of a previously specified subdomain with the ID modifier must be specified.**

**idsub2**      The name of the 2<sup>nd</sup> subregion to be operated upon. **There is no default value; a valid name of a previously specified subdomain with the ID modifier must be specified.**

**idsub**      A unique identifier for the subregion. **Only the first 32 characters are meaningful; any subsequent characters are ignored.** This identifier may be subsequently used in an input command to provide selective input for that subregion. If the identifier is omitted, then the modifier **LOCATE** or **SELECT** on the command can make a generic reference to the specified subregion until the next **LOCATE** or **SELECT** command. See Section 3.4 for additional details.

**fname**      If a file name is present then a file is generated that lists the elements and, for structured grids, a list of (I,J,K) grid indices. A valid name (see Section 3.3) must be specified.

#### **EXAMPLES**

---

**LOCA**te CORRelated Elements from ID=REGION1 and ID=REGION2

**LOCA**te CORRelated Elements from ID=REGION1 and ID=REGION2 as ID=MATCHED on file='match.fil'

**MODE 15: Boundary Specification by Paired Element and Surface Numbers****SYNTAX** **LOCA** {PAIR} [ID=idsub] [fname] {N<sub>1</sub>|N<sub>1</sub>, N<sub>2</sub>, N<sub>3</sub>, ... , N<sub>n</sub>}**PAIR** Explicit paired list of element and surface numbers is specified**idsub** See Mode 1 specification.**fname** If a file name is present then the file may be used for input or output. The **input mode** is activated if a file name is specified with **only one** numerical value (**N<sub>1</sub>**). The file may contain a header field with up to 100 lines (records) of information. **The header field, if present, must end with the record “END HEADER” and the numerical values must immediately follow this record. In the output mode**, a file is generated that lists the element and surface number pairs and, for structured grids, a list of (I,J,K) grid indices. In the file input or output mode, a valid name (see Section 3.3) must be specified.**N<sub>1</sub>** If the file input mode is activated, then, **N<sub>1</sub>** denotes the number of pairs of element and surface numbers to be read from the file. Each pair must specify an element number and a surface number as explained in **N<sub>1</sub>**, **N<sub>2</sub>** below. If the file input mode is not **activated**, then **N<sub>1</sub>** is interpreted as given below.**N<sub>1</sub>, N<sub>2</sub>** **N<sub>1</sub>** is the element number and **N<sub>2</sub>** is the surface number for the 1<sup>st</sup> element and surface pair that comprise the subregion. The surfaces of the element are numbered from 1 to 4 in the 2D and 1 to 6 in the 3D mode. See Section 3.4 for further details of the manner in which these surface numbers are assigned.**N<sub>3,..,N<sub>n</sub></sub>** The element number and a surface number for the rest of the elements which comprise current subregion in the manner **N<sub>1</sub>** and **N<sub>2</sub>**.**EXAMPLES****LOCA**te (element, surface) PAIRs: (35,1), (53,3), (77,2), (13,4)**LOCA**te PAIR : (35,1), (53,3), (77,2), (13,4) as boundary ID=BNDRy**LOCA**te PAIR : 500 sets as ID=BNDRy from file 'BOUNDARY.BIG'**LOCA**te PAIR : (35,1), (53,3), (77,2), (13,4) as boundary ID=BNDRy**LOCA**te PAIR : (35,1), (53,3), (77,2), (13,4) as boundary ID=BNDRy**LOCA**te PAIR : (35,1), (53,3), (77,2), (13,4), (28,5), (33,3), (35,6)**LOCA**te PAIR (35,1), (53,3), (77,2), (13,4), (28,5), (33,3), (35,6) output on file "Boundary.LOC"

**MODE 16: Boundary Specification by List of Vertices****SYNTAX** `LOCA {VERT} [ ID=idsub ] [fname] [EXCL] [SEQU] [option] {N1|N1, ..., Nn}`**VERT** The boundary region defined by the specified sets of vertices is selected.**idsub** See Mode 1 specification.**fname** If a file name is present then the file may be used for input or output. The **input mode** is activated if a file name is specified with **only one** numerical value (**N<sub>1</sub>**). The file may contain a header field with up to 100 lines (records) of information. **The header field, if present, must end with the record "END HEADER" and the numerical values must immediately follow this record. In the output mode,** a file is generated that lists the element and surface number pairs and, for structured grids, a list of (I,J,K) grid indices. In the file input or output mode, a valid name (see Section 3.3) must be specified.**EXCL** If the modifier is present, then the boundary specified by the vertices is excluded and the complimentary set of the domain boundary is selected.**SEQU** The numerical input specifies a sequence of element numbers.**option**

option	INTERPRETATION
<b>ANY</b>	By default a boundary element (or node) is included in the selection only if all its vertices are included in the list of vertices. If <b>ANY</b> modifier is specified then a boundary element is included if any of its vertices is included in the list.
<b>ONE</b>	A boundary element is included if at least one of its vertices is included.
<b>TWO</b>	A boundary element is included if at least two of its vertices are included.
<b>THRE</b>	A boundary element is included if at least three of its vertices are included.
<b>FOUR</b>	A boundary element is included if at least four of its vertices are included

**N<sub>1</sub>** If the file input mode is activated, then, **N<sub>1</sub>** denotes the number of vertices to be read from the file. If the file input mode is not activated, then **N<sub>1</sub>, ..., N<sub>n</sub>** are described below.**N<sub>1</sub>,...,N<sub>n</sub>** In the absence of the **SEQUENCE** modifier, these numbers identify the vertices that define the boundary of interest. In the presence of the **SEQUENCE** modifier only **N<sub>1</sub>, N<sub>2</sub>** and **N<sub>3</sub>** are meaningful. In this case, these three numbers specify the starting, the ending and the interval index for the vertices in the sense of a **FORTRAN DO** loop.**EXAMPLES****LOCAte** VERTices 1,3,7,8,10,20,55,99,203,105,77**LOCAte** VERTices: in SEQUENCE from 23 through 231 in steps of 3 as boundary ID=BNDRy**LOCAte** VERTices: 500 values as ID=BNDRy from file 'BOUNDARY.BIG'**LOCAte** VERTices: in SEQUENCE 23 to 231 in steps of 3 ID=BNDRy output file: 'Boundary.LOC'

**MODE 17:** Boundary Specification by Exclusion of Previously Specified Boundary

**SYNTAX** **LOCA** {**BOUN**} {**UNSP** | **dir** | **ALL**} [**ID=idsub**]

**BOUN** A boundary subregion is selected from the total external boundary of the domain

**UNSP** Selects the previously unspecified boundary. The selected boundary is the total external boundary of the domain but excluding the surfaces defined by all the previous **LOCATE** commands which identify a boundary (**LOCATE PAIR**, **LOCATE VERTEX**, **LOCATE COORDINATE BOUNDARY** and **LOCATE POLYGON BOUNDARY** commands). **This modifier must be present to activate this mode of the command.**

**dir** One of more of the modifiers that identify the direction(s) of the external boundary as defined in Section 3.5. The appropriate boundary segment is then selected that corresponds to the combination of the modifiers.

**ALL** The total external boundary of the domain is selected. **This is the default option.**

**idsub** See Mode 1 specification.

#### EXAMPLES

---

**LOCATE BOUNDARY** that is previously **UNSPECIFIED**  
**LOCATE UNSPECIFIED BOUNDARY** as **ID=BND\_Unspecified**  
**LOCATE BOUNDARY X-** and **Z-** **ID=BND\_XZ**  
**LOCATE BOUNDARY ALL** as **ID=BND\_TOTAL**

**MODE 18:**     **Boundary Specification as a Complimentary Surface of a Previous Subregion**

**SYNTAX**     **LOCA {COMP} {ID=idsub1} {dir1} [ID=idsub] [fname]**

**COMP**         A new sub region is created that represents the complimentary side of the surface of a previous sub region. Each surface that connects two elements can be viewed from the side of either element. This command can be used to select the complimentary side. **It cannot be used for the exterior surface of the domain or on regions that are defined by MATCHED PAIR or CORRELATED type of element sub domains or on previous LOCATE commands defined with a FIELD modifier**

**Idsub1**        The identification or name of the previous subregion to be operated upon. **There is no default value; a valid name must be specified.**

**dir1**         The orientation index of the subregion **idsub1** that points to the surface to be matched with the complimentary surface. See Section 3.5 for available choices. **This input is not required if idsub1 was defined by a LOCATE PAIR command.**

**idsub**         See Mode 1 Specification.

**fname**        If a file name is present then a file is generated that lists the elements and, for structured grids, a list of (I,J,K) grid indices. A valid name (see Section 3.3) must be specified.

#### **EXAMPLES**

---

**LOCAte** COMPlimentary ID=DMN1 as ID=DMN2

**LOCAte** COMPlimentary of X- direction of ID=DMN1 ID=DMN2

**MODE 19:** Specification of a General Second Order Surface

**SYNTAX** **LOCA** { **SURF** } { **BOUN** } { **N<sub>1</sub>, N<sub>2</sub>, , N<sub>n</sub>** } [ **TOLE** = {  $\epsilon$  } ] [ **ID=idsub** ] [ **fname** ]

**SURF** All boundaries of the elements in the vicinity of a general second order surface are selected. The 2D and 3D surfaces are specified, respectively, as:

$$C_0 = C_1 x + C_2 y + C_3 x^2 + C_4 y^2 + C_5 x y$$

$$C_0 = C_1 x + C_2 y + C_3 z + C_4 x^2 + C_5 y^2 + C_6 z^2 + C_7 x y + C_8 y z + C_9 z x$$

This specification results in the location of a boundary “Pair” type of subregion (see **LOCATE PAIR** mode) that consists of the element number and the element boundary nearest to the surface defined by one of the above relations. To determine if an element boundary is to be included as falling on the surface, each element in the domain is tested. If the defined surface falls on the line connecting an element node with its neighbor node, then the element boundary that falls between the two nodes is included in the definition.

**BOUN** If this modifier is present then only the external boundary surfaces of the computational domain that satisfy the definition are included in the list.

**N<sub>1</sub>,...,N<sub>n</sub>** These specify the coefficients C<sub>0</sub> through C<sub>5</sub> (for 2D) or C<sub>0</sub> through C<sub>9</sub> (for 3D). Any trailing coefficients not specified are assumed to be zero. At a minimum C<sub>0</sub> through C<sub>2</sub> (for 2D) or C<sub>0</sub> through C<sub>3</sub> (for 3D) must be specified.

**TOLE** A tolerance is required since type REAL numbers are compared to determine if the surface equation is satisfied. In FORTRAN REAL\*8 (64 bit) computations, the machine zero is on the order of 10<sup>-16</sup>. By default it is assumed that if the surface definition is satisfied within a margin of 10<sup>-15</sup>, then the point lies on the surface. If this modifier is present then the last value on the command is assumed to be the tolerance for judging the proximity to the specified surface.

$\epsilon$  The tolerance for judging the proximity to the surface. This numerical value must be present if the modifier **TOLERANCE** is specified. The use of values larger than 10<sup>-7</sup> is not recommended. The default value is 10<sup>-15</sup>.

**idsub** See Mode 1 specification.

**fname** If a file name is present then a file is generated that lists the element boundary pairs, and, for structured grids, a list of (I,J,K) grid indices. A valid name (see Section 3.3) must be specified.

**EXAMPLES**

```

LOCAte SURFACE 2 = 1. *x + 1. * y                                ! 2D mode
LOCAte external BOUNDary SURFACE 2 = 1. *x + 1. * y + 0. * z    ! 3D mode
LOCAte SURFACE 1 = -1. x -1. y -1. z + 1. xx + 1. yy +1. zz +0. xy -1. yz
LOCAte SURFACE 1 = -1. x -1. y -1. z + 1. xx + 1. yy +1. zz +0. xy -1.yz ID=SURF2 file='SURF2.LOC'
LOCAte BOUNDary SURFACE 1 = -1. -1. -1. + 1. + 1. +1. TOLERance=1.E-9 ID=SURF2
    
```

**MODE 20:** Matched Surface Subregion from Specified List

**SYNTAX** **LOCA** {**MATC**} [**ID=idsub** ] [**fname**] { **N<sub>1</sub>**|**N<sub>1</sub>,...,N<sub>4</sub>**, **N<sub>5</sub>, ... , N<sub>n</sub>**}

**MATC** A matched surface type of subregion is defined which is a special type of subregion consisting of two matched surfaces. It can only be used with the **TRANSFER** command that transports fluid flux along with specified fluid properties from one surface to another with or without transformation.

**idsub** See Mode 1 Specification.

**fname** If a file name is present then the file may be used for input or output. The **input mode** is activated if a file name is specified with **only one** numerical value (**N<sub>1</sub>**). The file may contain a header field with up to 100 lines (records) of information. **The header field, if present, must end with the record "END HEADER" and the numerical values must immediately follow this record. In the output mode**, a file is generated that lists the element and surface number pairs and, for structured grids, a list of (I,J,K) grid indices. In the file input or output mode, a valid name (see Section 3.3) must be specified.

**N<sub>1</sub>** If the file input mode is activated, then, **N<sub>1</sub>** denotes the number of matched sets of pairs of element and surface numbers to be read from the file. Each set of pairs must specify 4 values as explained in **N<sub>1</sub>,...,N<sub>4</sub>** below. If the file input mode is not activated, then **N<sub>1</sub>** is interpreted as defined in **N<sub>1</sub>,...,N<sub>4</sub>** below.

**N<sub>1</sub>,...,N<sub>4</sub>** A set consisting of 2 element and surface number pairs which define the 1<sup>st</sup> and 2<sup>nd</sup> surfaces to be matched. **N<sub>1</sub>** and **N<sub>2</sub>** define the 1<sup>st</sup> surface whereas **N<sub>3</sub>** and **N<sub>4</sub>** define the 2<sup>nd</sup> surface. The surfaces of the element are numbered from 1 to 4 in the 2D and 1 to 6 in the 3D mode. See Section 3.4 for further details of the manner in which these surface numbers are assigned.

**N<sub>5</sub>,...,N<sub>n</sub>** The matched sets of element and surface number pairs in the manner of **N<sub>1</sub>,...,N<sub>4</sub>** above.

## EXAMPLES

---

**LOCA**te MATCh LIST of values: (23,1) & (57,2); (25,3) & (67,4); (27,3) & (69,5) ID=MATCHED

**LOCA**te MATCh LIST (23,1) & (57,2); (25,3) & (67,4); (27,3) & (69,5) ID=MATCHED on file='match.fil'

**LOCA**te MATCh LIST 3 sets from file 'match.loc'

<b>MODE 21:</b>	<b>Matched Surface Subregion from Two Previously Defined Subregions.</b>
<b>SYNTAX</b>	<b>LOCA {MATC} {ID=idsub1} [dir1] {ID=idsub2} [dir2] [INTE] [ID=idsub] [fname]</b>
<b>MATC</b>	A matched surface type of subregion is defined which is a special type of subregion consisting of two matched surfaces. It can only be used with the <b>TRANSFER</b> command that transports fluid flux along with specified fluid properties from one surface to another with or without transformation.
<b>idsub1</b>	The name of the 1 <sup>st</sup> subregion to be operated upon. <b>There is no default value; a valid name must be specified.</b>
<b>dir1</b>	The orientation index of the subregion <b>idsub1</b> that points to the surface to be matched with a similar surface of subregion <b>idsub2</b> . See Section 3.5 for available choices. <b>This input is not required if idsub1 was defined by a LOCATE PAIR command.</b>
<b>idsub2</b>	The name of the 2 <sup>nd</sup> subregion to be operated upon. <b>There is no default value; a valid name must be specified.</b> However <b>idsub2</b> may be identical to <b>idsub1</b> , if required.
<b>dir2</b>	The orientation index of the subregion <b>idsub2</b> that points to the surface to be matched with a similar surface of subregion <b>idsub1</b> . See Section 3.5 for available choices. <b>This input is not required if idsub2 was defined by a LOCATE PAIR command.</b>
<b>INTE</b>	<b>By default the faces that are matched are the outward directed surfaces (dir1 and dir2 with normal pointing out of the body) of the subregions idsub1 and idsub2. If the INTERIOR modifier is present, then the interior surfaces of the idsub1 and idsub2 are matched with each other.</b>
<b>idsub</b>	See Mode 1 Specification.
<b>fname</b>	If a valid file name is present, then an ordered list of elements and surface numbers selected by the command are written to the file. For a structured grid, the grid indices (I,J,K) are also written to the file.

## EXAMPLES

---

**LOCAtE MATCh** X- direction of ID=DMN1 and X+ direction of ID=DMN1 as ID=MATCHED  
**LOCAtE MATCh** X- direction of ID=DMN1 and X- of ID=DMN2 INTERnal surfaces as ID=MATCH2  
**LOCAtE MATCh** X- direction of ID=DMN1 and X- of ID=DMN2 as ID=MATCH2 also file='MATCH.FIL'

<b>MODE 22:</b>	<b>Subregion by Union or Intersection of Multiple Subregions</b>
<b>SYNTAX</b>	<b>LOCA {UNIO INTE DIFF}[NOT] [EXCL] [FIEL BOUN] {ID=id1} {ID=id2} {ID=...} {ID=idn} [ID=idsub] [fname]</b>
<b>UNIO</b>	The new subregion is created from the union (elements that belong to either) of two or more previously specified subregions. <b>This is the default option.</b>
<b>INTE</b>	The new subregion is created from the intersection (only the elements that belong to more than one subregion) of previously specified subregions. <b>This takes priority over the UNION modifier.</b>
<b>DIFF</b>	This modifier selects the elements that are present only in the 1 <sup>st</sup> subdomain but not in any other specified subdomain. If the 2 <sup>nd</sup> and subsequent subdomains are subsets of the 1 <sup>st</sup> domain, then this is equivalent to the combination of <b>NOT INTERSECTION</b> modifiers. <b>This modifier takes priority over UNION or INTERSECTION modifiers.</b>
<b>NOT</b>	This modifier selects the union of multiple subregions minus their intersection (elements that belong to one and not more than one subdomain). <b>This modifier is effective only if the INTERSECTION modifier is also specified.</b>
<b>EXCL</b>	If the modifier is present, then the specified elements are excluded and the complimentary set in the computational domain (or the external boundary of domain if the subdomains are of <b>LOCATE PAIR</b> type) is selected. For example, if the union is selected, and this modifier is present, then all the elements that belong in the union are excluded and rest of the computational domain is selected. <b>This modifier is ignored if the MINUS modifier is present.</b>
<b>FIEL</b>	See Mode 1 Specification.
<b>BOUN</b>	See Mode 1 Specification.
<b>id1, id2,...idn</b>	The identification or name of the 1 <sup>st</sup> through the n <sup>th</sup> subregion to be operated upon. <b>There is no default value; a set of valid names for previously specified subregions must be given.</b> . If one sub-domain is of the surface or boundary type (see, e.g., <b>LIST PAIR</b> ), then all sub-domains must be of the same type. <b>No more than 256 sub-regions may be specified on one command.</b>
<b>idsub</b>	See Mode 1 Specification.
<b>fname</b>	If a valid file name is present, then an ordered list of elements and/or surface numbers selected by the command are written to the file. For a structured grid, the grid indices (I,J,K) are also written to the file.

## REMARKS

---

The union and intersection operations cannot be performed on subregions that were previously defined by a **LOCATE** command with **MATCHED PAIR**, **CORRELATED** or **INJECTION** modifiers.

## EXAMPLES

---

**LOCATE UNION** of **ID=DMN1** and **ID=DMN2**

**LOCATE UNION** of **ID=DMN1** and **ID=DMN2** as **ID=DMN3**

**LOCATE** DIFFERENCE of **ID=DMN1** by subtracting **ID=DMN2** and store as **ID=DMN3**

**LOCATE** INTERsection **ID=DMN1** and **ID=DMN2**

**LOCATE UNION** of subregions **ID=DMN1** and **ID=DMN2** **EXCLUDE** selection only **FIELD** elements

**LOCATE UNION** of **ID=DMN1** and **ID=DMN2**; **EXCLUDE** selected and name **ID=DMN12**

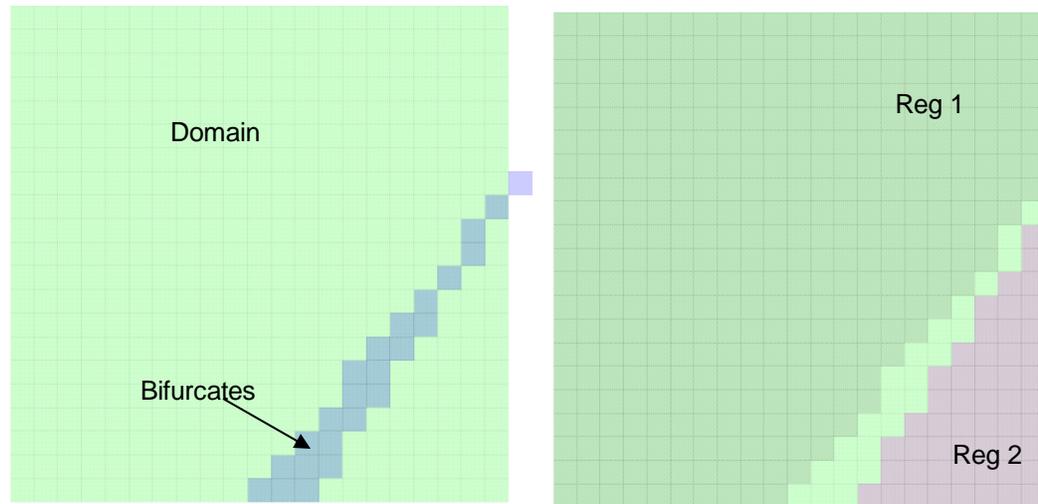
**LOCATE** **ID=DMN1** plus **ID=DMN2** select **INTERSECTION** & **EXCLUDE**

**LOCATE** **ID=DMN1** plus **ID=DMN2** select **NOT INTERSECTION** & **EXCLUDE** output on file 'DMN12.LOC'

**MODE 23:** Subregion Contiguous to a Boundary or Surface

**SYNTAX** **LOCA** {**CONT**} {**ID=idsub1**} {**ID=idsub2**} {**x<sub>0</sub>,y<sub>0</sub>, [z<sub>0</sub>]**} [**ID=idsub**] [**FIEL**] [**fname**]

**CONT** This modifier selects that part of the subregion **idsub1** that is bifurcated by the 2<sup>nd</sup> subregion (**idsub2**) that contains the coordinate specified by **x<sub>0</sub>,y<sub>0</sub>, [z<sub>0</sub>]**. The new subregion is created from the elements that belong to the 1<sup>st</sup> subregion and are contiguous to the 2<sup>nd</sup> subregion but only on the side of the 2<sup>nd</sup> subregion where the point specified by the **x<sub>0</sub>,y<sub>0</sub>,z<sub>0</sub>** coordinates is located. The nodes contained in the 2<sup>nd</sup> subregion will also be excluded. The 2<sup>nd</sup> subregion must bifurcate the 1<sup>st</sup> region completely into 2 parts for this command to be effective. If bifurcation is incomplete then all points minus the 2<sup>nd</sup> subregion will be selected. **There is no default value.**



**idsub1** The identification or name of the 1<sup>st</sup> subregion to be operated upon. This subregion must not be a boundary or surface type (such as **LOCATE PAIR**) of region. **There is no default value; a valid name must be specified.**

**idsub2** The identification or name of the 2<sup>nd</sup> subregion that divides the 1<sup>st</sup> subregion into two parts. It can be a region or a boundary or surface. **There is no default value; a valid name must be specified.**

**x<sub>0</sub>,y<sub>0</sub>, [z<sub>0</sub>]** The coordinates of the point that decides which part of the bifurcated region is to be selected. The elements selected will contain the node nearest to specified point and all nodes that are connected to it up to, but not including, the nodes in the 2<sup>nd</sup> subregion or nodes on the other side of the 1<sup>st</sup> subregion. **The coordinates must lie within the boundaries of the 1<sup>st</sup> subregion. Two values must be specified for 2D and three for 3D geometry. There is no default value.**

**idsub** See Mode 1 Specification.

**FIEL** See Mode 1 Specification.

**fname** If a valid file name is present, then an ordered list of elements and/or surface numbers selected by the command are written to the file. For a structured grid, the grid indices (I,J,K) are also written to the file.

**REMARKS**

If a **SCALE** command is specified immediately preceding this command, then **x<sub>0</sub>,y<sub>0</sub>, [z<sub>0</sub>]** values are scaled as specified in the **SCALE** command.

**EXAMPLES**

**LOCAte** CONTiguous part of ID=DMN1 bifurcated by ID=DMN2 and containing point (2.02, 1.002) **!2D**  
**LOCAte** CONTiguous section of ID=DMN1 bifurcated by ID=DMN2 with point (2.02, 1.002, 3.46) **!3D**

**MODE 24:**      **Subregion by Reassignment of a Previous Subregion**

**SYNTAX**      **LOCA** {**ID=idsub1**} [**EXCL**] [**ID=idsub**] [**FIEL**] [**fname**]

**idsub1**      The identification or name of the subregion to be operated upon. There is no default value; a valid name must be specified.

**EXCL**      By default a new subregion is defined which is identical to the previously specified subregion identified by **id=idsub1**. If this modifier is present, then the specified elements are excluded and the complimentary set in the computational domain is selected.

**idsub**      See Mode 1 Specification.

**FIEL**      See Mode 1 Specification.

#### **EXAMPLES**

---

**LOCA**te ID=DMN1 as new subdomain called ID=DMN2

**LOCA**te **EXCL**ude ID=DMN1 and define new complimentary set as ID=COMPDMN1

**LOCA**te **EXCL**ude ID=DMN1 and define new complimentary **FIEL**d only set as ID=COMPDMN1

**LOCA**te **EXCL**ude ID=DMN1 and define new complimentary **FIEL**d only set as active subregion

**MODE 25:** Modified Subregion with Post-Processing for Special Features

**SYNTAX** **LOCA** {**FLUI**|**SOLI**|**UNIQ**} { **ID=***idsub1* } [**ID=***idsub2*] [**FIEL**] [**fname**]

**FLUI** To modify or define a subregion so that only the fluid side elements are retained; any elements that are solid or blocked are eliminated. If the command defines a boundary surface, then only the surface numbers looking from the fluid side are retained; those that are defined by surface number for blocked or solid elements are eliminated.

**SOLI** To modify or define a subregion so that only the solid side elements are retained; any elements that are located in the fluid or unblocked region are eliminated. If the command defines a boundary surface, then only the surface numbers looking from the solid side are retained; those that are defined by elements and surface number pairs where the element is located in the fluid are eliminated.

**UNIQ** Any duplicate specifications of an element or boundary are removed; only the 1<sup>st</sup> occurrence is retained. If the subregion refers to a boundary segment, then this boundary is shared by two adjoining elements unless it is an exterior boundary of the domain. Any interior boundary can therefore be specified by two alternative pairs – each referring to one of the two elements that share that boundary. Mathematically this is Equivalent to specifying a surface and its direction normal pointing in one or the other opposite directions that are 180 degrees apart. Such duplicate specifications are also removed if this modifier is present.

**idsub1** The identification or name of the subregion to be operated upon. If **idsub2** is not present, then the existing **idsub1** sub domain will be modified. If **idsub2** name is present, the original **idsub1** domain is retained and modified sub-domain is defined with the new **idsub2** name. *There is no default value; a valid name must be specified.*

**idsub2** The identification name of the new subregion. If this modifier is present, then the original subdomain stays unmodified and a new subdomain is created with the modified features.

**FIEL** See Mode 1 Specification.

**fname** If a valid file name is present, then an ordered list of elements and/or surface numbers selected by the command are written to the file. For a structured grid, the grid indices (I,J,K) are also written to the file.

## REMARKS

---

This command can operate on either the element type of sub-domains (such as **LOCATE LIST** command) or boundary type of sub-domains (such as **LOCATE LIST PAIR**). If the **idsub1** specifies a set of elements, and **idsub2** is not present, then any boundary definition that was originally present is still retained. This implies that if a reference is made to a boundary of modified **idsub1** (e.g. a **BOUNDARY** command with the **dir** modifier), then the boundary of the original **idsub1** will be selected. If however **idsub2** is specified, then it will contain the definition of the modified boundary of the sub-domain unless the **FIELD** modifier is also specified on the command.

## EXAMPLES

---

**LOCAt**e SOLId elements ID=DMN1

**LOCAt**e FLUId elements of ID=DMN1 as ID=DMN2

**LOCAt**e UNIQue elements ID=DMN1

**LOCAt**e UNIQue boundary segments of ID=BOUN1 as ID=BOUN2

**MODE 26: Specification of Location of Injected Films and Curtains****SYNTAX** **LOCA** {**INJE**} [**ID=idsub1**] [**dir**] {**N<sub>1</sub>, ..., N<sub>n</sub>**} [**ID=idsub**] [**fname**]

**idsub1** The identification or name of the subregion to be operated upon. See Section 3.4. If no subregion is specified, the entire computational domain is selected. If **idsub1** identify is a surface subregion or if a “**dir**” is specified then the type of injection is automatically set in “**FILM** mode” else type is “**CURTAIN** mode” (see **INJECTION FILM** and **INJECTION CURTAIN** commands).

**dir** The orientation index for the boundary of the subregion to be selected. See Section 3.5 for available choices. If no “**dir**” is specified, all available “**dir**” for **idsub1** are selected.

**N<sub>1</sub>, ..., N<sub>n</sub>** **N<sub>1</sub>** through **N<sub>n</sub>** are the grid coordinates (x, y, z) for extremities of injection segment. A total of 4 numerical values for 2D and 6 for 3D input must be provided. See Section 3.4 for additional information. There is no default value; a valid set of values must be specified.

**idsub** See Mode 1 Specification.

**fname** See Mode 1 Specification.

**REMARKS**

The direction of the falling water film is defined by the direction of gravity. By default this is directed against the Z-axis of the domain.

The coordinates **N<sub>1</sub>, ..., N<sub>n</sub>** define two geometric points A and B. The segment AB must match points A and B that belong to the **idsub1 [dir]** surface. If this surface is not a plane then the segment AB may be a curve defined by the intersection between the **idsub1 [dir]** surface and the plane containing A and B with the normal vector::  $\overline{AB} \wedge (\overline{AB} \wedge \overline{g})$ .

**EXAMPLES**

**LOCAtE** INJEction on entire X- face and define ID=ZFILM

with injection from (0.0, 0.0, 1.8) to (0.0,5.0,2.0)

**LOCAtE** INJEction on ID=BLOCK1 Y+ and define ID=ZFILM

with injection from (2.0, 2.7, 0.0) to (5.0,2.7,2.0)

**LOCAtE** INJEction on entire domain and define ID=WCURTAIN

with injection from (2.0, 2.7, 0.0) to (5.0,2.7,2.0)

**LOCAtE** INJEction on ID=WALLRIGHT and define ID=WFILM

with injection from (2.0, 2.7, 0.0) to (5.0,2.7,2.0)

**LOCAtE** INJEction on SELEcted surface and define ID=WFILM

with injection from (2.0, 2.7, 0.0) to (5.0,2.7,2.0)

**LOCAtE** INJEction on SELEcted surface and define a automatically named subregion

with injection from (2.0, 2.7, 0.0) to (5.0,2.7,2.0)

**MODE 27: Moving or Evolving Rectangular Subregion by Grid Coordinates****SYNTAX** **LOCA** {**MOVE** | **MOVI**} {**COORD**} [**ID=idsub**] {**N<sub>1</sub>, ..., N<sub>n</sub>**} [**STOR = N<sub>add</sub>**] [**fname**]**MOVE** A rectangular window is specified by grid (x, y, z) coordinates that can evolve or move in time.**MOVI** Same as **MOVE** modifier.**COORD** The window is specified by 2 sets of (x, y, z) coordinates. The 1<sup>st</sup> set defines the “lower-left” corner and the 2<sup>nd</sup> set the “upper-right” corner of the window. A total of 4 numerical values for 2D and 6 for 3D geometry are required. There is no restriction on the coordinate values except that the “upper” values must be greater than the “lower” values.

The elements are selected based on the coordinate value of the element node. All elements with the node located inside or on the boundary of the window are selected by this command. An element is considered either in or out. No account is taken of the fact that the selected elements may be partially in or out.

**idsub** See Mode 1 Specification.**N<sub>1</sub>, ..., N<sub>n</sub>** N1 through Nn are the grid coordinates (x,y,z) for the “lower-left” and “upper-right” corners of the window (s). A total of 4 numerical values for 2D and 6 for 3D input must be provided. See Section 4.4 for additional information. **There is no default value; a valid set of values must be specified.****STOR** The moving window may evolve with time (see **MOVE** command). During this process elements may be added or removed from the window. At any given time, the set of elements that form the window and the immediate boundary nodes of the window (unless **FIELD** modifier is specified) are stored in memory. If the number of elements in the window grows, then additional memory is required to store new elements and boundary nodes. **By default the amount of additional memory allocated is set equal to the initial number of elements.** If the moving window is expected to increase beyond that, then this modifier along with the maximum additional memory must be specified.**N<sub>add</sub>** The additional memory for the moving window over and above that originally allocated for the window. If the specified number is larger than the number of elements in the entire computational domain, then the allocated amount is truncated to the number of total elements.**fname** See Mode 1 Specification.**REMARKS**

If a **SCALE** command is specified immediately preceding this command, then the input coordinate values are scaled as described in the **SCALE** command.

**EXAMPLES**

**LOCAte MOVING** region ID=MOVE\_FRAC with initial **COORD**inates (0.,40.0,4.40) to (45.,60.0,4.70)  
**LOCAte MOVING** ID=MOVE\_FRAC **COORD**inates (0.,40.0,4.40) to (45.,60.0,4.70) **STOR**=1000

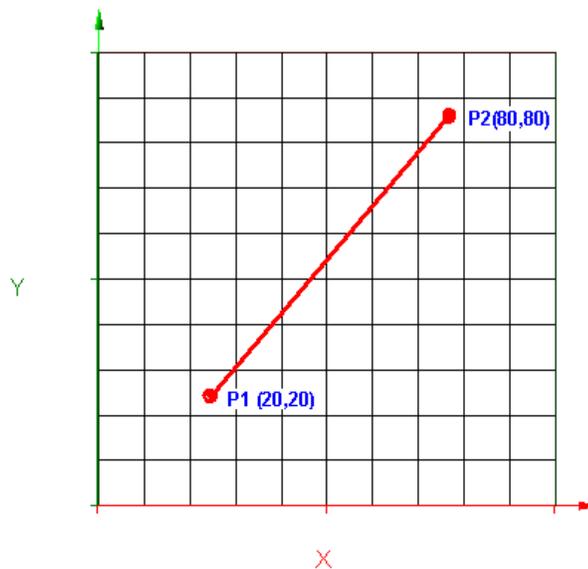
**MODE 28:** Stationary or Evolving Embedded Planar Fracture

**SYNTAX** **LOCA** {FRAC} [MOVE|MOVI] {ID=idsub} {N<sub>1</sub>, ..., N<sub>n</sub>}

**FRAC** A subregion of connected elements, through which a fracture passes is specified by (x, y, z) coordinates. The region can evolve or move in time if the **MOVE** modifier is present.. A **FRACTURE** command must be specified to activate the fracture. In addition one or more **MOVE** commands need to be specified if the fracture evolves in time.

The fracture is defined by the fracture plane specified by its coordinates. In 2D, the fracture plane is specified by exactly 2 sets of (x,y) points with a total of 4 numerical values. In 3D, the fracture plane can be any arbitrary simply connected planar polygon with non-intersecting edges. A minimum of 3 sets of (x,y,z) points with a total of 9 numerical values must be specified.. The points have to be specified in order around the polygon and must form a plane. If the specified points are non-planar, the results are unpredictable

The elements through which the fracture passes are selected based on the intersection of the fracture plane and the element faces. Elements are selected to create a contiguous path for flow within the fracture. An element is considered either in or out. No account is taken of the fact that the selected elements may be partially in or out.



**MOVE** This modifier designates a fracture that can move or evolve in time.

**MOVI** Same as **MOVE** modifier.

**idsub** See Mode 1 Specification.

**N<sub>1</sub>, ..., N<sub>n</sub>** The coordinates (x,y) or (x,y,z) for the points defining the fracture plane. For 2D geometry 4 numerical values must be specified. For 3D, a minimum of 9 values are required. There is no default value; a valid set of values must be specified.

## REMARKS

Currently this command is fully operational only in the 2D geometry mode. The 3D option is under development. Please contact ACRI for implementation of the 3D option. If a **SCALE** command is specified immediately preceding this command, then the input coordinate values are scaled as described in the **SCALE** command.

## EXAMPLES

**LOCA**te **FRAC**TURE ID=FRAC2D from (20,20) to (80,80) ! 2D fracture

**LOCA**te **MOVI**ng **FRAC**TURE ID=FRAC3D from (15,15,25), (35,15,25),(35,35,25),(15,35,25) !3D fracture

**5.63 MATERIAL**

**PURPOSE** Define material properties. This command is effective only for the **PORFLOW®** Software Tool.

**MODE 1:** **Specification of Material Density**

**SYNTAX** **MATE {DENS}**

**DENS** The solid or particle density of the material is specified.

**APPLICABILITY** 

---

This command is available only for the **PORFLOW®** Software Tool.

**REMARKS** 

---

This mode of the command has been superseded by the **DENSITY SOLID** command. Please refer to that command.

**MODE 2:**      **Specification of Material Porosity**

**SYNTAX**      **MATE {PORO}**

**PORO**          The material porosity is specified.

**APPLICABILITY** \_\_\_\_\_

This command is currently available only for the **PORFLOW®** Software Tool.

**REMARKS** \_\_\_\_\_

This mode of the command has been superseded by the **POROSITY** command. Please refer to that command.

**MODE 3:**      **Specification of Material Density and Porosity**

**SYNTAX**      **MATE** [**PROP**] { $\rho_s$ } [ $\Theta_E$ ,  $\Theta_T$ ,  $\Theta_D$ ] [**subrgn**]

**PROP**            The material properties are specified. In this mode both density and porosity are specified on one command. If this specification is omitted and the modifier TYPE is not present, and the number of numerical values on the command is less than 5, then the specification is assumed to be for material properties. **This mode of the command is being retained only for backward compatibility. Its function has been superseded by the DENSITY SOLID and POROSITY commands which are recommended for use instead of this mode of the command.**

$\rho_s$                 The density of a dry, solid component,  $\rho_s$  (> 0). The default value is 1.

$\Theta_E$               The effective (or flow) porosity,  $\Theta_E$ . The default value is 1.

$\Theta_T$               The total porosity,  $\Theta_T$ . If no value is specified, total porosity is set equal to the effective porosity.

$\Theta_D$               The diffusional porosity,  $\Theta_D$ . If no value is specified, diffusional porosity is set equal to the effective porosity.

#### **APPLICABILITY** \_\_\_\_\_

This command is currently available only for the **PORFLOW®** Software Tool.

#### **REMARKS** \_\_\_\_\_

This mode of the command has been superseded by the by the **DENSITY SOLID** and **POROSITY** commands. Please refer to those commands.

#### **EXAMPLES** \_\_\_\_\_

**MATE**rial density = 1; porosities: effective = 0.1, total 0.2, diffusive 0.15

**MATE**rial density 2200, porosities: 3\*0.15

**MATE**rial density 2200, porosities: 0.2, 0.25, 0.21

<b>MODE 4:</b>	<b>Material Type for a Subregion</b>
<b>SYNTAX</b>	<b>MATE</b> [TYPE] {N <sub>Mat</sub> } [subrgn] [FIEL] [dir]
<b>TYPE</b>	Identifies a material type with unique properties. <b>This mode of the command is being retained only for backward compatibility. It is no longer required. Its function has been superseded by the LOCATE command which is recommended for use instead of this mode of the command.</b>
<b>N<sub>Mat</sub></b>	A number that designates the material type or zone. A distinct number should designate each different material that has its own unique properties. <b>The default value is 1.</b>  For <b>PORFLOW®</b> , by default, the maximum assigned material type is limited to 100. If more than 100 material types are required, then the <b>ALLOCATE MATERIAL</b> command must be used to specify the maximum number. Though the material numbers may be assigned arbitrarily, most efficient use of memory results if these are defined sequentially.  For <b>TIDAL®</b> this number indicates the type of the element. A value of 0 indicates open water and value greater than 7 indicates land. A value of 4 implies that the water height is specified by the user and the fluid velocity at the open boundary of the element is computed from the condition of zero normal gradient.
<b>subrgn</b>	The subregion for which the input is specified. <b>If no subregion is specified, then the entire computational domain is selected.</b>
<b>FIEL</b>	Unless the <b>subrgn</b> was specified by a <b>LOCATE</b> command with the <b>FIELD</b> modifier, the values are set at all nodes in the subregion <b>and</b> any nodes at the <b>exterior</b> boundary that are right next to the <b>subrgn</b> (see the <b>LOCATE</b> command). If this modifier is present, then only the interior field nodes are set by the command
<b>dir</b>	By default, the input is applied to all the elements or nodes in the subregion defined by the <b>subrgn</b> modifier. If a modifier denoting a boundary orientation index is present, then the input is applied only to the nodes at the subregion boundary where the outward normal matches the specified modifier. See Table 3.5.1 for available choices.

**APPLICABILITY**

This command is currently used only by the **PORFLOW®** and **TIDAL®** Software Tools; it is not used by the **ANSWER®** Software Tool.

**REMARKS**

For **PORFLOW®** the material type is used to provide a unique numerical identity to each material type. It is used primarily only for output purposes. For **TIDAL®** the material designation is used to impose land or water boundary conditions for water height and velocity components.

**EXAMPLES**

**MATE**rial type 1! Total domain  
**MATE**rial type 5 as the currently **SELE**cted subregion  
**MATE**rial type 3 as subregion ID=COARSE\_SAND  
**MATE**rial type 5 for only the Y+ boundary of ID=GRAVEL  
**MATE**rial type 6 for only the **FIEL**D nodes of ID=GRAVEL

**MODE 5:**      **Material Type Data Input from a File**

**SYNTAX**      **MATE {fname}**

**fname**      The name of the file from which the material type information is obtained. See Section 3.3 for additional information. In this mode, the zone designation for the entire domain of computation must be read from the file sequentially in the manner of the increasing x, y and z grid nodes, in that order. This mode can also be used to provide input only for the porous matrix zone. Any fracture or borehole features must be defined explicitly by Mode 1 input.

#### **APPLICABILITY**

---

This command is currently available only for the **PORFLOW®** and **TIDAL®** Software Tools.

#### **EXAMPLES**

---

**MATE**rial type information from 'TYPE.DAT'

## 5.64 MATRIX

**PURPOSE** To select the method of solution for the matrix of algebraic equations.

**MODE 1:** ADI Matrix Solver

**SYNTAX** MATR {ADI} [NORM] [GLOB] [ $\Phi = N_1, \Phi = N_2, \dots, \Phi = N_n$ ] [ADI\_dir] [ADI\_options]

**ADI** Matrix is solved by an alternating direction implicit method. This is the default option.

**NORM** By default, the matrix is solved as computed where the coefficients of the matrix equations are in units of the computed fluxes. For example, if a species is in mass fraction units, the coefficients will be in units of  $[M t^{-1}]$ . If this modifier is present, then the matrix is normalized so that the diagonal coefficient is unity and other coefficients are non-dimensional.

**GLOB** By default, the specified solution method (ADI) and the NORM modifier, if specified, are applied only to the variable(s) that are specified on (or implied by) the command.. if the GLOB modifier is present, then the ADI and NORM are applied to all equations that are solved. If multiple problems are solved in sequence (See END command) then once specified, the designated option will also be applied to all subsequent problems that follow. Any specification of a new solver or normalization option through a new MATRIX command will be ignored.

$\Phi$  One or more symbols that denote the variable(s) for which ADI matrix solver will be invoked. Valid symbols are listed in Table 2.7.1.

$N_1, \dots, N_n$  Number of times the matrix is “swept” for the variable denoted by the symbol immediately preceding the value. The default value may vary with each installation though it is generally set to 1 for most of the variables.

**ADI\_dir** One or more of the characters: X, Y, Z to denote the direction in which the matrix sweeps are made. For example, a specification of X results in the matrix equations being solved for the x-direction nodes in increasing order of the I grid index, for fixed values of the J and K indices. By default, the matrix is swept in all active coordinate directions.

**ADI\_options** **OLD** By default the ADI rows are formed by starting from a boundary and searching for a string of element that connect that boundary with a boundary at an “opposite” face. If this modifier is present then an older element-based algorithm to be used to search for these strings. The use of this modifier is not recommended. It is retained only for compatibility with older versions.

**SING** The ADI row consists of a single string of meandering elements throughout the computational domain.

**COOR** The ADI rows are formed by a search based on physical coordinates rather than on the element connectivity. This is the default method for the Hybrid grids and the Gridless method.

**FACE** This modifier overrides the default COOR based search for the Hybrid grids and the Gridless method. The rows are formed from a face-based algorithm.

**REMARKS**

Each “sweep” through the matrix consists of one pass through the matrix of equations. A sweep is thus equivalent to a single inner iteration through the matrix described in Mode 2 of this command. During these sweeps the matrix coefficients stay fixed. The primary difference between a “sweep” and “Inner Iteration” is that during a sweep no checks are made to determine if the norm of the matrix residue meets any specified convergence criterion (CONVERGENCE command). The specified number of sweeps is always carried out. A sweep is more efficient than a full inner iteration because no matrix residue norms are computed. On the other hand a large number of sweeps may prove wasteful because, the sweeps are forced irrespective of the state of the solution.

**EXAMPLES**

MATRIX sweeps in X and Y directions: T=3! Sweep T equation 3 times

MATRIX sweeps: P=3, T=1, C=2 also perform REDBlack split

**MATR**ix for T to be solved 3 times by the ADI method with OLD row forming algorithm  
**MATR**ix for T ADI; REDBlack file for elements based on OLD method

<b>MODE 2:</b>	<b>SOR Matrix Solver</b>
<b>SYNTAX</b>	<b>MATR {SOR} [NORM] [GLOB] [EXPL] [<math>\Phi = N_1, \Phi = N_2, \dots, \Phi = N_n</math>]</b>
<b>SOR</b>	Matrix is solved by implicit successive over relaxation where values from the current iteration are used where available.
<b>NORM</b>	See Mode 1 specification.
<b>GLOB</b>	As for Mode 1 specification except that the <b>SOR</b> solver will be globally used.
<b>EXPL</b>	Matrix is solved by explicit successive over relaxation where only the values from the old iteration are used.
<b><math>\Phi</math></b>	One or more symbols that denote the variable(s) for which <b>SOR</b> matrix solver will be invoked. Valid symbols are listed in Table 2.7.1.
<b><math>N_1, \dots, N_n</math></b>	Number of times the matrix is “swept” for the variable denoted by the symbol immediately preceding the value. This input is ignored if the <b>EXPLICIT</b> modifier is present. The default value may vary with each installation though it is generally set to 1 for most of the variables.

## REMARKS

---

In general the **SOR** solver is not recommended for any except the simplest of problems. However, with the **EXPLICIT** modifier, it provides an effective method for time-explicit solution of a transient problem. It can also prove very useful to monitor the behavior and progress of the solution front in case of lack of convergence of a matrix.

See also comments under Mode 1 for importance of the numerical input given by  **$N_1, \dots, N_n$**  etc.

## EXAMPLES

---

**MATR**ix SOR for P=7 sweeps and T=3 sweeps  
**MATR**ix for P=3, T=2 SOR and generate REDBlack file  
**MATR**ix for T to be solved by EXPLicit SOR

**MODE 3: Built-In ACRi Matrix Solvers****SYNTAX** **MATR** { $\Phi$ } {**solver**} [**precon**] [**NORM**] [**GLOB**] [**N<sub>Degree</sub>** | **N<sub>Block</sub>**]

$\Phi$	One or more symbols that denote the variable(s) for which specified matrix solver will be invoked. Valid symbols are listed in Table 2.7.1.
<b>solver</b>	<p><b>BICG</b> Matrix is solved by the bi-conjugate gradient stabilized method.</p> <p><b>BICO</b> Matrix is solved by the bi-conjugate gradient stabilized method.</p> <p><b>CONJ</b> Matrix is solved by the conjugate gradient method. To be used for symmetric matrices only.</p> <p><b>CG</b> Matrix is solved by the conjugate gradient method. To be used for symmetric matrices only.</p> <p><b>CGNR</b> Matrix is solved by the conjugate gradient for normalized equations method. This method may be inferior to the BICG method.</p> <p><b>LUDE</b> Matrix is solved by sparse direct LU decomposition method. This method can be slow and memory intensive if the problem size exceeds 20000 elements. <b>MATRIX ITER</b> mode of the command is ineffective with this option.</p> <p><b>BLOC</b> Matrix is solved by a block by block version of sparse direct LU decomposition method. The main purpose is to overcome the performance limitations of the LUDE method. The block size can be specified on the same command.</p> <p><b>USER</b> Matrix is solved by an algorithm supplied by the user. See Section 4.7.</p>
<b>precon</b>	<p><b>NEUM</b> The default pre-conditioner for the <b>BICG</b>, <b>CG</b>, and <b>CGNR</b> methods. Uses a Neumann matrix polynomial (approximate inverse) pre-conditioner. The degree of the polynomial can be specified on the same command.</p> <p><b>SSOR</b> The symmetric <b>SOR</b> pre-conditioner for the <b>BICG</b>, <b>CG</b>, and <b>CGNR</b> methods.</p>
<b>NORM</b>	See Mode 1 specification.
<b>GLOB</b>	As for Mode 1 specification except that the <b>solver</b> , <b>precon</b> and <b>NORM</b> , (if present) will be globally used.
<b>N<sub>Block</sub></b>	The size of the block if <b>BLOC</b> modifier is present; the specified value must be greater than 99. The default value is 2000.
<b>N<sub>Degree</sub></b>	The degree of the polynomial if <b>NEUM</b> pre-conditioner is specified. The default value is 4.

**REMARKS**

Currently, the **CONJugate** gradient and **BICO**njugate gradient methods are optimized to work in co-operation with the CPU command. These methods will use multiple processors if the CPU command is used to specify the number of processors.

**EXAMPLES**

**MATR**ix for T USER option 5! Matrix Solver supplied by User  
**MATR**ix for T accelerator: BICOnjugate gradient preconditioner: NEUMann degree 7  
**MATR**ix for T accelerator: BICOnjugate gradient preconditioner: SSOR  
**MATR**ix for P by sparse LUDEcomposition (only for less than 20000 elements)  
**MATR**ix for P by BLOCk size=1000 elements  
**MATR**ix for P accelerator: CONJugate gradient preconditioner: SSOR (symmetric matrices only)

<b>MODE 4:</b>	<b>Number of Inner Matrix Iterations</b>
<b>SYNTAX</b>	<b>MATR {ITER} [MAX  MIN] {N1 <math>\Phi</math> =N<sub>1</sub>, <math>\Phi</math> =N<sub>2</sub>, ..., <math>\Phi</math> =N<sub>n</sub>} [NORM]</b>
<b>ITER</b>	The input pertains to the maximum or minimum number of inner iterations for the matrix. This mode of the command is ineffective if <b>MATRIX LUDE</b> command for the same variable is specified.
<b>MAX</b>	The input applies to maximum number of inner iterations for the matrix. By default the input pertains to the maximum number of inner iterations. If no maximum iterations for a variable are specified, then the maximum iterations are set equal to the maximum iterations specified on the <b>CONVERGENCE</b> command for the variable.
<b>MIN</b>	The input applies to minimum number of inner iterations for the matrix. If no minimum iterations for a variable are specified, then the minimum iterations are set equal to the minimum iterations specified on the <b>CONVERGENCE</b> command for the variable.
<b><math>\Phi</math></b>	Symbols that denote the variable(s) for which the <b>N<sub>1</sub>, N<sub>2</sub>, ...</b> , etc. are effective. Valid symbols are listed in Table 2.7.1. If no symbol is specified, <b>and only one numerical value is present</b> , then the input is applied to all variables for which a transport equation is solved.
<b>N<sub>1</sub>,..., N<sub>n</sub></b>	Number of times ( $\geq 1$ ) per time step that the matrix is solved for the variable denoted by the symbol immediately preceding the value. The default value is 1.
<b>NORM</b>	See Mode 1 specification.

**REMARKS**

Once the matrix coefficients are assembled, the matrix is solved iteratively till the specified convergence criterion (**CONVERGENCE** command) is satisfied. During this process the matrix coefficients stay constant even though the value of the computed variable is changing. This process of solution is here named the “**Inner Iteration**”. Each inner iteration consists of one pass through the matrix of equations followed by a check on the matrix residue. If the norm of the residue is larger than the specified tolerance, and the number of iterations is less than the maximum specified, then another pass follows.

**EXAMPLES**

**MATR**ix ITERations for all variables = 10  
**MATR**ix ITERations: P=8, T=2, C=2  
**MATR**ix MINIMUM ITERations for all variables = 3

**MODE 5:** Matrix Solvers from the University of Texas NSPCG Package

**SYNTAX** MATR {NSPC} [Φ] [precon] [accel] {ELIM [OFF]} [option] [NORM] [GLOB] [N<sub>degree</sub>]

**NSPC** One of the matrix solvers, consisting of a preconditioner and an accelerator, from the NSPCG package (NSPCG User's Guide Version 1.0, by T.C. Oppe, W.D. Joubert and D.R. Kincaid, Report No. CNA-216, April 1988, Center for Numerical Analysis, Univ. of Texas, Austin, TX 78713-8510) is used. The source code and manuals are available at <http://www.netlib.org/itpack>. A special agreement is required for use of this package.

**ELIM** Integer parameter IPARM (1) is set to 1. This implies that equations below a threshold level are dropped from the matrix system.

**ELIM OFF** Integer parameter IPARM (1) is set to 0. This turns off any previously specified ELIM modifier. This is also the default setting.

**Φ** One or more symbols that denote the variable(s) for which the specified options are effective. Valid symbols are listed in Table 2.7.1.

**precon** Preconditioner component of the Matrix Solver.

precon	INTERPRETATION
<b>NEUM</b>	Neumann matrix polynomial. This is the default option.
<b>LEAS</b>	Least squares matrix polynomial.
<b>DEGR</b>	Degree of NEUMann or LEASt squares polynomial (integer) default: 3
<b>REDU</b>	Reduced System preconditioner.
<b>CHOL</b>	Incomplete Cholesky Factorization.
<b>JACO</b>	The point Jacobi preconditioner.
<b>SOR</b>	The Successive Over-Relaxation preconditioner and accelerator

**accel** Accelerator Component of the Matrix Solver

accel	INTERPRETATION
<b>CONJ</b>	Conjugate Gradient accelerator.
<b>BCGS</b>	The Biconjugate Gradient Squared accelerator.
<b>GMRE</b>	GMRES accelerator. This is the default option.
<b>ORTH</b>	ORTHOMIN accelerator.
<b>CGNR</b>	Conjugate Gradient applied to Normal Equations.
<b>LANC</b>	Lanczos with ORTHOMIN accelerator.

**option** Modifier for the specification or matrix or solver method

option	INTERPRETATION
<b>MODI</b>	If <b>CHOLESKY</b> preconditioner is selected, then modified incomplete Cholesky decomposition is used; otherwise this input is ignored.
<b>PERM</b>	If <b>CHOLESKY</b> preconditioner is selected, then matrix is red-black permuted; otherwise this input is ignored.

**NORM** See Mode 1 specification.

**GLOB** As for Mode 1 specification except that the **precon**, **accel**, and **NORM**, (if present) will be globally used.

**TRUN** If **GMRE** modifier is specified, then the number of solution vectors saved is the smaller of the number of matrix iterations (**MATRIX ITER** command) and 200. If **TRUN** modifier is present, then the maximum of solution vectors is set to 5.

**N<sub>degree</sub>** The degree of the polynomial if **NEUMANN** modifier is present.

**EXAMPLES**

---

**MATRIX** for **P** by **NSPCG** (**PERMUTED CHOLESKY**).

**MATRIX** for **P** from **NSPCG** with **JACOBI** preconditioner and **BCGS** accelerator

**MATRIX U V W P** use **NSPCG** preconditioner: **NEUMANN** poly of **DEGREE 3 GMRES** accelerator

**MATRIX P NSPCG NEUMANN DEGREE 3 GMRES**

**MATRIX NSPCG P NEUMANN DEGREE 2 TRUNcated GMRES ELIMINATE** equations below threshold.

**MATRIX NSPCG** turn previously specified **ELIMINATE OFF**

**MODE 6:** Matrix Solvers from the HYPRE Library, Lawrence Livermore National Laboratory

**SYNTAX** MATR {HYPR} [ $\Phi$ ] [precon] [accel] [option] [VECT= N<sub>vector</sub>] [NORM] [GLOB]

**HYPR** One of the parallelized matrix solvers from the HYPRE package of the Lawrence Livermore National Laboratory (University of California), Livermore, California is used. A recent summary of the HYPRE library is described in: "Falgout, R.D. and Yang, U. M., "HYPRE: A Library of High Performance Preconditioners," in *Computational Science - ICCS 2002 Part III*, P.M.A. Sloot, C.J.K. Tan, J.J. Dongarra, and A.G. Hoekstra, Eds., Lecture Notes in Computer Science, vol. 2331, pages 632-641, 2002, Springer-Verlag. Also available as Lawrence Livermore National Laboratory technical report UCRL-JC-146175. The source code and manuals are available at <http://www.llnl.gov/CASC/hypre/>. A special agreement is required for use of this solver package.

$\Phi$  One or more symbols that denote the variable(s) for which the specified options are effective. Valid symbols are listed in Table 2.7.1.

**precon** Preconditioner component of the Matrix Solver.

precon	INTERPRETATION
<b>AMG</b>	Algebraic Multi-Grid Preconditioner. This is the default option.
<b>CHOL</b>	Incomplete Cholesky Factorization.
<b>DIAG</b>	Diagonal Scaling Preconditioner
<b>SPAR</b>	The Sparse Approximate Inverse Preconditioner.

**accel** Accelerator Component of the Matrix Solver

accel	INTERPRETATION
<b>AMGA</b>	The AMG accelerator.
<b>BICG</b>	The BI-Conjugate Gradient accelerator
<b>CONJ</b>	Conjugate Gradient accelerator.
<b>GMRE</b>	GMRES accelerator. This is the default option.

**option** Modifier for the specification of matrix or solver method

option	INTERPRETATION
<b>SYMM</b>	By default the matrix is assumed to be non-symmetric. If this modifier is present, then the matrix is assumed to be symmetric.
<b>INDE</b>	By default the matrix is assumed to be positive definite. If this modifier is present, then the matrix is assumed to be indefinite. This may be the case if Central Difference Integration method is used (see <b>INTEGRATION</b> Command)
<b>STAT</b>	By default the working memory required for the AMG method is set up for each solution level on the fly. However, if the matrix is not changing for a variable (such as for pressure equation under certain conditions) then the use of the <b>STATic</b> modifier will lead to a more efficient set up strategy. This modifier is ignored if any preconditioner other than AMG is specified.

**VECT** This modifiers defines the number of vectors in the Krylov space that are saved.

**N<sub>vector</sub>** The number of previous vectors that are stored to derive the new iterate of the solution. The typical values are between 5 and 20. The default is set at 5.

**NORM** See Mode 1 specification.

**GLOB** As for Mode 1 specification except that the **precon**, **accel**, and **NORM**, (if present) will be globally used.

**EXAMPLES**

---

**MATR**ix by HYPRe solver

**MATR**ix for P from HYPRe solver print detailed DIAGnostics

**MATR**ix for P from HYPRE with AMG preconditioner and CONJ accelerator and AMORtized set up

**MATR**ix for P is SYMMetric; use HYPRe with CHOLesky with CONJugate Gradient

**MATR**ix U V W use HYPRE: SPARse preconditioner with GMRES accelerator

**MATR**ix U V W use HYPRE CHOLesky with GMRES and 10 VECTors

**MODE 7:** Global Matrix Control Parameters

**SYNTAX** MATR [COEF=V<sub>Coef</sub>] [ZERO=V<sub>Zero</sub>] [NORM]

**COEF** The minimum value for the matrix diagonal coefficient is specified. The magnitude of the diagonal coefficient for any row of the matrix must be greater than zero otherwise the matrix is indeterminate. However due to limitations of digital arithmetic and round off, the value may become very small. This input allows the user to tune the minimum threshold to the machine accuracy. Currently this is applicable only to the **ADI** and **SOR** solvers.

**V<sub>Coef</sub>** The minimum threshold for any of the diagonal coefficients of the matrix of equations. The default value is 1.E-20.

**ZERO** The machine zero parameter is specified. This parameter is used by the **NSPCG** and **HYPRE** matrix solver packages to perform various numerical tests. In general, it treats any real number smaller in magnitude than the specified value as equivalent to machine zero. It is useful to modify this parameter in the instances where the right hand side of the equation system is less than the default value. If the tests indicate that normalized values of the right side of the matrix are smaller than this number then the matrix system may return without solving the system.

**V<sub>Zero</sub>** The numerical value for the machine zero. The default values are given in the table below.

1.192 x 10 <sup>-7</sup>	IEEE real with 32-bit precision.
2.22 x 10 <sup>-16</sup>	IEEE real with 64-bit precision. This is the default.
7.1 x 10 <sup>-15</sup>	Cray XMP.
1.49 x 10 <sup>-8</sup>	Dec 10 (single precision)
4.768 x 10 <sup>-7</sup>	IBM 370 / 158 (single precision)

**NORM** See Mode 1 specification.

**EXAMPLES**

---

**MATR**ix COEFFicient minimum value = 1.E-30

**MATR**ix machine ZERO =1.0E-300

! Above is useful if NSPCG thinks that the ||RHS|| < machine zero and returns prematurely.

**MATR**ix machine ZERO set to 1.0E-300, COEFFicient=1.E-30.

**MODE 8:** Tolerance Threshold for Minimum Value of Matrix Elements

**SYNTAX** **MATR** {**TOLE**} [**Φ**=**V**<sub>1</sub>, **Φ** =**V**<sub>2</sub>, ..., **Φ** =**V**<sub>n</sub>] [**NORM**]

**TOLE** For the **AMG** preconditioner in the **HYPRE** solver, this modifier controls the connections between matrix elements in constructing a coarse level. For other solvers, it is tolerance or threshold for the minimum value of matrix elements; in this case, matrix elements with numerical values below the specified tolerance are dropped.

**Φ** Symbols that denote the variable(s) for which the **V**<sub>1</sub>,**V**<sub>2</sub>,...**V**<sub>n</sub> etc. are effective. Valid symbols are listed in Table 2.7.1. If no symbol is specified, and only one numerical value is present, then the input is applied to all variables for which the **CHOLESKY**, **SPAR** or **AMG** methods have been selected (that is, the same numerical value is used for these methods if several of them are selected for different variables in a given simulation).

**V**<sub>1</sub>,**V**<sub>2</sub>,...**V**<sub>n</sub> If no value is specified, then any previous specification stays in effect. If only 1 value is specified, then it is applied to all specified variables. If more than 1 value is specified, then the number of values must equal the number of symbols on the command.

For the **SPAR** preconditioner, the typical values are between 0.01 and 0.1 and the default is set at 0.05. For the **CHOLESKY** preconditioner, the typical values are between 0.001 and 0.01 and the default is set at 0.01. For the **AMG** preconditioner, the default value is 0.25 for 2D problems and 0.5 for 3D problems.

**NORM** See Mode 1 specification.

#### EXAMPLES

---

**MATR**ix for P from HYPRE with AMG preconditioner and CG accelerator

**MATR**ix for U from HYPRE with SPAI preconditioner and GMREs accelerator

**MATR**ix for V and W from HYPRE with CHOLesky preconditioner and GMREs accelerator

**MATR**ix TOLERance for P=0.25, for U=0.1, for V=0.001 and for W=0.01

**MODE 9:** Number of levels for AMG and SPAR Preconditioners for HYPRE

**SYNTAX** **MATR** {**LEVE**} {**N<sub>Level</sub>**|**Φ=N<sub>Level1</sub>**, **Φ=N<sub>Level2</sub>**, ..., **Φ=N<sub>Leven</sub>**} [**NORM**]

**LEVE** For the **SPAR** preconditioner, this modifier controls the level of the pattern matrix. The pattern matrix is computed from the matrix **A** by dropping small amplitude elements. For the **AMG** preconditioner, this modifier controls the maximum number of coarse levels. This modifier is ignored for other preconditioners.

**N<sub>Level</sub>** The numerical value for the number of levels. For the **SPAR** preconditioner, values in the range of 1 to 3 are recommended and the default is set at 1. For the **AMG** preconditioner, values in the range of 10 to 15 are recommended and the default is set at 12. In this case, generally, the number of levels needs to be increased with the problem size.

**Φ** Symbols that denote the variable(s) for which the **N<sub>Level</sub>** numerical values are effective. Valid symbols are listed in Table 2.7.1. If no symbol is specified, **and only one numerical value is present**, then the input is applied to all variables for which the **SPAI** or **AMG** methods have been selected (that is, the same numerical value is used for both methods if several of them are selected for different variables in a given simulation).

**NORM** See Mode 1 specification.

### EXAMPLES

---

**MATR**<sub>ix</sub> LEVELS = 10 for all relevant solvers

**MATR**<sub>ix</sub> LEVELS U = 5, V=5, P=10 for

**MODE 10:** Maximum Number of Non-zero Elements for Any Row of Modified Matrix

**SYNTAX** **MATR** {**ELEM**} {**N<sub>0</sub>**|**Φ=N<sub>01</sub>**, **Φ=N<sub>02</sub>**, ..., **Φ=N<sub>0n</sub>**} [**NORM**]

**ELEM** This modifier controls the maximum number of non-zero elements that are to be retained in any row of the modified matrix during factorization. Currently it is applicable only to the **CHOLESKY** Preconditioner of the **HYPRE** package.

**N<sub>0</sub>** The maximum number of non-zero elements in any row of the modified matrix. Typical values are between 15 and 25 and the default is set at 15. This input is effective if no symbol is specified, and it is then applied to all variables for which the **CHOLESKY** method has been selected.

**N<sub>01</sub>, N<sub>02</sub>, N<sub>0n</sub>** As for **N<sub>0</sub>** above except that the input is effective if one or more variable (**Φ**) is specified.

**Φ** Symbols that denote the variable(s) for which the **N<sub>01</sub>, N<sub>02</sub>, N<sub>0n</sub>** are effective. Valid symbols are listed in Table 2.7.1.

**NORM** See Mode 1 specification.

### EXAMPLES

---

**MATR**ix maximum non-zero elements = 10 for all solvers

**MATR**ix non-zero **ELEM**ents for P=15 and for W=5

**MODE 11:** Maximum Number of Non-zero Elements for Any Row of Modified Matrix

**SYNTAX** **MATR** {RESI} [ $\Phi=V_1, \Phi=V_2, \dots, \Phi=V_n$ ] {ABSO} [NORM]

**RESI** For the **NSPCG** solvers, by default the specified convergence tolerance (**CONVERGENCE** command) is normalized with the larger of the machine zero or a norm of the current matrix residue. If this command along with the **ABSO** modifier is present, then the specified convergence tolerance is used as an absolute measure.

**$V_1, V_2, \dots, V_n$**  The specified matrix residue (>0) to determine if convergence is reached. If no value is specified, then any previous specification stays in effect. If only 1 value is specified, then it is applied to all specified variables. If more than 1 value is specified, then the number of values must equal the number of symbols on the command. The default value is set to equal that specified on the **CONVERGENCE** command.

**ABSO** The modifier to select the absolute measure of convergence tolerance.

**NORM** See Mode 1 specification.

### EXAMPLES

---

**MATRIX RESIDUE** in **ABSOLUTE** mode

**MATRIX RESIDUE** for **P = 1.E-8**

**MATRIX RESIDUE: T = 1.E-6, C = 1.E-4**

**MATRIX RESIDUE: T = 1.E-6, C = 1.E-4, C2 = 1.E-5** in **NORMALIZED** mode

**MATRIX RESIDUE: P, P2, T, S** is equal to **1.E-8** for all **ABSOLUTE** mode

**MATRIX RESIDUE 1.E-8** for all

**MATRIX** in **NORMALIZED** mode

**MODE 12:** Matrix Diagnostic Output for NSPCG and HYPRE Solvers

**SYNTAX** MATR [DIAG] [SUMM] [OFF] [NORM]

**DIAG** Produces detailed diagnostic output consisting of parameter values and informative comments from the matrix solver.

**DIAG SUMM** Produces diagnostic output in a compact and summarized form.

**DIAG OFF** Turns any previously specified diagnostic output off.

**NORM** See Mode 1 specification.

#### REMARKS

---

This mode of the **MATRIX** command may be combined with any of the other modes also. That is the modifiers above can also appear with any of the other previous modes.

#### EXAMPLES

---

**MATR**ix DIAGnostic output to be generated

**MATR**ix DIAGnostic SUMMary form output to be generated

**MATR**ix DIAGnostic output OFF from now on

## 5.65 META

**PURPOSE** To Generate Multiple Simulations form a Master File and a Meta Command File.

**SYNTAX** **META** {fname}

**fname** The name of a meta-command file that contains the commands that will replace the meta-commands in the master file to generate a sequence of simulations. This command must occur before the “**GRID**” command is encountered. The filename must be explicitly specified on the command; it cannot be specified through a **DEFINE** command.

**REMARKS**

This option works by replacing commands in a Master File with corresponding commands from a Meta-command File “**fname**”. Each meta-command contains a distinctive name that is enclosed between a starting and ending “%” character. The structure of the meta-command is:

**%metacomand1%** <valid ACRi FreeForm™ Command>

Where the **metacomand\_name** is an alphanumeric string with up to 32 characters and it is followed by a valid **ACRi FreeForm™** command. At run time the prefix **%metacomand\_name%** is stripped from the front of the command and the actual command becomes the effective input. Each meta-command in the Master File is replaced by a corresponding command starting with the same **metacomand\_name** in the Meta-command File **fname**. If no corresponding command is found in the Meta-command File, then the **ACRi FreeForm™** command in the Master File is retained. Each set of meta-commands in the **fname** terminates with the **END** command and the whole sequence is terminated if a **STOP** command is encountered. Thus the typical contents of a Meta-command File, **fname**, are as follows:

```
%metacomand1% <valid ACRi FreeForm™ Command>
%metacomand2% <valid ACRi FreeForm™ Command>
END
%metacomand1% <valid ACRi FreeForm™ Command>
END
%metacomand1% <valid ACRi FreeForm™ Command>
%metacomand3%<valid ACRi FreeForm™ Command>
STOP
```

Consider, for example, a set of statements in the Master file:

```
TITLE: Test Problem for Parametric Analysis by Creating Multiple sets of Input Data
META Command file “meta_commands.fil”
GRID: 22 in x and 12 in y direction
%1% BOUNDARY T at X- value = 100 !for Master File Problem
(Other input commands follow here)
%DENSITY%DENSITY = 990 !for Master File Problem
(Other input commands follow here)
SOLVE in STEADY mode for maximum of 1000 steps
%SAVE% SAVE in file ‘problem_1.sav’ !for Master File Problem
END
```

Assume now that the commands in the file “**meta\_commands.fil**” are:

```
%1% BOUNDARY T at X- value = 100 !for 2nd problem
%DENSITY% DENSITY = 1000 !for 2nd problem
%SAVE% SAVE in file ‘problem_2.sav’ !for 2nd problem
END
%1%BOUNDARY T at X- value = 120 !for 3rd problem
%SAVE%SAVE in file ‘problem_3.sav’ !for 3rd problem
END
%1%BOUNDARY T at X- value = 150 !for 4th problem
%DENSITY%DENSITY = 990 !for 4th problem
%SAVE%SAVE in file ‘problem_4.sav’ !for 4th Problem
STOP
```

The above Meta-command File will then generate 4 simulations; the 1st simulation is that defined in the

Master File and the other 3 are those in the Meta-command File. The resulting “unfolded” data set is saved in a file that has the same name as the Master File but with “\_META\_UNFOLD.DAT” appended to the file name. For the above example, this file shall contain 4 problems with input as follows:

```
! ** Unfolded ** META Command file “meta_commands.fil”
!
*****
! Problem # 1 created from Meta Command File
!*****
TITLE: Test Problem for Parametric Analysis by creating Multiple sets of Input Data
METACCommand file “meta_commands.fil”
GRID: 22 in x and 12 in y direction
BOUNDARY T at X- value = 100      !for Master File Problem
(Other input follows here)
DENSITY = 990      !for Master File Problem
(Other input follows here)
SOLVE in STEADY mode for maximum of 1000 steps
SAVE in file ‘problem_1.sav’      !for Master File Problem
END
!
!*****
! Problem # 2 created from Meta Command File
!*****
TITLE: Test Problem for Parametric Analysis by creating Multiple sets of Input Data
METACCommand file “meta_commands.fil”
GRID: 22 in x and 12 in y direction
BOUNDARY T at X- value = 100      !for 2nd problem
(Other input follows here)
DENSITY = 1000      !for 2nd d problem
(Other input follows here)
SOLVE in STEADY mode for maximum of 1000 steps
SAVE in file ‘problem_2.sav’      !for 2nd problem
END
!
!*****
! Problem # 3 created from Meta Command File
!*****
TITLE: Test Problem for Parametric Analysis by creating Multiple sets of Input Data
METACCommand file “meta_commands.fil”
GRID: 22 in x and 12 in y direction
BOUNDARY T at X- value = 120      !for 3rd problem
(Other input follows here)
DENSITY = 990      !for Master File Problem
(Other input follows here)
SOLVE in STEADY mode for maximum of 1000 steps
SAVE in file ‘problem_3.sav’      !for 3rd Problem
END
!
!*****
! Problem # 4 created from Meta Command File
!*****
TITLE: Test Problem for Parametric Analysis by creating Multiple sets of Input Data
METACCommand file “meta_commands.fil”
GRID: 22 in x and 12 in y direction
BOUNDARY T at X- value = 150      !for 4th problem
(Other input follows here)
DENSITY = 990      !for 4th problem
(Other input follows here)
SOLVE in STEADY mode for maximum of 1000 steps
SAVE in file ‘problem_4.sav’      !for 4th Problem
END
```

**EXAMPLES**

---

**META** command file "meta\_command.fil"

## 5.66 MOVE

**PURPOSE** To specify a moving coordinate system for the system geometry. This command is effective only for the **ANSWER®** and **PORFLOW®** Software Tools.

**MODE 1:** Grid Coordinate Frame in Uniform Translation

**SYNTAX** MOVE {dir} {V<sub>grid</sub>} [ABSO|RELA] [MASS] [MOME]

**dir** One of the **X**, **Y**, **Z**, **R** or **THETA** modifiers that, respectively, denotes the x, y, z, r or  $\theta$  coordinates for the axis along which the coordinate system is moving at a uniform speed. There is no default value; a value must be specified.

**V<sub>grid</sub>** The velocity of translation of the grid frame in the direction of the specified axis

**ABSO** By default the numerical solution proceeds in terms absolute velocity – that is the velocity observed by a stationary observer. In this case all input must be specified in terms of absolute velocity. The component that is moving must be specified as a moving block (**BLOCK** command with **MOVE** modifier). The velocity of the moving component should be specified in the same sense as the direction of rotation so the moving component is stationary in the rotating coordinate system.

**RELA** The solution is in terms of velocity relative to the rotating coordinate system. In this case all input must be specified in terms of relative velocity. The moving component must be specified as stationary and all initial and boundary conditions must be specified in terms of relative velocity. The velocity of the system boundaries should be specified in a sense opposite to that implied by the direction of rotation so the stationary fluid moves opposite to the rotating coordinate system.

**MASS** By default the mass flux at the moving boundary is added to the mass continuity equation. This is required by the governing equations. If this modifier is present, then the mass flux at the moving boundary is not added. Its effect is to relieve local pressure changes next to moving body. The use of this modifier is intended as a debugging device and is not generally recommended.

**MOME** By default the momentum flux at the moving boundary is automatically accounted for by the moving mass flux at the moving boundary. This is implied by the governing equations. If this modifier is present, then the momentum flux at the moving boundary is explicitly added. This for example may be necessary if **MASS** modifier is present. The use of this modifier is intended as a debugging device and is not generally recommended.

**REMARKS**

The **ABSOLute** velocity mode is recommended for most applications. In general it depicts better numerical behavior. The **RELAtive** mode is prone to poor numerical behavior especially for problems with high rotation speed or large domains since the relative velocity grows without bounds away from the body.

**EXAMPLES**

**MOVE** coordinate frame along X-axis at 15 meters per second

**MOVE** coordinate frame along X-axis at 15 meters per second in Relative mode.

**MOVE** coordinate frame along X-axis at 15 meters per second do not add MASS; add MOMEtum.

**MODE 2:**      **Grid Coordinate Frame in Pure Rotation**

**SYNTAX**      **MOVE {dir} {ROTA} {ω} [RPM] [ABSO|RELA] [MASS] [MOME]**

**dir**            **One** of the **X, Y, Z, R** or **THETA** modifiers that, respectively, denotes the x, y, z, r or **θ** coordinates for the axis around which it is rotating. **There is no default value; a value must be specified.**

**ROTA**            The coordinate frame is in simple rotation around the specified axis.

**ω**                The speed of rotation in radians per second or revolutions per minute. The value is positive if the axis is rotating in anti-clockwise sense for a right-handed coordinate system. For example for rotation around z-axis, the x and y components of velocity of rotation of the coordinate frame are defined by:

$$u = -\omega y; \quad v = \omega x$$

**RPM**            **By default the rotation speed is assumed to be in radians per second.** If this modifier is present then the speed is taken to be in revolutions per minute.

**ABSO**            See Mode 1 of the command.

**RELA**            See Mode 1 of the command.

**MASS**           See Mode 1 of the command.

**MOME**           See Mode 1 of the command.

#### REMARKS

---

The **ABSOLUTE** velocity mode is recommended for most applications. In general it depicts better numerical behavior. The **RELATIVE** mode is prone to numerical diffusion especially for problems with high rotation speed or large domains since the relative velocity grows without bounds away from the body.

#### EXAMPLES

---

**MOVE** grid by ROTAtion around Z-axis at 2.5 radians per second

**MOVE** coordinate frame by ROTAtion around Z-axis at 1000 RPM.

**MOVE** coordinate frame by ROTAtion around Z-axis at 2.5 radians per second in RELAtive velocity mode.

**MODE 3:** A Moving or Evolving Rectangular Subdomain

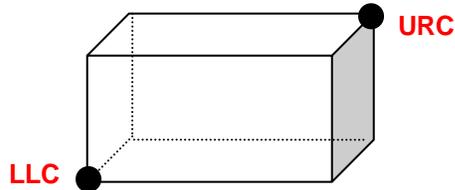
**SYNTAX** MOVE {ID=subrgn} {LLC|URC} {X|Y|Z} {func [time]}

**ID** The modifier to indicate that the input is for a previously defined subregion.

**subrgn** The rectangular subregion for which the input is specified. The subregion must have been previously specified with a **LOCATE MOVE COORDINATE** command.

**LLC** The input pertains to the coordinates of the “lower left” corner of the subregion.

**URC** The input pertains to the coordinates of the “upper right” corner of the subregion.



**X|Y|Z** The **LLC** or **URC** coordinate for which the functional input is specified.

**func** One of the modifiers listed in Table 4.2.1, which denotes the functional form of the dependent variable. If no function is specified, the value is assumed to be constant.

**time** The independent variable for the function.

#### REMARKS

---

This mode of the command allows the input of an evolving or moving rectangular domain. The domain is always assumed to be the enveloping rectangle of the **LLC** and **URC** set of coordinates at any time. Multiple command may be specified for the same domain with any of the **X**, **Y** or **Z** modifiers.

#### EXAMPLES

---

**MOVE** ID=MOVING\_REGION LLC coordinates: LINEAR function  $X = 0. + 0.001 * TIME$

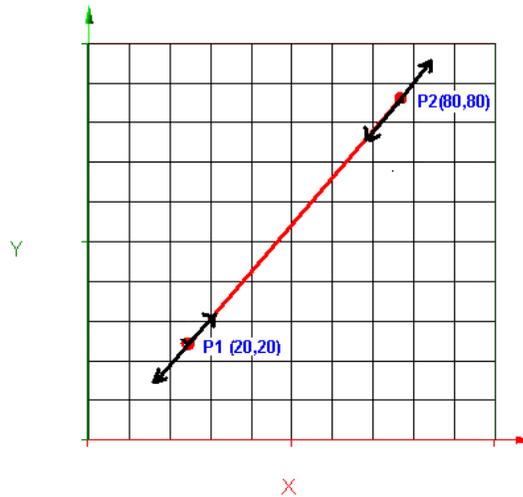
**MOVE** ID=MOVING\_REGION LLC coordinates: LINEAR function  $Y = 0. + 0.0001 * TIME$

**MOVE** ID=MOVING\_REGION URC coordinates:  $Y = 1. EXP ( 0.1 * TIME + 0. )$

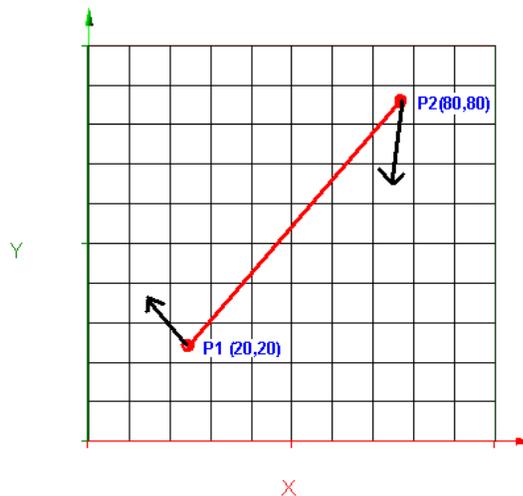
**MODE 4:** A Moving or Evolving Planar Fracture

**SYNTAX** MOVE {ID=subrgn} {N1} {X|Y|Z} {func [TIME]}

- ID** The modifier to indicate that the input is for a previously defined subregion.
- subrgn** The fracture subregion for which the input is specified. The subregion must have been previously specified with a **LOCATE MOVE FRAC** command.
- N1** The input pertains to the N<sup>th</sup> point defining the plane in order in which the points are specified by the **LOCATE MOVE FRAC** command. N1 can only be 1 or 2 for a 2D fracture. For 3D, N1 is between 1 and the number of points defining the fracture plane.
- X| Y| Z** The input is specified for the corresponding x, y or z coordinates of the **N1** point. Multiple commands must be specified for movement of each of the x, y or z coordinate.
- func** One of the modifiers listed in Table 4.2.1, which denotes the functional form of the dependent variable. Currently only functions of TIME can be specified. **The combination of functions must ensure that the vertices of the fracture always define a plane.**



**The combination of functions must also ensure that the orientation of the fracture is not changed.** The image above shows a valid moving fracture, with movement restricted to the plane of the fracture. The moving fracture shown below changes in orientation and so it is **not a valid movement for the fracture.**



**TIME** The independent variable for the function. Currently only functions of time can be specified. **This modifier must be specified.**

**REMARKS**

This mode of the command allows the input of an evolving or moving fracture. The points defining the fracture

can only move in the fracture plane. Multiple commands may be specified for the same domain with any of the **X**, **Y** or **Z** modifiers, but the vertices must only move in the plane of the fracture and not out of the plane. The movement of the vertices is restricted by the condition that the orientation of the fracture plane must not change.

**Currently this option is fully operational only for the fracture in a 2D geometry. Please contact ACRi for 3D implementation.**

#### EXAMPLES

---

**MOVE** ID=MOVING\_FRAC2D point no 1 coordinates: LINEAR function  $X = 0. + 0.001 * TIME$   
**MOVE** ID=MOVING\_FRAC3D point no 4 coordinates: LINEAR function  $Z = 0. + 0.0001 * TIME$   
**MOVE** ID=MOVING\_FRAC point no 2 coordinates:  $Y = 1. EXP ( 0.1 * TIME +0. )$

5.67 MULTIPHASE

**PURPOSE** To specify the nature of the characteristic curve and the values of the empirical characteristic constants for multiphase or variably saturated flow. This command automatically initiates the multi-phase mode of simulation. This command is effective only for the **PORFLOW®** Software Tool.

**MODE 1:** The Brooks & Corey Soil Characteristic

**SYNTAX** **MULT** {**BROO**}|**SATU**|**COND**|**BURD**|**MUAL**|**COMP** [**phase**] { $\lambda$ ,  $\Psi_a$ } [**S<sub>r</sub>**, **S<sub>g</sub>**, **k<sub>rmin</sub>**] [**subrgn**]

**BROO** The fluid phase saturation and the capillary pressure are related according to the Brooks and Corey expression given below.

$$\hat{S}^n = (\Psi_a / \Psi^n)^\lambda ; \Psi > \Psi_a$$

$$\hat{S}^n = 1 ; \Psi \leq \Psi_a$$

$$\hat{S}^1 = \frac{S^1 - S_r}{1 - S_r - S_g} ; \hat{S}^2 = \frac{S^1 + S^2 - S_r}{1 - S_r - S_g} ; \hat{S}^3 = 1$$

$$\Psi^n = P^n - P^{n+1}$$

$$\sum_{n=1}^N S^n = 1$$

In these relations, the superscript “n” denotes the n<sup>th</sup> fluid phase,  $\hat{S}$  is the normalized saturation,  $\Psi$  is the capillary pressure, P is the computed fluid pressure, and  $\Psi_a$ ,  $\lambda$ ,  $S_r$  and  $S_g$  are empirical constants as defined below. The relations above assume a 3-phase system (such as water-oil-gas); the same relations apply to a 2 phase system except that the 2<sup>nd</sup> (rather than the 3<sup>rd</sup>) phase normalized saturation is then unity.

**SATU** By default the moisture characteristic is used to compute both phase saturation and relative hydraulic conductivity. If the modifier **SATU** is present, the specification is used only to calculate the phase saturation.

**COND** By default the moisture characteristic is used to compute both phase saturation and relative hydraulic conductivity. If the modifier **COND** is present, the specification is used only to calculate the relative conductivity.

**BURD** The relative conductivity,  $k_r$ , is computed from the Burdine (1953) predictive model which leads to the equations given below. This is the default option.

$$k_r^n = (\hat{S}^n - \hat{S}^{n-1})^2 [(\hat{S}^{n-1})^{(1+2/\lambda)} - (\hat{S}^n)^{(1+2/\lambda)}] ; n > 1$$

$$k_r^1 = (\hat{S}^1)^{(3+2/\lambda)} ; n = 1$$

**MUAL** The relative conductivity,  $k_r$ , is computed from the Mualem (1976) model which leads to:

$$k_r^n = (\hat{S}^n - \hat{S}^{n-1})^{1/2} [(\hat{S}^{n-1})^{(2+2/\lambda)} - (\hat{S}^n)^{(2+2/\lambda)}] ; n > 1$$

$$k_r^1 = (\hat{S}^1)^{(5/2+2/\lambda)} ; n = 1$$

**COMP** If the modifier **SECOnd** is also present then  $k_r$  for the second phase is computed from the complimentary function:

$$k_r^2 = 1 - k_r^1$$

**phase** The fluid phase for which the input is specified. See Section 3.6 for available options. If no phase modifier exists, the input is assumed to be for the first phase of the fluid.

**$\lambda$**  The exponent (> 0) for Brooks and Corey relation. There is no default value.

$\Psi_a$	The so-called air entry pressure ( $> 0$ ) for Brooks and Corey relation. There is no default value.
$S_r$	An empirical fitting constant ( $\geq 0$ ) sometimes also called the residual saturation for the porous matrix. The default value is 0.
$S_g$	An empirical fitting constant ( $\geq 0$ ) sometimes also called residual gas-phase saturation. For a flow with 2 phases, this is always zero. The default value is 0.
$k_{rmin}$	The minimum value of $k_r$ . The default value is $10^{-6}$ .
subrgn	The subregion for which the input is specified. See Section 3.4 for more details. If no subregion is specified, then entire computational domain is selected.

## EXAMPLES

---

**MULT**iphase flow: BROOKs & COREY: lamda = 0.5; air entry p = 0.1

**MULT**iphase: BROOKs with MUALem: lamda=4.5, h\* = 5.4, 2\*0

**MULT**iphase: BROOKs for SECONd phase with: N=3.5; Aow=0.100

**MULT**iphase: BROOKs CONDUCTivity SECONd phase: lamda=3, psi=0.1

**MULT**iphase: BROOKs CONDUCTivity COMPLimentary SECONd l=3, psi=0.2

**MODE 2:** The Van Genuchten Characteristic

**SYNTAX** MULT {VAN} [SATU | COND] [MUAL | BURD] [COMP] [phase] {N, α} [S<sub>r</sub>, S<sub>g</sub>, M, k<sub>rmin</sub>] [subgrn]

**VAN** The saturation-capillary pressure characteristic is defined by the van Genuchten relations as shown below.

$$\hat{S}^n = \frac{1}{\left[1 + (\alpha \Psi^n)^N\right]^M} ; \Psi^n > 0$$

$$\hat{S}^n = 1 ; \Psi^n \leq 0$$

$$\hat{S}^1 = \frac{S^1 - S_r}{1 - S_r - S_g} ; \hat{S}^2 = \frac{S^1 + S^2 - S_r}{1 - S_r - S_g} ; \hat{S}^3 = 1$$

$$\Psi^n = P^n - P^{n+1}$$

$$\sum_{n=1}^N S^n = 1$$

In these relations, the superscript “n” denotes the n<sup>th</sup> fluid phase,  $\hat{S}$  is the normalized saturation,  $\Psi$  is the capillary pressure, P is the computed fluid pressure, and  $\alpha$ , N, M, S<sub>r</sub> and S<sub>g</sub> are empirical constants as defined below. The relations above assume a 3-phase system (such as water-oil-gas); the same relations apply to a 2 phase system except that the 2<sup>nd</sup> (rather than the 3<sup>rd</sup>) phase normalized saturation is then unity.

**SATU** See Mode 1 specification.

**COND** See Mode 1 specification.

**MUAL** k<sub>r</sub> is computed from the Mualem (1976) model. This is the default option.

$$k_r^n = (\hat{S}^n - \hat{S}^{n-1})^{1/2} \left[ \{1 - (\hat{S}^{n-1})^{1/M}\}^M - \{1 - (\hat{S}^n)^{1/M}\}^M \right]^2 ; n > 1$$

$$k_r^1 = (\hat{S}^1)^{1/2} \left[ \{1 - \{1 - (\hat{S}^1)^{1/M}\}^M\} \right]^2 ; n = 1$$

By default: M=1-1/N for this choice; it is overwritten if M is specified below.

**BURD** k<sub>r</sub> is computed from the Burdine (1953) predictive model.

$$k_r^n = (\hat{S}^n - \hat{S}^{n-1})^2 \left[ \{1 - (\hat{S}^{n-1})^{1/M}\}^M - \{1 - (\hat{S}^n)^{1/M}\}^M \right]^{1/2} ; n > 1$$

$$k_r^1 = (\hat{S}^1)^2 \left[ \{1 - \{1 - (\hat{S}^1)^{1/M}\}^M\} \right] ; n = 1$$

By default: M=1-2/N for this choice; it is overwritten if M is specified below.

**COMP** See Mode 1 specification.

**phase** See Mode 1 specification.

**N** The N (> 0) of the equation above. There is no default value for this input.

**α** The α (> 0) of the above equation. There is no default value for this input.

**S<sub>r</sub>** An empirical fitting constant (≥ 0) sometimes also called the residual saturation for the porous matrix. The default value is 0.

**S<sub>g</sub>** An empirical fitting constant (≥ 0) sometimes also called residual gas-phase saturation. For a flow with 2 phases, this is always zero. The default value is 0.

**M** The exponent, M, in the above equation. The default value is 1-1/N if the Mualem model is selected and 1-2/N if the Burdine model is selected.

**k<sub>rmin</sub>** The minimum value of k<sub>r</sub>. The default value is 10<sup>-6</sup>.

**subrgn** The subregion for which the input is specified. See Section 3.4 for more details. If no subregion is specified, then entire computational domain is selected.

#### EXAMPLES

---

**MULT**iphasе flow VAN Genuchten: n = 2.0, alpha = 0.5

**MULT**iphasе: VAN Genuchten BURDine: N=3.5; Alf=0.167, sr=0.067, sg=0.090

**MULT**iphasе: VAN Genuchten: N=3.5 a=0.167 sr=0.067 Sg=0.09, M=0.5 kr=0.02

**MULT**iphasе: VAN Genuchten for CONDuctivity only with: N=3.5; Aow=0.1

**MULT**iphasе: VAN Genuchten COND SECOnd COMPlimentary: N=3.5; a=0.1

<b>MODE 3:</b>	<b>Tabulated Data Input</b>
<b>SYNTAX</b>	<b>MULT</b> { <b>TABL</b> } [ <b>COND</b> ] [ <b>HEAD</b> ] [ <b>COMP</b> ] [ <b>phase</b> ] { <b>N<sub>sets</sub></b> } { <b>N<sub>1</sub>, ..., N<sub>n</sub></b>   <b>fname</b> } [ <b>subrgn</b> ]
<b>TABL</b>	The moisture characteristic is specified in tabular form.
<b>COND</b>	By default, the tabulated data are for the S- $\Psi$ characteristic. If the modifier <b>CONDUCTIVITY</b> is present, the specification is assumed to be for the S- $k_r$ characteristic.
<b>HEAD</b>	By default, the relative moisture characteristic is assumed to be specified as S vs. $k_r$ . If the modifier <b>HEAD</b> is present, the characteristic is assumed to be tabulated values of $\Psi$ vs. $k_r$ .
<b>COMP</b>	See Mode 1 specification.
<b>phase</b>	See Mode 1 specification.
<b>N<sub>sets</sub></b>	The number of sets of tabulated values which follow.
<b>N<sub>1</sub></b>	The first value of S for the S- $\Psi$ or S- $k_r$ characteristic or the first value of the $\Psi$ for the $\Psi$ - $k_r$ characteristic. The numerical value must be $\geq 0$ .
<b>N<sub>2</sub></b>	The first value of $\Psi$ for the S- $\Psi$ or the first value of $k_r$ for the S- $k_r$ or $\Psi$ - $k_r$ characteristic. The input value may be scaled internally by the <b>SCALE</b> command. The numerical value must be $\geq 0$ .
<b>N<sub>3</sub>, ..., N<sub>n</sub></b>	The sets of values of S, $\Psi$ or $k_r$ in a manner similar to <b>N<sub>1</sub></b> and <b>N<sub>2</sub></b> . A total of <b>N<sub>sets</sub></b> sets (including <b>N<sub>1</sub></b> and <b>N<sub>2</sub></b> ) must be specified.
<b>fname</b>	The name of the file from which numerical values <b>N<sub>1</sub></b> through <b>N<sub>n</sub></b> for the moisture characteristic curve are read. See Section 3.3 for additional information.
<b>subrgn</b>	The subregion for which the input is specified. See Section 3.4 for more details. If no subregion is specified, then entire computational domain is selected.

## EXAMPLES

**MULT**iphase flow: TABLE of 4 sets: (0,1.E6), (0.1,1.E3), (0.9,1.E2), (1.,1.)

**MULT**iphase: COND: TABLE 4 sets: (0,0), (0.2,0.4), (0.8,0.7), (1,1) ! S vs.  $k_r$

**MULT**iphase CONDUCTivity COMPLimentary SECOnd phase: TABLE of 4 sets:  
(0,0), (0.2,0.4), (0.8,0.7) (1., 1.)

**MULT**i COND vs HEAD: TABLE 4: (100,0) (60,0.4) (40,0.7) (0, 1) !  $\Psi$  vs.  $k_r$

**MULT**i COND vs HEAD: TABLE of 40 sets from file 'SVSKRH'

**MODE 4:** Saturation as an Exponential or Logarithmic Function of Capillary Pressure

**SYNTAX** **MULT** {EXPO | LOGA} [phase] {N, α} [C] [S<sub>r</sub>, S<sub>g</sub>, k<sub>rmin</sub>] [subrgn]

**EXPO** Saturation is an exponential function of  $\Psi$  as shown in below equation

$$\hat{S} = \frac{C}{C + [\exp(\alpha\Psi)]^N - 1} ; \Psi > 0$$

$$\hat{S} = 1 ; \Psi \leq 0$$

**LOGA** Saturation is a logarithmic function of  $\Psi$  given in below equation (\*) as

$$\hat{S} = \frac{C}{C + [\ln(\alpha\Psi)]^N} ; \Psi > 1/\alpha$$

$$\hat{S} = 1 ; \Psi \leq 1/\alpha$$

**phase** See Mode 1 specification.

**N** The exponent N of the equations above. There is no default value.

**α** The constant α of the equations above. There is no default value.

**C** The constant C the equations above. The default value is 0.

**S<sub>r</sub>** An empirical fitting constant S<sub>r</sub> in the definition of the normalized saturation (see Mode 1 specification)

**S<sub>g</sub>** An empirical fitting constant S<sub>g</sub> in the definition of the normalized saturation (see Mode 1 specification)

**k<sub>rmin</sub>** The minimum value of k<sub>r</sub>. The default value is 10<sup>-6</sup>.

**subrgn** See Mode 1 specifciaiton.

## EXAMPLES

**MULT**iphase EXPOnential characteristic: N=2., alpha=0.2

**MULT**iphase LOGAarithmic relation: n = 1.0, alpha = 1, C=200, sr=0.05, sg=0.

**MODE 5:** Capillary Pressure as a Polynomial Function of Saturation

**SYNTAX** MULT {POLY} [phase] {A, B, C, D, E} [S<sub>r</sub>, S<sub>g</sub>, k<sub>rmin</sub>] [subrgn]

**POLY** Ψ is a 4<sup>th</sup> order polynomial function of saturation as given below.

$$\psi = A + B\hat{S} + C\hat{S}^2 + D\hat{S}^3 + E\hat{S}^4$$

Where A, B, C, D and E are empirical constants.

**phase** See Mode 1 specification.

**A** The constant A of above equation. There is no default value for this input.

**B** The constant α of the above equations or the constant B of above equation. There is no default value for this input.

**C** The constant C of the above equations The default value is 0 for the above equation and is 1 otherwise.

**D** The fitting constant, S<sub>r</sub> in the definition of the normalized of saturation (see Mode 1)

**E** The residual saturation, S<sub>g</sub> of the above equation (≥ 0), for the exponential or logarithmic relations, or the constant, E, for the polynomial relation. The default value is 0.

**S<sub>r</sub>** An empirical fitting constant S<sub>r</sub> in the definition of the normalized saturation (see Mode 1 specification)

**S<sub>g</sub>** An empirical fitting constant S<sub>g</sub> in the definition of the normalized saturation (see Mode 1 specification)

**k<sub>rmin</sub>** The minimum value of k<sub>r</sub>. The default value is 10<sup>-6</sup>.

**subrgn** See Mode 1 specifiaiton.

## EXAMPLES

---

**MULT**iphase **POLY**nomial: a = 1., b = -1.

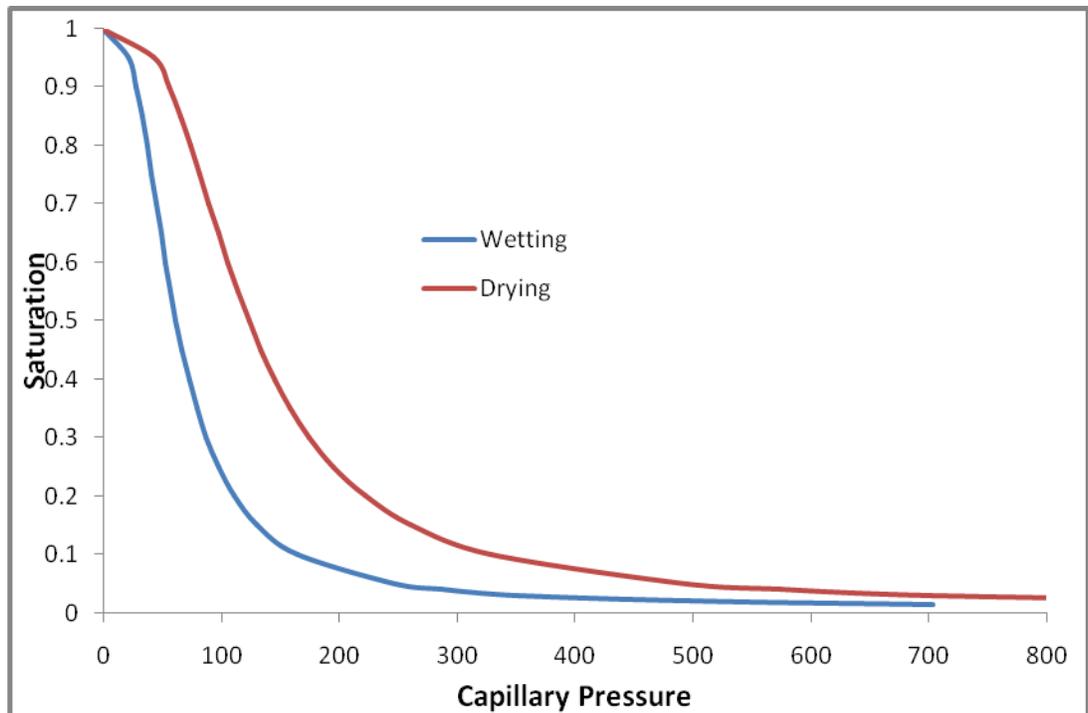
**MULT**iphase **POLY**nomial: a=10, b=-5., c=-2.5, d=-1.5, e=-1, sr=0.1, sg=0.05

**MODE 6:** Tabulated Saturation Characteristic with Hysteresis

**SYNTAX** **MULT** {**TABL**} [**HYST**] {**N<sub>sets</sub>**} {**fname** | **N<sub>1</sub>, ..., N<sub>n</sub>** | **fname**} [**subrgn**]

**TABL** The moisture characteristic is specified in tabular form.

**HYST** The moisture characteristic is specified with hysteresis; that is, with separate wetting and drying cycles. A typical soil characteristic with hysteresis is shown below. The moisture characteristics consist of discrete values of saturation, **S**, and capillary pressure, **Ψ**. See comments below.



**N<sub>sets</sub>** The number of sets of tabulated values which follow. Three values must be specified for each set in the order of (S, wetting Ψ and drying Ψ).

**N<sub>1</sub>** The first value of S (≥0) for the S- Ψ characteristic.

**N<sub>2</sub>** The first value of Ψ for wetting cycle. The input value may be scaled internally if a **SCALE** command is specified immediately preceding the command.

**N<sub>3</sub>** The first value of Ψ for drying cycle.

**N<sub>4</sub>, ..., N<sub>n</sub>** The sets of values of S, Ψ<sub>wet</sub>, Ψ<sub>dry</sub>, in a manner similar to **N<sub>2</sub>**, **N<sub>3</sub>** and **N<sub>4</sub>**. The total number of values specified must of 3 times **N<sub>sets</sub>**.

**fname** The name of the file from which numerical values **N<sub>1</sub>** through **N<sub>n</sub>** for the moisture characteristic curve are read. See Section 3.3 for additional information.

**subrgn** The subregion for which the input is specified. See Section 3.4 for more details. If no subregion is specified, then entire computational domain is selected.

## REMARKS

Capillary pressure as defined in ACRi Software Tools is the negative of the commonly used soil moisture potential that is defined as a negative quantity. The input value for capillary pressure may be scaled internally if a **SCALE** command is specified immediately preceding the command.

The relation between soil saturation and relative conductivity must be separately specified. Care must be taken that if an analytical form (such as van Genuchten) relation is specified, then it must specify the modifier **COND** on the command and that the characteristic curve specified here is not overwritten.

## EXAMPLES

**MULT**iphase flow: TABLE of 4 sets: (0,1.E6), (0.1,1.E3), (0.9,1.E2), (1.,1.)

**MULT**iphase: COND: TABLE 4 sets: (0,0), (0.2,0.4), (0.8,0.7), (1,1) ! S vs.  $k_r$

**MULT**iphase CONDuctivity COMPLimentary SECOnd phase: TABLE of 4 sets:  
(0,0), (0.2,0.4), (0.8,0.7) (1., 1.)

**MULT**i COND vs HEAD: TABLE 4: (100,0) (60,0.4) (40,0.7) (0, 1) !  $\Psi$  vs.  $k_r$

**MULT**i COND vs HEAD: TABLE of 40 sets from file 'SVSKRH'

**MODE 7:** Saturation as a Function of Temperature

**SYNTAX** **MULT** {**WHEE**} {**λ**, **T<sub>d</sub>**} [**S<sub>r</sub>**, **S<sub>g</sub>**] [**subrgn**]

**WHEE** Saturation relation is that of Wheeler (1973) equation as shown below

$$\hat{S} = \frac{1}{\left[1 + (T_F - T)/T_D^\lambda\right]} ; T \leq T_F$$

$$\hat{S} = 0 ; T > T_F$$

In the equation above, T is the fluid temperature; T<sub>F</sub> is the freezing temperature, T<sub>D</sub> is a characteristic delay temperature for freezing and λ is an empirical power-law exponent. The normalized saturation is defined in Mode 1 of this command.

**λ** The exponent, λ, of the above equation (> 0). The default value is 10.

**T<sub>D</sub>** The delay temperature, T<sub>D</sub> (> 0), for the freezing/thawing algorithm. The default value is 1.E-30.

**S<sub>r</sub>** An empirical fitting constant S<sub>r</sub> in the definition of the normalized saturation (see Mode 1 specification)

**S<sub>g</sub>** An empirical fitting constant S<sub>g</sub> in the definition of the normalized saturation (see Mode 1 specification)

**subrgn** See Mode 1 of command

## EXAMPLES

---

**MULT**iphase flow: with **WHEE**ler option: n=4, Td=0.1

**MULT**iphase flow: **WHEE**ler option: n=4, Td=0.1, sr=0.2, sg=0.02

**MODE 8:** Relative Hydraulic Conductivity as a Function of Saturation.

**SYNTAX** **MULT** {**COND**} {**POWE**} {**N, A**} [**B, C**] [**S<sub>r</sub>, S<sub>g</sub>, k<sub>rmin</sub>**] [**COMP**] [**phase**] [**subrgn**]

**COND** The input is for the relative hydraulic conductivity,  $k_r$ .

**POWE**  $k_r$  is a power law of saturation as shown in below equation

$$k_r = A(B + \hat{S})^N + C$$

**N** The exponent N of the power law relation. There is no default value.

**A** The constant A for the power law (default value=1) or the constant, B, for the polynomial relation (default value=0).

**B** The constant B for the power law. The default value is 0.

**C** The constant C for the power law. The default value is 0.

**S<sub>r</sub>** An empirical fitting constant  $S_r$  in the definition of the normalized saturation (see Mode 1 specification)

**S<sub>g</sub>** An empirical fitting constant  $S_g$  in the definition of the normalized saturation (see Mode 1 specification)

**k<sub>rmin</sub>** The minimum value of  $k_r$ . The default value is  $10^{-6}$ .

**COMP** See Mode 1 specification.

**phase** See Mode 1 specification.

**subrgn** See Mode 1 specifiaiton.

## EXAMPLES

---

**MULT**iphasE **COND**uctivity: **POWE**r law: N=4.

**MULT**iphasE **COND**uctivity **POWE**r: N=4, A=1 (B, C, sr, sg) =4\*0. min =1.E-7

**MULT**iphasE **COND** **COMPL**imentary **SECO**nd **POWE**r law: N=4. A=1. sr =0.2

**MODE 9:** Relative Hydraulic Conductivity as a Function of Saturation.

**SYNTAX** **MULT** {**COND**} {**POLY**} {**A**} [**B, C, D, E**] [**S<sub>r</sub>, S<sub>g</sub>, k<sub>rmin</sub>**] [**COMP**] [**phase**] [**subrgn**]

**COND** The input is for the relative hydraulic conductivity,  $k_r$ .

**POLY**  $k_r$  is a 4<sup>th</sup>-order polynomial of saturation as shown in below equation

$$k_r = A + B\hat{S} + C\hat{S}^2 + D\hat{S}^3 + E\hat{S}^4$$

**A** The constant A of the polynomial relation. There is no default value.

**B** The constant B for the polynomial relation. The default value is 0..

**C** The constant C for the polynomial relation. The default value is 0.

**D** The constant D for the polynomial relation. The default value is 0.

**E** The constant E for the polynomial relation. The default value is 0.

**S<sub>r</sub>** An empirical fitting constant  $S_r$  in the definition of the normalized saturation (see Mode 1 specification)

**S<sub>g</sub>** An empirical fitting constant  $S_g$  in the definition of the normalized saturation (see Mode 1 specification)

**k<sub>rmin</sub>** The minimum value of  $k_r$ . The default value is  $10^{-6}$ .

**COMP** See Mode 1 specification.

**phase** See Mode 1 specification.

**subrgn** See Mode 1 specification.

#### EXAMPLES

---

**MULT**iphase **POLY**nomial: a = 1., b = -1., C=0.5, D=-0.5

**MODE 10:** Relative Hydraulic Conductivity as a Function of Capillary Pressure.

**SYNTAX** **MULT** {COND} {EXPO | LOGA | INVE} [COMP] [phase] {N, α} [C] [S<sub>r</sub>, S<sub>g</sub>, k<sub>rmin</sub>] [subrgn]

**COND** The input is for the relative hydraulic conductivity, k<sub>r</sub>.

**EXPO** Relative conductivity is an exponential function of Ψ as defined by equation given below

$$k_r = \frac{C}{C + [\exp(\alpha\psi)]^N - 1} ; \psi > 0$$

$$k_r = 1 ; \psi \leq 0$$

**LOGA** Relative conductivity is a logarithmic function of Ψ as defined by the below equation

$$k_r = \frac{C}{C + [\ln(\alpha\psi)]^N} ; \psi > 1/\alpha$$

$$k_r = 1 ; \psi \leq 1/\alpha$$

**INVE** Relative conductivity is an inverse power law function of Ψ as shown by below equation

$$k_r = \frac{C}{C + [\alpha\psi]^N} ; \psi > 0$$

$$k_r = 1 ; \psi \leq 0$$

**COMP** See Mode 1 specification.

**phase** See Mode 1 specification.

**N** N, is Exponent of the above equations . There is no default value.

**α** α is constant of the above equations . There is no default value.

**C** C is constant, of the above equations . The default value is 1.

**k<sub>rmin</sub>** Minimum permissible value of k<sub>r</sub>. Any computed value smaller than this value is set equal to this value. The default value is 10<sup>-6</sup>.

**subrgn** See Mode 1 of command

## EXAMPLES

---

**MULT**iphase **EXPO**ntial **CON**Ductivity characteristic: N=2., alpha=0.2

**MULT**iphase **LOGA**rithmic **CON**Ductivity: n = 1.0, alpha = 1, C=200.

**MULT**iphase **CON**Ductivity **INVE**rse power law: N=2., alpha = 0.05, C=1.

**MULT**iphase **CON**D **INVE**rse **COM**plimentary **SECO**nd phase: N=2, alpha=0.05

## 5.68 NOZZLE

**PURPOSE** To specify properties and parameters relating to injection of liquid droplets through nozzle. This command is effective only for the **ANSWER®** Software Tool.

**MODE 1:** Location and Properties for Individual Droplets

**SYNTAX** **NOZZ** {**DROP**} [**FUEL|WATE**] {**Q<sub>m</sub>**} [**option**] [[**ABSO**] **RATE=Q<sub>Evap</sub>**] [**STAR=t<sub>Start</sub>**] [**STOP=t<sub>Stop</sub>**] [[**STOR=N<sub>Stor</sub>**] [**TEMP=T<sub>NZL</sub>**] [**FREQ=N<sub>Frq</sub>**] {**[SETS]=N<sub>Sets</sub>**} {**datatype**} [**ELEM**] [**UNIF**] {**fname|N<sub>1</sub>,..., N<sub>n</sub>**}

**DROP** The starting location and properties of each droplet injected by the nozzle are specified. The transport and evaporation of droplets is computed by the built-in algorithm.

**FUEL** Nozzle injects fuel droplets. This is the default option.

**WATE** Nozzle injects water droplets.

**Q<sub>m</sub>** The total mass flow rate [MT<sup>-1</sup>] of fuel injected by the nozzle. This must be the first numerical value on the command. There is no default value.

**option** Droplet options to be activated or deactivated.

Symbol	Description of Corresponding Option
<b>DISA</b>	The specified option(s) listed below is (are) deactivated
<b>EVAP</b>	The droplets evaporate during move. The option is active by default.
<b>DRAG</b>	The droplets undergo drag during move. The option is active by default.
<b>OUTP</b>	A diagnostic message is printed if the injected and evaporated amount from the nozzle differ by more than 0.1%. The option is active by default.

**ABSO** This modifier is effective only if the **RATE** modifier is present, otherwise it is ignored. In the presence of this modifier, the specified rate of evaporation is computed from the original rather than the current remaining mass of the droplet.

**RATE** If this modifier is present, then the number immediately following the modifier is interpreted as rate of evaporation to be forced on the droplets from this nozzle. By default the evaporation rate is computed automatically from the droplet algorithm. This modifier and the corresponding numerical value, if present, may be placed anywhere after **Q<sub>m</sub>** but before the actual droplet information (**N<sub>1</sub>,...,N<sub>n</sub>**). If **SETS** modifier is missing, then it must occur before the **N<sub>Sets</sub>** value.

**Q<sub>Evap</sub>** The rate of evaporation as a fraction of the droplet mass. In the presence of the **ABSOLUTE** modifier, the specified fraction of the original droplet is forced to evaporate at each droplet time step (see **NOZZLE** command with **TIME** modifier for droplet time step). For example, a specification of 0.1 will result in complete evaporation in 10 droplet steps. If the **ABSOLUTE** modifier is not present, then the fraction is applied to the remaining mass of the droplet; that is with a specification of 0.1, the evaporated fractions will be 0.1, 0.09, 0.081, and so on.

**STAR** If this modifier is present, then number immediately following the modifier is interpreted to be the starting time for the discharge of droplets from the nozzle. This modifier and the corresponding numerical value, if present, may be placed anywhere after **Q<sub>m</sub>** but before the actual droplet information (**N<sub>1</sub>,...,N<sub>n</sub>**). If **SETS** modifier is missing, then it must occur before the **N<sub>Sets</sub>** value.

**T<sub>Start</sub>** The starting time for the discharge of droplets from the nozzle if **START** modifier is present. By default for transient flow, the default discharge time is set to zero.

**STOP** If this modifier is present, then number immediately following the modifier is interpreted to be the stopping time for the discharge of droplets from the nozzle. This modifier and the corresponding numerical value, if present, may be placed anywhere after **Q<sub>m</sub>** but before the actual droplet information (**N<sub>1</sub>, ..., N<sub>n</sub>**). If **SETS** modifier is missing, then it must occur before the **N<sub>Sets</sub>** value.

**T<sub>Stop</sub>** The stopping time for the discharge of droplets from the nozzle if **STOP** modifier is present. By default, there is no stopping time for discharge.

- STOR** For transient mode of computations, by default, it is assumed that the droplets that are discharged from the nozzle evaporate relatively fast compared to the computational time step. This implies that for each time step a set of droplets with the same characteristic are discharged; any previous droplets are assumed to have been evaporated or moved out of computational domain. This assumption preserves computational resources in that only one set (generation) of droplets is tracked (determined by **N<sub>Set</sub>** and **N<sub>Frq</sub>**) and stored at each time step. However, if the droplet evaporation time-constant is larger than the computational time step, then some of the droplets released may not fully evaporate during the current time step and need to be kept track of during subsequent time steps.
- The **STOR** modifier allows for storage and tracking of multiple generations of droplets. If this modifier is present then the number immediately following the modifier is interpreted to be the number of generations of droplets that are kept track of. This modifier and the corresponding numerical value, if present, may be placed anywhere after **Q<sub>m</sub>** but before the actual droplet information (**N<sub>1</sub>,...,N<sub>n</sub>**). If **SETS** modifier is missing, then it must occur before the **N<sub>Sets</sub>** value.
- N<sub>Stor</sub>** The maximum number of generations of droplets that are tracked during the computational process. An estimate of the required number can be arrived at by dividing the characteristic evaporation time for largest droplet by the computational time step. A value of 10 is estimated to be adequate for most typical applications. A diagnostic message is printed in the Standard Output File if droplets are removed from storage due to lack of storage space.
- TEMP** If this modifier is present, then number immediately following the modifier is interpreted to be the temperature of the droplets. This modifier and the corresponding numerical value, if present, may be placed anywhere after **Q<sub>m</sub>** but before the actual droplet information (**N<sub>1</sub>, ,N<sub>n</sub>**). If **SETS** modifier is missing, then it must occur before the **N<sub>Sets</sub>** value.
- T<sub>NZL</sub>** The temperature of the droplets discharged from the nozzle if **TEMP** modifier is present. The temperature may also be specified individually for each droplet by using the **T** modifier (see **datatype** below). The temperature must be specified in absolute SI units for the evaporation algorithm to work correctly. The default value is set to 300 degrees.
- FREQ** If this modifier is present, then the number immediately following the modifier is interpreted as the skip frequency (**N<sub>Frq</sub>**) for processing the droplets data. This modifier and the corresponding numerical value, if present, may be placed anywhere after **Q<sub>m</sub>** but before the actual droplet information (**N<sub>1</sub>, ,N<sub>n</sub>**). If **SETS** modifier is missing, then it must occur before the **N<sub>Sets</sub>** value.
- N<sub>Frq</sub>** The skip frequency for processing the droplets. A value of 2 for example, results in the record number, 1, 3, 5, ..., etc. to be read from the file. If no skip frequency is specified, then all droplets are processed.
- SETS** If this modifier is present, then the number immediately following the modifier is interpreted as the number of sets of droplets injected by the nozzle (**N<sub>Sets</sub>**) for processing the droplets to be read from the data. This modifier and the corresponding numerical value, if present, may be placed anywhere after **Q<sub>m</sub>** but before the actual droplet information (**N<sub>1</sub>,...,N<sub>n</sub>**)
- N<sub>Sets</sub>** The number of sets of droplets injected by the nozzle. Each set is assumed to be composed of multiple droplets (see **UNIF** below). There is no default value. If the droplet data is read from a file then this number may not be specified; in this case the file is read to the end of data.

**datatype** Symbols denoting the information contained in each record. The order in which data occurs corresponds to the order of the symbols. The valid symbols are listed below.

Symbol	Description of corresponding datatype
<b>X</b>	The X-coordinate location of the droplet is specified
<b>Y</b>	The Y-coordinate location of the droplet is specified
<b>Z</b>	The Z-coordinate location of the droplet is specified
<b>M</b>	Element number for the location of the droplet is specified
<b>I</b>	Grid index I for the location of the droplet is specified
<b>J</b>	Grid index J for the location of the droplet is specified
<b>K</b>	Grid index K for the location of the droplet is specified
<b>N</b>	The corresponding field is not processed
<b>Skip</b>	The corresponding field is not processed
<b>Dia</b>	The diameter of the droplet, in microns, is specified
<b>Φ</b>	The property denoted by the symbol <b>Φ</b> is specified. Any of the properties listed in Table 2.8.1-3 may be specified.

The location of droplet may be specified by its (x, y, z) coordinates, its grid element number (M), or its (I, J, K) grid indices. If multiple modes of location are present, then the coordinate location takes precedence over other forms, and element location takes precedence over the grid indices. For 2D flows, the coordinate Z and the grid index K, if present, are ignored. **The symbol Φ may specify any combination of droplet properties.** Currently only the droplet diameter, Dia, the velocity components, U, V, W and the temperature, T, are processed. Other input is ignored. For reactive flow, it is assumed that the droplets consist entirely of fuel species in liquid form.

**ELEM** For a structured grid, if the droplet location is specified by its (I, J, K) indices, then by default it is assumed that the I, J, K refer to the grid nodes. If this modifier is present along with the I, J, or K modifiers, then it is assumed that the I, J, K refer to the element numbers rather than the grid nodes (see **GRID Command**). For unstructured grids this input is ignored.

**UNIF** If this modifier is present, then the number of droplets in each of the **N<sub>Set</sub>** sets is assumed to be the same. The number of droplets in the n<sup>th</sup> set, N<sub>n</sub>, is computed as:

$$N_n = \frac{Q_m}{\rho_l \sum_{n=1}^{N_{Set}} \frac{\pi}{6} d_n^3}$$

Here **Q<sub>m</sub>** is the injected mass, **N<sub>Set</sub>** is the number of sets, **ρ<sub>L</sub>** is the liquid density and **d<sub>n</sub>** is the diameter of the n<sup>th</sup> set. If this modifier is not present, then the fuel is equally distributed in the **N<sub>Set</sub>** sets and the number of droplets in the n<sup>th</sup> set is computed as:

$$N_n = \frac{Q_m}{N_{Set} \rho_l \frac{\pi}{6} d_n^3}$$

**fname** The name of the file containing the droplet location and properties. The file must contain **N<sub>Set</sub>** records. Each record must have the same number of values in the same order as the **datatype** symbols on the command.

**N<sub>1,.., N<sub>n</sub></sub>** The data for the droplets if a file name is not specified. **N<sub>Set</sub>** records of data must be present. Each record must have the same number of values in the same order as the **datatype** symbols on the command.

**REMARKS**

The evaporation algorithm works correctly only if SI units are used for all quantities except for the droplet diameter, which must be specified in microns ( $10^{-6}$  m).

**EXAMPLES**

**NOZZle** DROPlets q\_fuel= 0.02, 500 records; data order: N I J K DIA U V W T file 'Droplets.NZL'  
**NOZZle** DROPlets q\_fuel= 0.02, 500 sets. ELEMEnts N I J K DIA U V W T file 'Droplets.NZL'  
**NOZZle** with WATER DROPlets q=0.02, 500 sets. ELEMEnts N I J K DIA U V W T file 'Droplets.NZL'  
**NOZZle** DROPlets data on 'Nozz2D.FIL' q\_fuel= 0.02, 500 records Data order is: DIA J I V U T  
**NOZZle** DROPlets data on 'NOZZLE.FIL' q\_fuel= 0.02, SETS=500, FREQuency = 2  
Data order is: N DIA I J K SKIP U V W T RHO! 1<sup>st</sup> & 6<sup>th</sup> columns skipped; RHO ignored.  
**NOZZle** DROPlets data on 'Nozz2D.FIL' q\_fuel= 0.02, FREQuency =2, SETS=500 records  
Data order is: N DIA I J SKIP SKIP U V N T N! 1<sup>st</sup>, 5<sup>th</sup>, 6<sup>th</sup>, 9<sup>th</sup> & 11<sup>th</sup> columns skipped.  
**NOZZle** DROPlets on 'NOZZ3D.FIL' q = 0.02, 500 sets ELEMEnt based  
Data order is: N DIA K I J SKIP W U V T RHO! 1<sup>st</sup> & 6<sup>th</sup> columns skipped; RHO ignored.  
**NOZZle** DROPlets data from 'NOZZ3D.FIL' q = 0.02, evaporation RATE = 0.1 FREQ=2, SETS=500  
Data order is: N DIA K I J SKIP W U V T RHO! 1<sup>st</sup> & 6<sup>th</sup> columns skipped; RHO ignored.

**MODE 2: Nozzle Location and Injection Parameters (3D Geometry)**

**SYNTAX** **NOZZ** {**LOCA|DEGR**} [**FUEL|WATE**] {**X<sub>o</sub>, Y<sub>o</sub>, Z<sub>o</sub>**} {**Q<sub>m</sub>**} {**Rad, φ**} {**n<sub>x</sub>, n<sub>y</sub>, n<sub>z</sub>**} {**Θ<sub>0</sub>, Θ<sub>F</sub>**} {**d<sub>SMD</sub>**} {**N<sub>Ray</sub>**} [**option**] [[**ABSO**] **RATE=Q<sub>Evap</sub>**] [**STAR=t<sub>Start</sub>**] [**STOP=t<sub>Stop</sub>**] [**STOR=N<sub>Stor</sub>**] [**TEMP=T<sub>NZL</sub>**] {**N<sub>Vel</sub>**} {**V<sub>1,..V<sub>nvel</sub></sub>**}

<b>LOCA</b>	The location of the nozzle discharge flow is specified. Individual droplets are internally generated.
<b>DEGR</b>	By default all angles specified with this mode of command are in radians. If this modifier is present then all angles are in degrees. See Figure below for a schematic of the nozzle.
<b>FUEL</b>	Nozzle injects fuel droplets. This is the default option.
<b>WATE</b>	Nozzle injects water droplets.
<b>X<sub>o</sub>, Y<sub>o</sub>, Z<sub>o</sub></b>	The x, y, and z coordinates of the location of the nozzle. If the geometry is radial, then <b>Z<sub>o</sub></b> must be in radians or degrees. There is no default value.
<b>Q<sub>m</sub></b>	The mass flow rate injected through the nozzle. There is no default value.
<b>Rad</b>	The nozzle radius. There is no default value.
<b>φ</b>	The nozzle cone angle. There is no default value.
<b>n<sub>x</sub>, n<sub>y</sub>, n<sub>z</sub></b>	Direction cosines of nozzle orientation in the X, Y, and Z-directions, respectively). There is no default value.
<b>Θ<sub>0</sub>, Θ<sub>F</sub></b>	Initial and final spray cone angles. There is no default value.
<b>d<sub>SMD</sub></b>	Sauter mean diameter (SMD) in microns. The actual distribution of droplet diameter for each ray (see <b>N<sub>Ray</sub></b> below) is controlled by the <b>NOZZLE</b> command with <b>DISTRIBUTION</b> modifier. There is no default value.
<b>N<sub>Ray</sub></b>	Number of spray cone rays issuing from the nozzle. Each is then subdivided into individual droplets (sub rays) with different diameters. The diameter distribution is controlled by the <b>NOZZLE</b> command with <b>DISTRIBUTION</b> modifier. There is no default value.  The location of each of these droplets is computed so as to cover the total discharge area of the nozzle in a uniform manner. The droplet velocity of each droplet is computed in succession for each sub-ray from the <b>N<sub>Vel</sub></b> values of spray velocity distribution.
<b>option</b>	Same as Mode 1.
<b>ABSO</b>	This modifier is effective only if the <b>RATE</b> modifier is present, otherwise it is ignored. In the presence of this modifier, the specified rate of evaporation is computed from the original rather than the current remaining mass of the droplet.
<b>RATE</b>	If this modifier is present, then the number immediately following the modifier is interpreted as the rate of evaporation to be forced on the droplets from this nozzle. By default the evaporation rate is computed automatically from the droplet algorithm. This modifier and the corresponding numerical value, if present, may be placed anywhere after <b>N<sub>Ray</sub></b> but before the location of <b>N<sub>Vel</sub></b> .
<b>Q<sub>Evap</sub></b>	The rate of evaporation as a fraction of the droplet mass. In the presence of the <b>ABSOLUTE</b> modifier, the specified fraction of the original droplet is forced to evaporate at each droplet time step (see Mode 8 of command for droplet time step). For example, a specification of 0.1 will result in complete evaporation in 10 droplet steps. If the <b>ABSOLUTE</b> modifier is not present, then the fraction is applied to the remaining mass of the droplet; that is with a specification of 0.1, the evaporated fractions will be 0.1, 0.09, 0.081, and so on.
<b>STAR</b>	If this modifier is present, then number immediately following the modifier is interpreted to be the starting time for the discharge of droplets from the nozzle. This modifier and the corresponding numerical value, if present, may be placed anywhere after <b>N<sub>Ray</sub></b> but before the location of <b>N<sub>Vel</sub></b> .
<b>T<sub>Start</sub></b>	The starting time for the discharge of droplets from the nozzle if <b>START</b> modifier is present. By default for transient flow, the default discharge time is set to zero.

- STOP** If this modifier is present, then number immediately following the modifier is interpreted to be the stopping time for the discharge of droplets from the nozzle. This modifier and the corresponding numerical value, if present, may be placed anywhere after **N<sub>Ray</sub>** but before the location of **N<sub>Vel</sub>**.
- T<sub>Stop</sub>** The stopping time for the discharge of droplets from the nozzle if **STOP** modifier is present. By default, there is no stopping time for discharge.
- STOR** For transient mode of computations, by default, it is assumed that the droplets that are discharged from the nozzle evaporate relatively fast compared to the computational time step. This implies that for each time step a set of droplets with the same characteristic are discharged; any previous droplets are assumed to have been evaporated or moved out of computational domain. This assumption preserves computational resources in that only one set (generation) of droplets is tracked (determined by **N<sub>Set</sub>** and **N<sub>Frq</sub>**) and stored at each time step. However, if the droplet evaporation time-constant is larger than the computational time step, then some of the droplets released may not fully evaporate during the current time step and need to be kept track of during subsequent time steps.
- The **STOR** modifier allows for storage and tracking of multiple generations of droplets. If this modifier is present then the number immediately following the modifier is interpreted to be the number of generations of droplets that are kept track of. This modifier and the corresponding numerical value, if present, may be placed anywhere after **N<sub>Ray</sub>** but before the location of **N<sub>Vel</sub>**.
- N<sub>Stor</sub>** The maximum number of generations of droplets that are tracked during the computational process. An estimate of the required number can be arrived at by dividing the characteristic evaporation time for largest droplet by the computational time step. A value of 10 is estimated to be adequate for most applications; a warning is printed if droplets remain unevaporated even at the end of **N<sub>Stor</sub>** computational time steps.
- TEMP** If this modifier is present, then number immediately following the modifier is interpreted to be the temperature of the droplets. This modifier and the corresponding numerical value, if present, may be placed anywhere after **N<sub>Ray</sub>** but before the location of **N<sub>Vel</sub>**.
- T<sub>NZL</sub>** The temperature of the droplets discharged from the nozzle if **TEMP** modifier is present. The temperature must be specified in absolute SI units for the evaporation algorithm to work correctly. The default value is set to 300 degrees.
- N<sub>Vel</sub>** The number of values ( $\geq 1$ ) in the spray velocity distribution. There is no default value.
- V<sub>1,..V<sub>nvel</sub></sub>** Spray velocity distribution; the specified number of values must equal **N<sub>Vel</sub>**. These values are sampled in succession for each sub-ray within each ray. See **REMARKS** for further explanation of how the starting velocity of each droplet is computed.

## REMARKS

---

The individual droplets issued by this command and their characteristics are controlled by the input given on this command and the droplet distribution specified by the **NOZZLE DISTRIBUTION** mode of the command. For example if 7 values for droplet distribution, say, 0.5, 0.8, 1.0, 1.2, 1.5, 1.8 and 2.1, are specified on the **NOZZLE DISTRIBUTION** command and 10 rays (**N<sub>Ray</sub>**) are specified on this command, then each ray will contain 7 sub-rays and a total of 70 droplets will be injected from the nozzle at each step. Further if the droplet SMD (**d<sub>SMD</sub>**) is 40 then each of 10 rays will inject 7 droplets with respective diameters of 20, 32, 40, 48, 60, 72 and 80 microns. The total amount of fuel injected is divided equally between these droplets; in this case each of the droplets represents 1/70<sup>th</sup> of the total injected fuel (**Q<sub>m</sub>**).

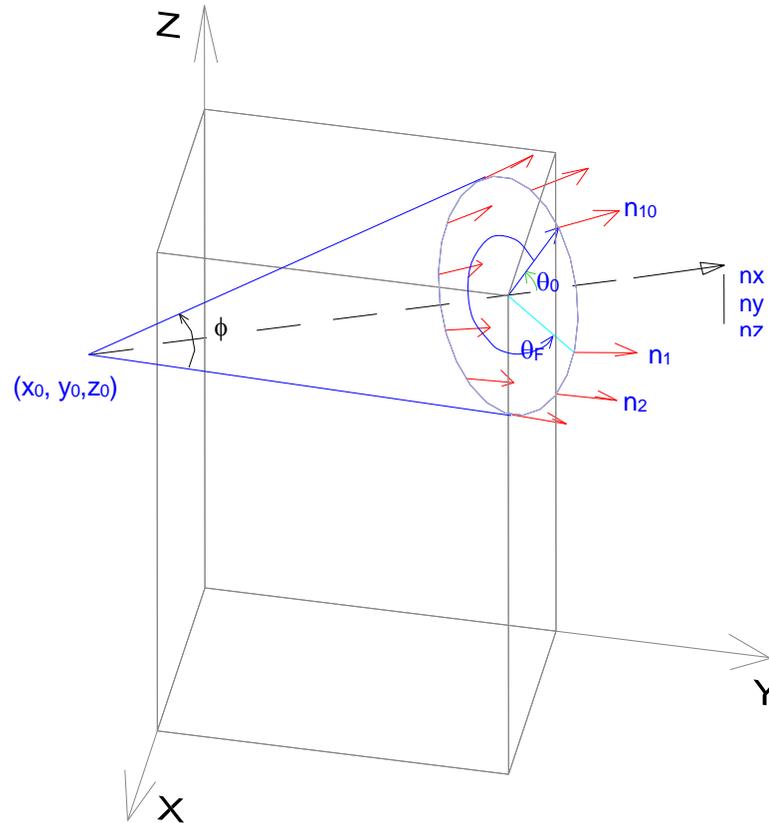
The location of droplets in each ray is computed so as to cover the total range of spray cone angles in a uniform manner. For example if the number of rays is 10 and the initial and final spray cone angles (**Θ<sub>0</sub>**, **Θ<sub>F</sub>**) are 45 and 315, then the 1<sup>st</sup> through the 10<sup>th</sup> rays will emit from locations corresponding to the angles of 45, 75, 105, 135, 165, 195, 225, 255, 285 and 315 degrees, respectively. The actual coordinates, of course, will depend on the location, radius and direction cosines of the nozzle.

The velocity of each droplet that issues is determined by the **N<sub>Vel</sub>** values of velocity distribution (**V<sub>1,..V<sub>nvel</sub></sub>**), the **N<sub>Ray</sub>** values of the number of rays and the **N<sub>Subray</sub>** values on the **NOZZLE DISTRIBUTION** command. The starting velocity of each droplet is computed from the logic given below.

```

IV = 0
do n = 1, NRay
  do ns = 1, Nsubray
    IV = MOD(IV, NVel) + 1
    Vdroplet(n,ns) = Vspecified(IV) !Vspecified ()= V1,..Vnvel
  enddo
enddo

```



**Schematic of the Nozzle and Corresponding Input Parameters**

## EXAMPLES

**NOZZLE** LOCAted at (0.1,1.0,0.02), 0.1 kg/s, 0.005 radius, 0.05 radian cone angle, dir cosines=(0,1,0), (0 6.28), 20 microns, 8 rays, 1 velocity value, 100 m/s

**NOZZLE** with angles in DEGRees, at (0.1,1.0,0.02), 0.1 kg/s, 0.005 radius, 10 deg cone angle, dir cosines=(0,1,0), (0 360) deg, 20 microns, 8 rays, 1 velocity value, 100 m/s

**MODE 3: Nozzle Location and Injection Parameters (2D Geometry)**

**SYNTAX** **NOZZ** {**LOCA**|**DEGR**} [**FUEL**|**WATE**] {**X<sub>o</sub>**, **Y<sub>o</sub>**} {**Q<sub>m</sub>**} {**H<sub>0</sub>**, **φ**} {**n<sub>x</sub>**, **n<sub>y</sub>**} {**d<sub>SMD</sub>**} {**N<sub>Ray</sub>**} [**option**]  
 [[**ABSO**] **RATE=Q<sub>Evap</sub>**] [**STAR=t<sub>Start</sub>**] [**STOP=t<sub>Stop</sub>**] [**STOR=N<sub>Stor</sub>**] [**TEMP=T<sub>NZL</sub>**] {**N<sub>Vel</sub>**}  
 {**N<sub>1,..N<sub>n</sub></sub>**}

<b>LOCA</b>	The location of the nozzle and flow parameters are specified. Individual droplets are internally generated.
<b>DEGR</b>	Same as Mode 2.
<b>FUEL</b>	Nozzle injects fuel droplets. This is the default option.
<b>WATE</b>	Nozzle injects water droplets.
<b>X<sub>o</sub>, Y<sub>o</sub></b>	The x, and y coordinates of the location of the nozzle. There is no default value.
<b>Q<sub>m</sub></b>	The mass flow rate injected through the nozzle. There is no default value.
<b>H<sub>0</sub></b>	The nozzle opening height in cross-section. There is no default value.
<b>φ</b>	The included angle of the spray cone. There is no default value.
<b>n<sub>x</sub>, n<sub>y</sub></b>	Direction cosines of nozzle orientation in the X and Y directions, respectively). There is no default value.
<b>d<sub>SMD</sub></b>	Same as Mode 2.
<b>N<sub>Ray</sub></b>	Same as Mode 2.
<b>option</b>	Same as Mode 2.
<b>ABSO</b>	Same as Mode 2.
<b>RATE</b>	Same as Mode 2.
<b>Q<sub>Evap</sub></b>	Same as Mode 2.
<b>STAR</b>	Same as Mode 2.
<b>T<sub>Start</sub></b>	Same as Mode 2.
<b>STOP</b>	Same as Mode 2.
<b>T<sub>Stop</sub></b>	Same as Mode 2.
<b>STOR</b>	Same as Mode 2.
<b>N<sub>Stor</sub></b>	Same as Mode 2.
<b>TEMP</b>	Same as Mode 2.
<b>T<sub>NZL</sub></b>	Same as Mode 2.
<b>N<sub>Vel</sub></b>	Same as Mode 2.
<b>N<sub>1,.., N<sub>n</sub></sub></b>	Same as Mode 2.

**EXAMPLES**

**NOZZ**le DEGRees: (0.1, 1.0) Q=0.1, R=0.005, cone=45 deg, dir cosines (0.707, 0.707), 40, 5,1,100

**NOZZ**le LOCAted at (0.1, 1.0) Q=0.1, R=0.005, cone=0.75, dc's (0.707, 0.707), 40, 5, TEMP =350, 1,100

**MODE 4:** User Specified Droplet Algorithm with Atomization

**SYNTAX** NOZZ {ATOM} {Q<sub>m</sub>} {N<sub>Sets</sub>}

**ATOM** The droplet computation algorithm is specified by the user and the droplet location and properties will be computed by an atomization algorithm specified by the user. The built-in droplet transport and evaporate algorithm is by-passed. **This option requires source code access or license from ACRi.**

**Q<sub>m</sub>** The total mass flow rate [M T<sup>-1</sup>] of fuel injected by the nozzle. **There is no default value.**

**N<sub>Sets</sub>** The number of individual droplets injected by the nozzle. **There is no default value.**

#### EXAMPLES

---

**NOZZ**le **ATOM**ization algorithm with q\_fuel= 0.02 and 500 droplets

**MODE 5:** Nozzle Sauter Mean Diameter Distribution.

**SYNTAX** NOZZ {DIST} { N1, Nm]

**DIST** Distribution of droplets emanating from the nozzle as a fraction of the given sauter mean diameter (SMD) is specified. This is applicable only to the **NOZZLE** command with **LOCATION** or **DEGREE** modifier.

**N1, Nm** The numerical values to compute the droplet diameters for each subset (ray) of droplets issuing from the nozzle. Each ray of droplets is sub-divided into this many sub-rays for tracking the spray droplets to compute the rate and location of phase change. Each sub-ray has a droplet diameter equal to the product of the SMD ( $d_{SMD}$  specified by **NOZZLE** command with **LOCATION** or **DEGREE** modifier) and the distribution specified here. The number of values (each >0) must lie between 1 and 20. The default values are 0.6, 0.9, 1.2, 1.5 and 2.1.

#### REMARKS

---

This command applies **ONLY** to the Mode 2 or 3 **NOZZLE** commands that follow this command. Any Mode 2 or 3 **NOZZLE** command that precedes this command will use the previously supplied (or default) droplet distribution. Thus each Mode 2 or 3 **NOZZLE** command may have its own droplet distribution. The number of values specified on this command determines the number of “sub-rays” that each spray “ray” is divided into. The number of rays is specified on the corresponding Mode 2 or 3 command.

This command is ignored if droplet information is directly specified by Mode 1.

#### EXAMPLES

---

**NOZZle** SMD DISTribution with values: 0.4, 0.8, 1.2, 1.6, and 2.0

**NOZZle** DISTribution with values: 0.1, 0.4, 0.6, 0.8, 1.0, 1.2, 1.6, 1.8, 2.0, and 2.5

**MODE 6:** Nozzle Spray Computations Parameters

**SYNTAX** NOZZ {FREQ} {V<sub>frq</sub>} [TIME] [N<sub>Begin</sub>, N<sub>End</sub>]

**FREQ** The spray computation frequency parameters are specified.

**V<sub>frq</sub>** The frequency (step or time interval) at which the droplet computations are performed. The default value is 10.

**TIME** By default, V<sub>frq</sub> is the frequency of computations in terms of number of steps. If this modifier is present, then V<sub>frq</sub> is interpreted to be time interval between successive computations.

**N<sub>Begin</sub>** The first computational step at which the droplet computations are performed. The default value is 1.

**N<sub>End</sub>** The last computational step at which droplet computations is performed. The default frequency is set to 9999999.

#### REMARKS

---

Spray droplet computations are automatically triggered if a **NOZZLE** command in Mode 1 through 3 is encountered. This input is required only if the user wants to modify the default values.

This command is equivalent to the **INJECTION FREQUENCY** command and the input on either command will govern both the **NOZZLE** and the **INJECTION** command computations.

#### EXAMPLES

---

**NOZZLE FREQUENCY** of calculations: step = 1 start at step number 10

**NOZZLE FREQUENCY** of calculations: 10 start at step number 10 and stop at step 500

**NOZZLE FREQUENCY TIME** units 2.5 , start at t=0 and stop at t=15.5

**MODE 7:** Nozzle Output Options**SYNTAX** NOZZ {TRAJ} [fname] [NStart, NEnd, NFrq]**TRAJ** A detailed file of trajectory and evolving particulars for each droplet is generated. The frequency of output is controlled by **NStart**, **NEnd** and **NFrq**. A **NOZZLE** command with this modifier must be present to generate the trajectory output.**fname** The name of the file to which the output is directed. See Section 3.3 for additional information. If no name is specified then the output is directed to a file with the same name as the Standard Output file but with the extension removed and replaced by “\_NOZZLE.TMP”. For example, if the Standard Output file is “PROBLEM.OUT”, then the file is named “PROBLEM\_NOZZLE.TMP”. **At any time only one nozzle file can be open.** If a new file name is given, then the previous file is closed and output is directed to the new file**NStart** The first computational step at which the trajectory output is generated. If no value is specified, or the value is less than the nozzle computation frequency (see **NOZZLE FREQUENCY** mode of command), then output is obtained only once at the start of the droplet computations.**NEnd** The last computational step at which the trajectory output is generated. If no value is specified, then the default value is set to 9999999 steps.**NFrq** The frequency of output in terms of computational steps. If both **NEnd** and **NFrq** are missing, then the default value is set to 9999999 so that the output is obtained only once as determined by **NStart**. If on the other hand, **NEnd** is specified but **NFrq** is missing, then the default value is set to unity so that the output is obtained at every step between **NStart** and **NEnd**.**EXAMPLES**

---

**NOZZle** TRAJjectory output the first time droplet computations are performed**NOZZle** TRAJjectory output on file ‘TRAJ.NZL’**NOZZle** TRAJjectory output on file ‘TRAJ.NZL’ at step number 50**NOZZle** TRAJjectory output on file ‘TRAJ.NZL’ start at step# 50 up to step 100 with frequency 20,

**MODE 8: Droplet Time Step Factors and Tracking Options****SYNTAX** **NOZZ** [TIME= $V_{dt}$ ] [STEP] [INCR= $\alpha_{dt}$ ] [ITER= $N_{iter}$ ]**TIME** A suitable time step for moving each droplet is computed by examination of the kinetic and dynamic time scales. This modifier indicates that additional input for time step control is specified by the user. **$V_{dt}$**  If the **STEP** modifier is absent then the time step computed internally is multiplied by  $V_{dt}$ . If the **STEP** modifier is also present, then the droplet computation time step is forced to be equal to  $V_{dt}$ ; any internal computations are ignored.**STEP** The modifier to assist in the interpretation of  $V_{dt}$  input.**INCR** The time step increase factor,  $\alpha_{dt}$ , is specified. If the droplet stays at its current element location after a time step, then the next step is increased by the specified factor. The default value is 1.4142. **$\alpha_{dt}$**  The time step increase factor as defined above.**ITER** If this modifier is present, then the maximum number of iterations for each droplet is set by  $N_{iter}$ . Each droplet is moved continuously in its trajectory unless it:

- ◆ Completely evaporates,
- ◆ Hits a solid boundary or object,
- ◆ Moves out of the computation domain or
- ◆ Completes a specified number of iterations (by default set to 500.).

 **$N_{iter}$**  The maximum number of iterations for droplet. The numerical value must be >0; otherwise the input is ignored and the default or any previously specified value is used.**EXAMPLES****NOZZ**le TIME 2.0!**NOZZ**le TIME STEP is = 1.E-6 seconds!**NOZZ**le TIME multiplier is 2 and INCREASE factor is = 1.**NOZZ**le ITERations=200**NOZZ**le TIME STEP = 1.E-6, ITERations=200**NOZZ**le ITERations=200, TIME STEP = 1.E-6, INCRease factor 1.0Multiply internally computed value by 2  
Force this time step for droplet Movement

<b>MODE 9:</b>	<b>Droplet Temperature and Evaporation Factors</b>
<b>SYNTAX</b>	<b>NOZZ</b> [MINI= $T_{min}$ ] [MAXI= $\delta_{Tmax}$ ] [EVAP= $\delta_{qmax}$ ] [RANZ] [FUEL] [WATE] [GRAV] [STOC] [AUTO]
<b>MINI</b>	If the liquid droplet evaporation rate exceeds the heat input rate, then the liquid temperature can drop. This modifier is used to set the minimum permitted liquid temperature.
<b><math>T_{min}</math></b>	The minimum droplet temperature in absolute units. The default value is 273.15.
<b>MAXI</b>	This modifier sets the maximum change in the temperature of the droplet in a single time step as a fraction of the current liquid temperature.
<b><math>\delta_{Tmax}</math></b>	The maximum change in temperature of the droplet as a fraction of current liquid temperature. The default value is 0.1.
<b>EVAP</b>	This modifier sets the maximum fraction of the mass of the droplet that can evaporate in any single time step.
<b><math>\delta_{qmax}</math></b>	The maximum permitted value of evaporated mass of the droplet in any single step as a function of its current mass. The default value is 0.1.
<b>RANZ</b>	By default the heat and mass transfer rates for droplets are computed from the Abramzon-Sirignano (1989) correlation. If this modifier is present then the more commonly used Ranz-Marshall (1952) correlation is used. If neither <b>FUEL</b> nor <b>WATER</b> modifier is present, then the correlation is used for all nozzles.
<b>FUEL</b>	The Ranz-Marshall correlation is applied to the nozzles with fuel injection.
<b>WATE</b>	The Ranz-Marshall correlation is applied to the nozzles with water injection.
<b>GRAV</b>	By default the effect of gravitational force on the droplets is ignored. If this modifier is present then the droplet velocity is modified by the gravitational force.
<b>STOC</b>	By default the droplets move only due to the deterministic convective movement due to fluid and droplet velocity. If this modifier is present then the droplet movement due to turbulent motion is also accounted for. A random walk algorithm, coupled with the local state of turbulence, is used to generate the stochastic component of particle motion.
<b>AUTO</b>	A fixed evaporation rate for droplets may be explicitly set by the Mode 1 of the command with <b>RATE</b> modifier. This <b>AUTO</b> modifier disables the specified evaporation rate and switches back to the built-in thermal and mass transfer based computation of droplet evaporation rate. One of the uses of this option is to allow the droplets to be forced to evaporate in the early stages of a computation when the flame is not yet established. Once the flame is established, the forced evaporation rate can be disabled.

## EXAMPLES

---

**NOZZ**le **MINI**imum temperature=300 degrees

**NOZZ**le **MINI**imum temperature=300 degrees; **MAXI**imum temperature change=0.01

**NOZZ**le **MINI**imum temperature=300 degrees; Evaporated fraction in one step=0.01

**NOZZ**le **MINI**imum temp=300; **AUTO**matic mode

**NOZZ**les with **GRAV**itational and **STOC**hastic forces and **RANZ** marshall correlation

**NOZZ**les of **WATER** type with **RANZ** Marshall correlation

**NOZZ**le switch back to **AUTO**matic mode

**5.69 OPEN**

**PURPOSE** To specify an open boundary through which fluid may enter or leave based upon prevailing flow conditions. This command is effective only for the ANSWER® Software Tool.

**MODE 1:** Open Boundary with Default Boundary Conditions

**SYNTAX** OPEN {dir} [subrgn] [FREE] [DIAG]

**dir** The orientation index for the open boundary. There is no default value; a value must be specified.

**subrgn** The subregion to be identified as an open boundary. If no subregion is specified, the outermost "dir" oriented boundary of the entire computational domain is selected.

**FREE** By default, diffusion at an OPEN boundary is set to zero. If this modifier is present then no assumption is made about diffusion and the boundary diffusion is computed from the specified boundary conditions and the diffusivity coefficient.

**DIAG** By default only summary statistics of total boundary area, average density and velocity, and flow rate are printed to the standard output device. If this modifier is present then a complete detail of area, velocity components, density and flow rate for each segment of the boundary is printed.

**REMARKS**

---

An open boundary, by definition, is assumed to be a boundary where the values of all dependent variables are fixed if fluid enters through the boundary but the normal gradients of all dependent variables are zero if fluid leaves through the boundary. If the fluid enters through the boundary then the value of the variable for the entering fluid is taken to be that existing at the boundary. This value may be specified (or subsequently changed) by the INITIAL, READ or SET commands and is maintained even if the fluid leaves through the boundary.

**EXAMPLES**

---

OPEN at Y+ boundary at maximum y  
OPEN at X+ boundary for most recently SELEcted subregion  
OPEN at Y- boundary for subregion ID = OPEN  
OPEN OFF at Y- boundary for subregion ID = OPEN

**MODE 2:** Open Boundary with flow based on Pressure and drag Coefficient.

**SYNTAX** OPEN {DISC|GATE} {dir} [subrgn] {C<sub>D</sub>} [P|PRES=P<sub>B</sub>] [AREA=A<sub>B</sub>] [IN|OUT]

**DISC | GATE** The flow at the boundary is computed from:

$$Q_f = C_D \rho \sqrt{\frac{2(P_B - P)}{\rho}} A_B$$

Where  $Q_f$  is the mass flow,  $C_D$  is a discharge or drag coefficient,  $A_B$  is the boundary area,  $P_B$  is the boundary pressure,  $P_f$  is the computed pressure in the flow field interior to the boundary and  $\rho$  is the fluid density at the boundary. If the boundary comprises more than one element face, then the total mass flow is the sum of the individual components.

With  $A$  as the total area of the face, and  $A_j$  as its vector components, the velocity components,  $v_j$ , at the element face are computed from:

$$v_j = \frac{Q_f}{\rho} \frac{|A_j|}{A}$$

**dir** The orientation index for the open boundary. There is no default value; a value must be specified or must be implicit in the **subrgn** definition.

**subrgn** The subregion to be identified as an open boundary. If no subregion is specified, the outermost "**dir**" oriented boundary of the entire computational domain is selected.

**C<sub>D</sub>** The discharge coefficient for the boundary

**P|PRESS** The modifier to indicate that the pressure at the boundary is being specified explicitly.

**P<sub>B</sub>** By default it is assume that the boundary pressure is equal to zero. If the **P** or **PRES** modifier is present, then this is taken to be the value of the boundary pressure.

**AREA** The modifier to indicate that the effective flow at the boundary is being specified explicitly.

**A<sub>B</sub>** By default it is assumed that the area through which the flow occurs is the actual area of the face of the element; that is **A<sub>B</sub>=A**. If the modifier **AREA** is present, then this is taken to be area through which the flow occurs..

**IN** If this modifier is present, then only inflow is allowed at the boundary. Any pressure gradient that indicates an outflow is set to zero.

**OUT** If this modifier is present, then only outflow is allowed at the boundary. Any pressure gradient that indicates an inflow is set to zero.

## EXAMPLES

**OPEN** at ID=OPEN surface: DISCHARGE coef = 0.5

**OPEN** at ID=OPEN surface: DISCHARGE coef = 0.5 , P=100

**OPEN** at ID=OPEN surface: DISCHARGE coef = 0.5 , P=100, AREA=0.02

**MODE 3:** Turn Off a Previously Specified OPEN Port

**SYNTAX** OPEN {dir} {subrgn} {OFF}

**dir** The orientation index for the open boundary. There is no default value; a value must be specified.

**subrgn** The subregion to be identified as an open boundary. If no subregion is specified, the outermost "dir" oriented boundary of the entire computational domain is selected.

**OFF** Previously specified open boundary for the identified subregion is deactivated. A new specification for this subregion may follow.

#### EXAMPLES

---

OPEN at Y+ boundary OFF

## 5.70 OPTION

**PURPOSE** To select or modify built-in default options.

**MODE 1:** **Geometry, Interpolation and Solver Options**

**SYNTAX** **OPTI** **[NEW|OLD]** **[QUAD|VOLU]** **[INTE]** **[ROBU]**

**NEW** The volume fractions and interpolation factors for computing interface values are computed as the vector dot product of the interface areas and the direction vector connecting the two adjoining element. **This is the default option.**

**OLD** **Use of this modifier is not recommended.** The volume fractions and interpolation factors for computing interface values are obtained from the fraction of volume contained between the interface and the element node.

**QUAD** By default the values of a field variable such as pressure at the element interface are computed as a linear function of the distance of the interface from the two nearest element values. If this modifier is present, then the values are computed as inverse square function of the distance of the interface from the two nearest element values. **This modifier is effective only if the OLD modifier is not present.**

**VOLU** The values of a field variable at the element interface are computed as a volume-weighted function of the two nearest element values. **This modifier is effective only if the OLD modifier is not present.**

**INTE** By default the distance of the element nodal point to the element interface is computed as the Euclidian distance from the node to the center of the face. If this option is selected, then the node to interface distance is computed as the projected normal distance of the node to the plane of the interface. For orthogonal grids the two are equal.

**ROBU** By default the solvers related settings are optimized for computational speed rather than numerical stability and robustness. If this option is specified, then the initial settings are optimized for robustness of the solution procedure.

**EXAMPLES**

---

**OPTI**on **NEW** geometry computations with **QUAD**ratic interpolation

**MODE 2: Mach Number Computation Options****SYNTAX** OPTI {MACH} {SONI|GAMM|T|GAS} [V<sub>1</sub>]**MACH** This modifier selects the method used to compute the Mach number for the flow. It must be present for one of these options to be invoked.**SONI** The Mach number is computed from the relation

$$\text{Ma} = \frac{V}{c}$$

Here V and c are the fluid and the sound speeds. The latter can be specified with the **GAS SONIC** command. This is the default option unless **COMPRESSIBLE** or **REACTION** command is specified.

**GAMM** The Mach number is computed from the relation:

$$\text{Ma} = \frac{V}{\sqrt{\gamma C_p \frac{P}{\rho}}}$$

Here V is the fluid speed,  $\gamma$  is the ratio of specific heat at constant pressure to that at constant volume,  $C_p$  is the specific heat at constant pressure, P is the gas pressure and  $\rho$  is the gas density. The  $\gamma$  can be specified by **N1** or the **GAS GAMMA** command.

**T** If this modifier is present **without** the **GAMM** modifier then with, R as the gas constant and T as the temperature in absolute units, the Mach number is computed from:

$$\text{Ma} = \frac{V}{\sqrt{\frac{R}{C_p - R} C_p T}}$$

**T** If this modifier is present **together with** the **GAMM** modifier, then the Mach number is computed from the relation:

$$\text{Ma} = \frac{V}{\sqrt{(\gamma - 1) C_p T}}$$

**GAS** The Mach number is computed from:

$$\text{Ma} = \frac{V}{\sqrt{\frac{C_p}{C_p - R} \frac{P}{\rho}}}$$

Here V is the fluid speed, R is the gas constant,  $C_p$  is the specific heat, P is the pressure and  $\rho$  is the gas density. R is computed from the mass fractions and molecular weights of the gas components. This is the default option if **COMPRESSIBLE** or **REACTION** command is specified.

**V<sub>1</sub>** The sonic speed if the **SONI** modifier is present; the default value is 300. The specific heat ratio  $\gamma$  if the **GAMM** modifier is present; the default value is 1.4.**EXAMPLES**

OPTI on MACH number from GAMMA

**MODE 3:** Pressure Related Options for ANSWER®

**SYNTAX** OPTI [LINE|GRAD|FIXE|FIX] [P|PRES]

**LINE** For ANSWER®, pressure at inlet, outlet and open boundaries is computed from linear extrapolation. This is the default option for hexahedral grids. This is equivalent to assuming that the 2<sup>nd</sup> gradient normal to the boundary is zero. This selection may be numerically unstable if the grid quality is poor or if the pressure gradients near the boundary are very high.

**GRAD** For ANSWER®, pressure at inlet, outlet and open boundaries is computed by assuming that the normal gradient at the boundary is zero. This option is numerically more stable than the LINEar extrapolation. This is the default option for unstructured non-hexahedral grids.

**FIXE** For ANSWER®, pressure at inlet, outlet and open boundaries is assumed to be fixed. This option is more suited for compressible flow computations. In this case the boundary pressure must be explicitly specified by the user.

**FIX** Same as the FIXEd modifier.

**P** If the FIXE option is selected, then by default it is assumed that the pressure at the specified boundary is fixed and the change in pressure ( $p'$ , computed from the continuity equation) is zero. If this modifier is present, then it is assumed that while the boundary P is fixed, the  $p'$  can vary based on the other default settings – such as linear extrapolation at the boundaries. This modifier is effective only in the presence of the FIXE modifier.

**PRES** Same as the P modifier.

#### EXAMPLES

---

**OPTI**on GRAD for pressure ! For extrapolation of pressure for this highly skewed grid

**MODE 4:**      **Enthalpy Related Options for ANSWER®**

**SYNTAX**      **OPTI [ENTH|NORM|GRAD] [DIFF] [ARIT|SOLI|FLUI|OFF]**

**ENTH**      If this modifier is present with the **NORM** modifier, then its interpretation is as described under **NORM**. If it is present with the **GRAD** modifier, then its interpretation is as described under **GRAD**. If this modifier is present by itself, then it activates the computation of thermal energy equation with enthalpy as the dependent variable.

**NORM**      If this modifier is present together with the **ENTH** modifier, then thermal energy equation is solved in terms of normalized enthalpy which is enthalpy divided by reference specific heat (see **REFE**rence command). This has the advantage that the computed value is on the order of the temperature. Since specific heat in metric units can be large (~1000), enthalpy variable can have values on the order of 10<sup>6</sup> or higher which can increase truncation errors with real number arithmetic. **This is the default mode of solving thermal energy equation.**

**GRAD**      The thermal diffusion term is computed as the product of effective thermal conductivity and the temperature gradient. This formulation is the correct expression of the Fourier's Law of Conduction. However, if it is desired to compute thermal diffusion from gradients of enthalpy, then this modifier must be specified together with the **ENTH**aply modifier.

**DIFF**      In **ANSWER® Software Tool**, the thermal conductivity is multiplying by dividing the diffusivity with the specific heat. This modifier along with the modifiers described below selects the manner of computation of the interface thermal conductivity.

**ARIT**      The specific heat at the element interface is computed as the arithmetic average of the values at the two nodes that straddle the interface. **This is the default option. This modifier is effective only in the presence of the DIFF modifier for the ANSWER® Software Tool.**

**SOLI**      The specific heat at the interface between a fluid and solid is taken to be value for solid. If the interface is between two solid (blocked) elements or two fluid elements, then the interface specific heat is computed as the arithmetic average between the two nodes that straddle the element. **This modifier is effective only in the presence of the DIFF modifier for the ANSWER® Software Tool.**

**FLUI**      The specific heat at the interface between a fluid and solid is taken to be value for fluid. If the interface is between two solid (blocked) elements or two fluid elements, then the interface specific heat is computed as the arithmetic average between the two nodes that straddle the element. **This modifier is effective only in the presence of the DIFF modifier for the ANSWER® Software Tool.**

**OFF**      The interface diffusion is set to zero. **This modifier is effective only in the presence of the DIFF modifier for the ANSWER® Software Tool.**

## EXAMPLES

---

**OPT**ion use ENTHalpy      **! Formulation for conjugate heat transfer**

**OPT**ion use TEMPerature formulation for heat transfer even though there are no solid blocks.

**OPT**ion thermal DIFFusion with specific heat of SOLId at the interface

**MODE 5:** Flow Related Options for ANSWER®

**SYNTAX** OPTI [OUTL] [OFF|VELO]

**OUTL** *For all incompressible flow simulations and for steady state compressible flow simulations, the flow at **OUTLET** boundaries is corrected so that the total outflow equals the net global inflow into the domain. For incompressible flow, the convergence of the solution process may be adversely affected if this correction is not performed. For compressible flow, this may or may not be necessary based on the specified boundary conditions for pressure and/or density. This option is available only for ANSWER® Software Tool.*

**OFF** The outflow correction indicated by the **OUTLET** modifier is suppressed. For incompressible flow, convergence of the solution process may be adversely affected and is strongly influenced by the boundary conditions for pressure and/or density. On the other hand, for compressible flow, this modifier may improve the convergence of the solution process. This modifier is effective only in the presence of the **OUTLET** modifier for the ANSWER® Software Tool.

**VELO** By default the corrections relating to the **OUTLET** boundaries are made by directly modifying outgoing convective flux; the outlet velocity is not corrected to correspond to the local values of density and boundary orientation. The outlet velocity is controlled by the specified or default boundary conditions. If this modifier is specified, then the outlet velocity is corrected so that it corresponds to the outflow flux. This modifier is effective only in the presence of the **OUTLET** modifier for the ANSWER® Software Tool. **The use of this modifier is not generally recommended.**

#### EXAMPLES

---

OPTIon OUTLET VELOCITY from convective flux

**MODE 6:** Salt Dissolution Options

**SYNTAX** OPTI {SALT} [FORC|CONS|VELO|OFF] [Z<sub>min</sub>] [Z<sub>max</sub>] [α] [β]

**SALT** This option is applicable only for ANSWER® software. This option affects the dissolution rate of the salt. By default, the dissolution is determined from the correlation given under the **BLOCK SALT** command. That correlation was developed for flows that are dominated by natural convection. In some situations, forced convection may strongly influence the dissolution rate. One such circumstance is when the location of injection is higher than the that of outflow. In this case descending heavy brine near the wall may be assisted by the injected flow traveling towards the lower outflow point.

**FORC** Implements the forced convection enhancement that occurs in salt caverns when the injection point of fresh water is located above the brine outflow point. In this case the velocity of injected fluid is co-directional with the downward movement of heavy brine. **This is the default option.**

With  $Nu_e$ ,  $Nu_{fc}$  and  $Nu_{nc}$ , respectively, the effective, the forced convection and the natural convection Nusselt numbers, the forced convection enhancement is based on the correlation:

$$(Nu_e)^3 = (Nu_{fc})^3 + (Nu_{nc})^3$$

The discussion below will be carried out in terms of Nusselt numbers but the same arguments apply to Mass transfer if Nusselt number is replaced by Sherwood number. For fully developed turbulent flow in tubes, with  $V$  as the characteristic velocity and  $R$  as the radius, the  $Nu_{fc}$  is given by:

$$Nu_{fc} \propto [V R]^{0.8}$$

We may further assume that the characteristic velocity,  $V$ , is related to the injection velocity,  $\beta$ , and the radius,  $R$  as:

$$V \propto \frac{\beta}{R^2}$$

From these equations, we obtain:

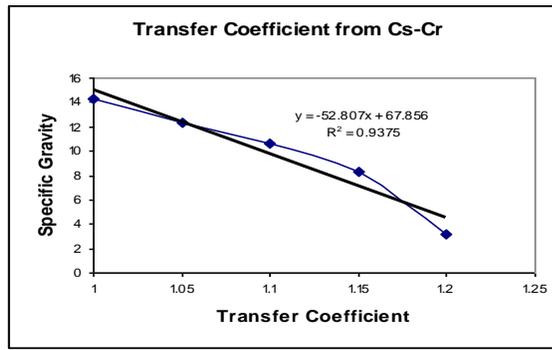
$$\frac{Nu_e}{Nu_{nc}} = \left[ 1 + \left( \frac{C V^{0.8}}{R^{0.8} Nu_{nc}} \right)^3 \right]^{1/3}$$

We write the  $Nu_{nc}$  as:

$$Nu_{nc} = \frac{h_{nc} L}{\Gamma} = \frac{q L}{\Gamma \Delta C_s}; \quad h_{nc} = \frac{q}{\Delta C_s}$$

Where,  $h_{nc}$  is a transfer coefficient,  $L$  is a characteristic length,  $\Gamma$  is a diffusivity,  $q$  is the mass transfer rate and  $\Delta C_s$  is a driving force based on the salt concentration in the fluid. The ratio of mass transfer rate to driving force can be determined from the Sandia correlation for mass transfer rate (see **BLOCK SALT** command). It is graphically represented in the figure below where the transfer coefficient is in metric units. This can be expressed as:

$$Nu_{nc} = \frac{L}{\Gamma} (67.856 - 52.807 C_s)$$



Therefore the forced convection enhancement factor,  $f_{fc}$ , can be written:

$$f_{fc} = \frac{Nu_e}{Nu_{nc}} = \left[ 1 + \left( \frac{\alpha \beta^{0.8}}{R^{0.8} L (67.856 - 52.807C_s)} \right)^3 \right]^{1/3}$$

The constant of proportionality,  $\alpha$ , must be determined from empirical data. In metric units, its value is expected to be on the order of 100.

**CONS**

Forced convection enhancement is based on a user specified constant rather than on a computed function based on Nusselt numbers. The enhancement factor is given as:

$$f_{fc} = \alpha$$

**VELO**

Forced convection enhancement is based on a representative cavern velocity. The enhancement factor is computed from:

$$f_{fc} = 0.98077 - 5.15405V + 74.73588V^2 - 108.98371V^3; \quad V > 0.008467; \quad f_{fc} \leq 4$$

$$= 1; \quad V \leq 0.008467$$

The V is in m/s and it is taken as the velocity just upstream of the outflow location.

**OFF**

Forced convection enhancement is deactivated.

**Z<sub>min</sub>**

The minimum value of the ending value of the axial coordinate of the salt cavern where the forced convection effects are significant.

**Z<sub>max</sub>**

The ending value of the axial coordinate of the salt cavern where the forced convection effects are significant. It should be noted that the L required for the forced convection enhancement option is computed as :  $L = Z_{max} - Z_{min} > 0$ .

**$\alpha$**

The forced convection enhancement constant if the modifier **FORC** or **CONS** is present.

**$\beta$**

The injection velocity if the modifier **FORC** is present. If the injection velocity is not specified, its default value is assumed to be 5.

**EXAMPLES**

**OPT**ion SALT FORCed convection with Zmin=0. Zmax = 200, alpha=100, beta=5

**OPT**ion SALT CONStant enhancement factor for Zmin=0. Zmax = 200, alpha=3

- MODE 7:** Direct Computation without Rhie-Chow Correction for ANSWER®
- SYNTAX** OPTI [DIRE] [P] [IO|IOGR|INLE] [DEBU] [END]
- DIRE** By default ANSWER® Software Tool employs Rhie-Chow correction for computing fluxes at the control-volume interfaces. If this option is invoked, then the flux is computed simply from the weighed average of velocity components.
- P** The DIRE modifier can lead to pressure checker-boarding of pressure field for problems where the boundary conditions are predominantly defined as that of the Neumann type. If this modifier is specified, then a smooth pressure field is computed by direct computation of a pressure equation derived from the computed pressure gradients. The smooth pressure field is printed out with the variable name as PN while the original pressure field is P.
- IO** This modifier is effective only in the presence of the P modifier. By default the smooth pressure field is computed from the basic pressure value only at the interior elements. The boundary values are retained as computed. If this modifier is present then only boundary values at INLET and OUTLET boundaries are retained, the values at the walls and symmetry axis are set to that of zero gradient.
- IOGR** This modifier is effective only in the presence of the P modifier. If this modifier is present then the computed gradient at the INLET and OUTLET boundaries is retained. The values at the walls and symmetry axis are set to that of zero gradient.
- INLET** This modifier is effective only in the presence of the P modifier. If this modifier is present then the computed values at INLET boundaries are retained and the gradient at the outlet boundary is retained. The values at the walls and symmetry axis are set to that of zero gradient.
- DEBU** This modifier is effective only in the presence of the P modifier. If this modifier is present then detailed debug output is produced during the computation process.
- END** This modifier is effective only in the presence of the P modifier. It results in the smooth pressure field being computed only at the end of solution process. For incompressible flow, the Pressure field itself plays no role in the solution procedure; only its gradient appears in the governing equation. Hence computation of pressure field at intermediate stages is wasteful.

## EXAMPLES

---

OPTI on DIRECT P solution method

5.71 OUTLET

**PURPOSE** To specify an outflow boundary for the domain of computation. This command is effective only for the ANSWER® Software Tool.

**MODE 1:** Outflow at an External Boundary or Boundary Next to an Obstacle

**SYNTAX** OUTL {[subrgn], [dir]} [FREE] [FLUX|AREA|MASS|Q<sub>f</sub>] [DIAG]

**subrgn** The subregion to be identified as an outlet. The subregion, together with any orientation specified with the **dir** modifier, must point to a surface or boundary of a region. See Section 3.5 for further details.

For unstructured grids, a subregion defined as a surface (such as with a **LOCATE PAIR** command) must be specified. For structured grids the subregion may consist of a surface or a volume (such as with a **LOCATE COORD** or **LOCATE LIST** command). Volume type of subregions specified with a **LOCATE** command with **FIELD**, **STATION** or **CORRELATION** modifiers cannot be used with this command. For structured grids, if no subregion is specified, then entire computational domain is selected.

**dir** This input is only relevant for structured grids. It specifies the orientation index for the outlet boundary if the subregion specified is not a surface but is of the volume type as described above.

**FREE** By default it is assumed that at an **OUTLET** boundary, no inflow can occur in any part of the boundary. If inflow is detected then it is set to zero. If this modifier is present, then parts of the outlet boundary that have inflow are left uncorrected and the outflow correction is applied only to that part of the boundary that has outflow.

**FLUX** For steady state or incompressible flow, the flow rate at each outlet port is adjusted so that the total outflow equals the total inflow to the computational domain (See Comments below). If this modifier is present, then the outflow is distributed in proportion to the computed outflow flux rate. This is the default option. The default practice is equivalent to treating all the outlets as a single outlet. The outflow through the m<sup>th</sup> port, Q<sub>m</sub> is computed from:

$$Q_m = Q_{out} \frac{\sum_{n=1}^N \sum_{i=1}^3 (\rho u_i A_i)_m}{\sum_{m=1}^M \sum_{n=1}^N \sum_{i=1}^3 (\rho u_i A_i)_m}$$

Here subscript i denotes the i<sup>th</sup> coordinate direction, n the element number, N is the total number of elements in the m<sup>th</sup> outflow port with a total of M outflow ports. The velocity and area vectors, u and A, and the density, ρ, are taken at the element outlet boundary of the element which is defined by the **dir** modifier.

**AREA** The outflow rate at each port is distributed in proportion to the ratio of its area to the total outflow area. This modifier is effective only if multiple **OUTLET** commands are specified. The outflow through the m<sup>th</sup> port, Q<sub>m</sub> is computed from:

$$Q_m = Q_{out} \frac{\sum_{n=1}^N A_{n,m}}{\sum_{m=1}^M \sum_{n=1}^N A_{n,m}}$$

Where A<sub>n,m</sub> is the area of the n<sup>th</sup> element of the m<sup>th</sup> port.

**MASS** The total mass flow rate at the outlet boundary for the m<sup>th</sup> port is set equal to the specified Q<sub>f</sub>; that is: Q<sub>m</sub> = Q<sub>f</sub>.

**Q<sub>f</sub>** If the **MASS** modifier is specified, then this input is treated as mass flow rate at the outlet port. In the absence of the **MASS** modifier, this is the flow fraction that exits through the m<sup>th</sup> port as a fraction of the total outflow through all outlet ports. The outflow rate through the port is computed as:

$$Q_m = Q_{out} \frac{Q_f}{\sum_{m=1}^{m=N} Q_f}$$

**DIAG** By default only summary of total boundary area, average density and velocity, and flow rate are printed to the standard output device. If this modifier is present then a complete detail of area, velocity, density and flow rate for each boundary segment is printed.

## REMARKS

---

An outlet boundary, by definition, is assumed to be a boundary where the normal gradients of all dependent variables are zero. This command provides a compact way to specify the zero normal gradient boundary conditions for all variables at a given boundary.

For incompressible flow, the only other action that is specific to the outflow boundary is the correction of outflow so that the total outflow from all ports equals the total inflow to the computational domain. Some further details of the correction procedure are given in Section 3.14.2 checkup

**Though it is possible to specify, a mix of these options, it is recommended that a consistent practice be followed for specification of outflow if more than one outflow port is specified.**

## EXAMPLES

---

**OUT**Let at X+ boundary at right (maximum x)

**OUT**Let at Y+ boundary for most recently SELEcted subregion

**OUT**Let port at Y+ for SELEcted subregion; outflow fraction = 0.5

**OUT**Let port at Y+ for SELEcted subregion; outflow fraction in proportion to AREA

**OUT**Let at X-: FIXEd pressure boundary for subregion ID = OUTFlow

**OUT**Let at X-: MASS flow = 0.1 at ID = OUTFlow

**MODE 2:** Outflow at an External Boundary or Boundary Next to an Obstacle to Match Inflow at Another Boundary

**SYNTAX** OUTL {[subrgn], [dir]} {[subrgn\_in], [dir\_in]} [DIAG]

**subrgn** See Mode 1 of the command.

**dir** See Mode 1 of the command.

**subrgn\_in** The subregion at which the mass inflow will be computed to set the mass flow at the specified outlet boundary. The subregion, together with any orientation specified with the **dir** modifier, must point to a surface or boundary of a region. See Section 3.5 for further details.

**dir\_in** This input is only relevant for structured grids. It specifies the orientation index for the outlet boundary if the subregion specified is not a surface but is of the volume type.

**DIAG** See Mode 1 of the command.

#### EXAMPLES

---

**OUTL**et at X+ boundary of ID=DOMAIN equal to X- boundary of ID=DOMAIN

**OUTL**et flow at X+ boundary of ID=OUTLET to be equal to X- boundary of ID=INLET

**OUTL**et flow ID=OUTLET\_SURFACE = ID=INLET\_SURFACE !Both sub regions defined as surfaces

**MODE 3:** Outflow from an Internal Element**SYNTAX** OUTL {SOUR} {subrgn} [dir] [FLUX|AREA|VOLUME]**SOUR** The outflow occurs from one or more elements in the interior of the domain. Internally the outflow is treated as a (negative) source.**subrgn** The subregion to be identified as an outlet. The subregion may represent either a volume subregion or a surface. The latter is necessary if a flux or **AREA** modified is specified. See Mode 1 command.**dir** This input is only relevant for structured grids. It specifies the orientation index for the outlet boundary if the subregion specified is not a surface but is of the volume type as described above. **There is no default value; a value must be specified.****FLUX** The outgoing flow is divided according to the relation:

$$Q_m = Q_{out} \frac{\sum_{n=1}^N \sum_{i=1}^3 (\rho u_i A_i)_m}{\sum_{m=1}^M \sum_{n=1}^N \sum_{i=1}^3 (\rho u_i A_i)_m}$$

Here subscript  $i$  denotes the  $i^{\text{th}}$  coordinate direction,  $n$  the element number,  $N$  is the total number of elements in the  $m^{\text{th}}$  outflow port with a total of  $M$  outflow ports. The velocity and density,  $u$  and  $\rho$ , are taken at the element node location and the area vector,  $A$ , is taken at the outlet boundary of the element defined by the **dir** modifier which must be specified for this option. **This is the default option.**

**AREA** The outgoing flux,  $q_m$ , removed from the  $m^{\text{th}}$  element is computed according to the relation:

$$Q_m = Q_{out} \frac{\sum_{n=1}^N A_{n,m}}{\sum_{m=1}^M \sum_{n=1}^N A_{n,m}}$$

Where  $A_{n,m}$  is the area of the  $n^{\text{th}}$  element of the  $m^{\text{th}}$  port taken at the element boundary defined by the **dir** modifier which must be specified for this option.

**VOLUME** The outgoing flow is divided according to the relation:

$$Q_m = Q_{out} \frac{\sum_{n=1}^N V_{n,m}}{\sum_{m=1}^M \sum_{n=1}^N V_{n,m}}$$

Where  $V_{n,m}$  is the volume of the  $n^{\text{th}}$  element of the  $m^{\text{th}}$  port. The **dir** modifier must not be specified for this option.

**REMARKS**

This command is useful for if outflow is to be extracted from interior of the domain so that the total outflow matches the inflow into the computational domain. The outflow specified by Mode and 2 of the command require that the outlet boundary be an exterior boundary or a boundary of an obstacle. This command can be used anywhere in the interior of the computational domain.

**EXAMPLES****OUTL**et SOURCE type at ID=OUTLET divide by VOLUME**OUTL**et SOURCE type at X+ of ID=OUTLET divide by FLUX**OUTL**et SOURCE type at ID=OUTLET\_SURFACE divide by AREA

**MODE 4:** Turn Off a Previously Specified Outflow Port

**SYNTAX** OUTL {[subrgn], [dir]} [OFF]

**subrgn** See Mode 1 of the command.

**dir** See Mode 1 of the command.

**OFF** Previously specified outlet boundary for the identified sub region is deactivated. A new specification for this sub region may follow.

#### EXAMPLES

---

OUTLet OFF at X- boundary for subregion ID = OUTFlow

## 5.72 OUTPUT

**PURPOSE** To select the field arrays to be written to the standard output device and to specify the extent, manner and frequency of output.

**MODE 1:** Output of Phase Space Variables

**SYNTAX** **OUTP** [ $\Phi$ ] [**plane**|**TABL**] [**ADD**] [**STAN**|**fname**] [**subrgn**] [**STAT**|**NOST**|**NOTA**] [**NARR**|**WIDE**] [**TIME**] [**V<sub>irq</sub>**] [**IMME**|**NOW**|**ONLY**|**OFF**]

$\Phi$  **One or more** of the symbols that represent the variables for which output is desired. The valid symbols are listed in Table 2.8.1-3. **A maximum of 64 symbols may be specified on a single command.** The output for listed variables is produced in the order of specification. **If no symbols are specified, and the **ADD** modifier is not present, then a suitable default set is automatically selected after the 1<sup>st</sup> SOLVE command is encountered. The default set includes the variables for which equations are solved, and some important supporting variables based on the nature of the problem.**

**plane** **One** of the character strings: **XY, XZ, YZ., YX, ZX, ZY, RX, RY, RZ and ZR.** The strings **XY, YX, XR** and **RX** all denote the xy plane, **YZ, ZY, RZ** and **ZR** denote the yz plane and, **XZ** and **ZX** denote the xz plane. **By default, for structured grids, the output is generated for xy planes.**

For structured grids the output is presented in a two-dimensional planar tabular form. For 2D geometry only one plane (xy) is possible; hence this modifier is redundant. For 3D geometry, the output can be presented plane-by-plane ordered as xy, yz or xz planes.

**TABL** The output is generated as a table of values with one column per variable. Each record contains the element number followed by the specified variables. A maximum of 16 variables are written in each record. If more than 16 variables are specified then the remaining variables appear in supplemental tables. **This is the default mode for unstructured grids.**

**ADD** **One OUTPUT command is activated by default.** If the **ADD** modifier is present, then the specified command is added to the list of active **OUTPUT** commands. On the other hand, if the **ADD** modifier is not specified, then the default command is modified or replaced by user specified command.

**STAN** The output is directed to the Standard Output Unit. **This is the default option. The default name for this file is the name of the Input Command file but with the extension changed to "OUT".** For example if the input command file name is "MYCASE.INP" or "MYCASE.DAT", then the default output file name will be "MYCASE.OUT".

**fname** The name of the file to which the output is directed. If a file name is specified on an **OUTPUT** command with an **ADD** modifier, then this file is uniquely attached to a unit for output from that command and any subsequent **OUTPUT** command that specifies the **same file name.** **The file name specification is case sensitive and the case must be consistent, otherwise some operating systems may report an error.** See Section 3.3 for additional information.

Output from **all OUTPUT** command without an **ADD** modifier is directed to a **single common file (by default, the Standard Output Unit).** If any such command specifies a file name, then that file becomes the common output file for all commands without the **ADD** modifier. If a new file name or the **STANDARD** modifier is subsequently specified, then the previous file is closed and all subsequent output from **all OUTPUT** commands without an **ADD** modifier is directed to the new file or the Standard Output Unit.

**subrgn** The subregion for which the output is required. For the **plane** type of output, the specified subregion must be in the mode of a grid index window (Mode 1 of **LOCATE** command). **If any other type of subregion is specified, then the TABLE mode (MODE 2) of the command is automatically invoked.** For **TABLE** type of output, any type of subregion can be specified. **If no subregion is specified, the entire computational domain is selected.** See Section 3.4.

**STAT** Statistics of the selected variables are computed and printed at the end of the tabular outputs. The output includes minimum, maximum and mean values, standard deviation, mass weighted averages and other relevant information.

**NOST** Output of statistics of the selected variables is suppressed. **This is the default option.**

<b>NOTA</b>	If this modifier exists together with the <b>STAT</b> modifier, then only the statistics for the variables are printed out; the tabular output is suppressed.
<b>NARR</b>	The output tables are produced in an 80-column (narrow) format.
<b>WIDE</b>	The output tables are produced in a 132-column (wide) format. This is the default mode.
<b>TIME</b>	By default, $V_{frq}$ is the frequency of output in terms of number of steps. If this modifier is present, then $V_{frq}$ is interpreted to be time interval between successive outputs.
$V_{frq}$	The frequency (step or time interval) at which the output is written to the output device. See Section 3.7 for further details. The specified value is ignored if it is zero or negative. The default value is set so that output is obtained only at the end of simulations.
<b>IMME</b>	The output is produced immediately. This modifier is necessary only before the 1 <sup>st</sup> <b>SOLVE</b> command is given. It then enables the output of the current value of specified variable(s) immediately. After the 1 <sup>st</sup> <b>SOLVE</b> command, the output is always produced as soon as the command is encountered and this modifier is redundant. The symbol(s) for the output variable(s) must be specified with this modifier since the default output symbols are not available till after the 1 <sup>st</sup> <b>SOLVE</b> command.
<b>NOW</b>	If this modifier is specified before the 1 <sup>st</sup> <b>SOLVE</b> command, then the output is produced just before the 1 <sup>st</sup> step of the solution but after all the initial and boundary conditions have been processed. Thus, this modifier provides the output of the initial conditions for the variable(s) at the start of the solution process. After the 1 <sup>st</sup> <b>SOLVE</b> command. The output is always produced as soon as the command is encountered; this modifier is redundant.
<b>ONCE</b>	If this modifier is specified then the output is produced only once in response to a <b>NOW</b> or <b>IMMEDIATE</b> modifier or at the frequency determined by $V_{frq}$ .
<b>OFF</b>	Output for any previous command(s) for the same <b>subrgn</b> and type ( <b>plane</b> or <b>TABLE</b> ) is subsequently suppressed.

## REMARKS

---

If no **OUTPUT** command is specified then a command is automatically invoked to produce output for active variables at the end of simulations. The active variables consist of all variables for which the equations are solved plus other variables that are of interest to the problem being solved. Successive commands may be used to accommodate changing output requirements.

Any modifiers or numerical input specified on an **OUTPUT** command without the **ADD** modifier stay active for all commands without the **ADD** modifier unless changed by a subsequent command. For example, any output frequency specified on one such command will stay in effect till it is replaced by new value on a subsequent command.

## EXAMPLES

---

**OUTPut:** U, V, W in NARRow tabular format

**OUTPut:** IMMEdiately the current values of U and V

**OUTPut:** T, V, U and P in WIDE tabular format NOW with variable STATistics

**OUTPut:** U, V, for subregion ID=OUTDomain in WIDE tabular format NOW with STATistics

**OUTPut:** U, V, T, P, K by XZ planes in WIDE format NOW for SELEcted subregion

**OUTPut:** OFF for all variables

**OUTPut:** NOW for subregion specified by the most recent LOCAte command

**OUTPut:** ONLY at step number 50

**OUTPut:** U, V, W as an element-by-element TABLE

**OUTPut:** TABLE of values for U, V and W NOW with STATistics

**OUTPut:** T, V, U and P in WIDE tabular format NOW as TABLE

**OUTPut:** TABLE for U and V for the SELEcted active subregion

**OUTPut** TABLEs for V, W, P and K and L by YZ planes every 15 steps

**OUTPut** TABLE OFF for all variables

<b>MODE 2:</b>	<b>Output of Properties and Coefficients for the Differential Equations</b>
<b>SYNTAX</b>	<b>OUTP {<math>\Phi</math>} {[COND DIFF] [STOR] [SOUR] [MATR] [RESI]} [plane TABL] [ADD] [STAN   fname] [subrgn] [STAT NOST NOTA] [NARR WIDE] [TIME] [<math>V_{frq}</math>] [IMME NOW ONLY OFF]</b>
<b><math>\Phi</math></b>	<b>One, and only one,</b> of the symbols that denotes the dependent variable for which a differential equation is solved. The valid symbols are listed in Table 2.7.1. <b>There is no default value; a valid symbol must be specified.</b>
<b>COND</b>	The conduction or diffusion coefficient for the variable is printed to the output file. If the conduction or diffusion coefficient is a tensor, then the output is produced for each component of the tensor.
<b>DIFF</b>	Same as <b>COND</b> .
<b>STOR</b>	The storage coefficient for the variable is printed to the output file.
<b>SOUR</b>	The net source (algebraic sum of source and sink) for the variable is printed to the output file.
<b>MATR</b>	The matrix coefficients for the variable are printed to the output file. The coefficients are printed in the order of the forcing function (rhs of the matrix), the diagonal coefficient and the influence coefficients for each of the neighboring elements.
<b>RESI</b>	The residue of the matrix equation or the governing differential equation is printed to the output device. If the output is requested at an intermediate stage of computation, then the matrix residue ( $B_j - A_{ij} X_i$ ) is printed. If the output is requested at the final stage of computations, then the residue of the governing differential equation is printed.
<b>plane</b>	See Mode 1 specification.
<b>TABL</b>	See Mode 1 specification.
<b>ADD</b>	<b>This modifier is assumed by default.</b> Each command in this mode, unless an <b>OFF</b> modifier is present, is treated as an additional command that adds to the active commands.
<b>STAN</b>	See Mode 1 specification.
<b>fname</b>	The name of the file to which the output is directed. See Section 3.3 for additional information. Since the <b>ADD modifier is assumed by default</b> , the filename, if specified, is considered unique for the command and any subsequent <b>OUTPUT</b> command that specifies the <b>same file name</b> .
<b>plane</b>	See Mode 1 specification.
<b>subrgn</b>	See Mode 1 specification.
<b>STAT</b>	See Mode 1 specification.
<b>NOST</b>	See Mode 1 specification.
<b>NOTA</b>	See Mode 1 specification.
<b>NARR</b>	See Mode 1 specification.
<b>WIDE</b>	See Mode 1 specification.
<b>TIME</b>	See Mode 1 specification.
<b><math>V_{frq}</math></b>	The frequency (step or time interval) at which the output is written to the output device. <b>In this mode of the command, there is no default value. A value must be specified or the modifiers NOW or IMMEDIATE must be present for an output to be obtained. See REMARKS below.</b>
<b>IMME</b>	The output is produced at the start of computations. In this mode of input, this modifier is equivalent to the <b>NOW</b> modifier because the coefficients and components of the transport equation are not available till the 1 <sup>st</sup> <b>SOLVE</b> command is given and the computation process has started.
<b>NOW</b>	See Mode 1 specification.
<b>ONLY</b>	See Mode 1 specification.
<b>OFF</b>	Output for any previous command(s) for the same <b>subrgn</b> , same type ( <b>plane</b> or <b>TABLE</b> ) and

same combination of **COND**, **DIFF**, **STOR**, **SOUR**, **MATR**, and **RESI** modifiers is subsequently suppressed.

## REMARKS

---

This mode of the **OUTPUT** command provides the ability to monitor the coefficients and contribution of the various components of the transport equation. Any combination of the modifiers **COND**, **DIFF**, **STOR**, **SOUR**, **MATR**, and **RESI** may be specified. **At least one of these must be specified to trigger this mode of the command.**

This command mode triggers output for quantities which are computed (and needed) only during the solution of the equations. Most of these quantities are not allocated permanent storage but are computed on as needed basis. Therefore this output is possible only during the solution of the equation for the corresponding variable. **This command must therefore be given before the last SOLVE command.** Also a frequency (**V<sub>freq</sub>** with or without the **ONLY** modifier) or **NOW** modifier **must be specified** for the output to be

**Multiple commands may be used to obtain output for different variables.** Output is generated after all the coefficients for the solution matrix have been computed. These commands essentially form a queue and are executed for each variable in order as the solution for that variable at the specified step is initiated.

**This mode of the OUTPUT command may be combined with Mode 3 of the command. That is, the modifiers of Mode 2 may be simultaneously specified with those of Mode 3 on the same command.**

## EXAMPLES

---

**OUTPUT:** for U and P of DIFFusion coefficients at the final stage

**OUTPUT:** for T DIFFusion, coefficients NOW and the end to file = 'DIFFUSION.T'

**OUTPUT:** for T STORage and MATRix coefficients NOW (at the next step)

**OUTPUT:** for T CONDuction, STORage, SOURce, MATRix and RESIDUe at step number 52 ONLY

**OUTPUT:** for C: SOURce at step number every 52 steps

**OUTPUT:** in TABLE format of MATRIX for T and P at frequency of 75 steps

**OUTPUT:** of RESIdue for T at the end

<b>MODE 3:</b>	<b>Output of Convective, Diffusive and Total Flux for each Element</b>
<b>SYNTAX</b>	<b>OUTP {<math>\Phi</math>} {FLUX} [CONV] [DIFF] [TOTA] [AREA] [plane TABL] [ADD] [STAN fname] [subrgn] [STAT NOST NOTA] [NARR WIDE] [TIME] [V<sub>frq</sub>] [IMME NOW ONLY OFF]</b>
<b><math>\Phi</math></b>	<b>One, and only one,</b> of the symbols that denotes the dependent variable for which a differential equation is solved. The valid symbols are listed in Table 2.7.1. <b>There is no default value; a valid symbol must be specified.</b>
<b>FLUX</b>	The total flux for each face of the elements of the computational domain is printed. The total flux is the sum of the convective and diffusive components.
<b>CONV</b>	The convective flux for each face of the elements of the computational domain is printed.
<b>DIFF</b>	The diffusive flux for each face of the elements of the computational domain is printed.
<b>TOTA</b>	The sum total of the convective and diffusive fluxes for each face of the elements of the computational domain is printed. <b>This modifier is assumed by default if the CONV or DIFF modifiers are not present.</b>
<b>AREA</b>	The computed flux is divided by the projected area of the face for output.
<b>plane</b>	See Mode 1 specification.
<b>TABL</b>	See Mode 1 specification.
<b>ADD</b>	<b>This modifier is assumed by default.</b> Each command in this mode, unless an <b>OFF</b> modifier is present, is treated as an additional command that adds to the active commands.
<b>STAN</b>	See Mode 1 specification.
<b>fname</b>	See Mode 2 specification.
<b>plane</b>	See Mode 1 specification.
<b>subrgn</b>	See Mode 1 specification.
<b>STAT</b>	See Mode 1 specification.
<b>NOST</b>	See Mode 1 specification.
<b>NOTA</b>	See Mode 1 specification.
<b>NARR</b>	See Mode 1 specification.
<b>WIDE</b>	See Mode 1 specification.
<b>TIME</b>	See Mode 1 specification.
<b>V<sub>frq</sub></b>	The frequency (step or time interval) at which the output is written to the output device. <b>In this mode of the command, there is no default value. A value must be specified or the modifiers NOW or IMMEDIATE must be present for an output to be obtained. See REMARKS below.</b>
<b>IMME</b>	The output is produced at the start of computations. In this mode of input, this modifier is equivalent to the <b>NOW</b> modifier because the coefficients and components of the transport equation are not available till the 1 <sup>st</sup> <b>SOLVE</b> command is given and the computation process has started.
<b>NOW</b>	See Mode 1 specification.
<b>ONLY</b>	See Mode 1 specification.
<b>OFF</b>	Output for any previous command(s) for the same <b>subrgn</b> , same type ( <b>plane</b> or <b>TABLE</b> ) and same combination of <b>FLUX, CONV, DIFF</b> and <b>TOTAL</b> modifiers is subsequently suppressed.

**REMARKS**

---

This mode of the **OUTPUT** command provides the output of the convective, diffusive and total fluxes at the faces of the elements. Any combination of the choice of fluxes may be specified. **At least one of these must be specified to trigger this mode of the command.**

This command mode triggers output for quantities which are computed (and needed) only during the solution of the equations. These quantities are not allocated permanent storage but are computed on as needed basis. Therefore this output is possible only during the solution of the equation for the corresponding variable. **This command must therefore be given before the last SOLVE command.** Also a frequency (**V<sub>freq</sub>** with or without the **ONLY** modifier) or **NOW** modifier **must be specified** for the output to be generated.

**Multiple commands may be used to obtain output for different variables.** Output is generated after all the fluxes required for solution of the transport equation have been computed. These commands essentially form a queue and are executed for each variable in order as the solution for that variable at the specified step is initiated.

**This mode of the OUTPUT command may be combined with Mode 2 of the command. That is, the modifiers of Mode 2 may be simultaneously specified with those of Mode 3 on the same command.** Also, if the output of both the diffusion coefficients and the diffusive fluxes is desired through such a single command, then the modifier **DIFF** must be specified twice or both the **COND** and **DIFF** modifiers must appear simultaneously on such a command.

**EXAMPLES**

---

**OUTPut:** FLUX for T for each element at the final stage

**OUTPut:** CONVective FLUX for T NOW (at the next step) to file 'CFLUX.TMP'

**OUTPut:** CONVective and DIFFusive FLUX for T at step number 50 ONLY

**OUTPut:** CONVective, DIFFusive and TOTAL FLUX divided by AREA for T every 50 steps

**OUTPut:** FLUX divided by AREA for T at step# 52 ONLY in NARRow mode for ID=SUBREGION

**OUTPut:** in TABLE mode FLUX for T in NARRow mode for ID=SUBREGION at end of simulations

**MODE 4:** Output of Special Derived Flow Based Variables

**SYNTAX** **OUTP** **{[GRAD] [VORT] [STRUCTURE] [STRAIN] [STRESS] [LIGHTHILL]}** **[plane|TABL]** **[ADD] [STAN|fname] [subrgn] [STAT|NOST|NOTA] [NARR|WIDE] [TIME] [Vfrq]** **[IMME|NOW|ONLY|OFF]**

**GRAD** The tensor components of the gradients of velocity,  $\phi_{ij}$  are output, where:

$$\phi_{ij} = \frac{\partial u_i}{\partial x_j}$$

Here  $u_i$  is the  $i^{\text{th}}$  component of velocity and  $x_j$  is the  $j^{\text{th}}$  coordinate. The gradient consists of 4 components for 2D and 9 for 3D.

**VORT** Output of the components of vorticity vector,  $\omega$ , is obtained, where:

$$\omega_1 = \frac{\partial u_3}{\partial x_2} - \frac{\partial u_2}{\partial x_3}; \quad \omega_2 = \frac{\partial u_1}{\partial x_3} - \frac{\partial u_3}{\partial x_1}; \quad \omega_3 = \frac{\partial u_2}{\partial x_1} - \frac{\partial u_1}{\partial x_2}$$

There are 3 components of vorticity for 3D flow but only one for 2 D flow.

**STRUCTURE** Output of a “vorticity” structure variable,  $\Omega$ , is obtained. It is defined as:

$$\begin{aligned} \Omega &= -\frac{\partial u_1}{\partial x_2} \frac{\partial u_2}{\partial x_1} \quad \text{for 2D} \\ &= -\frac{\partial u_1}{\partial x_2} \frac{\partial u_2}{\partial x_1} - \frac{\partial u_2}{\partial x_3} \frac{\partial u_3}{\partial x_2} - \frac{\partial u_3}{\partial x_1} \frac{\partial u_1}{\partial x_3} \quad \text{for 3D} \end{aligned}$$

**STRAIN** Output of the components of strain tensor,  $S_{ij}$ , is obtained, where:

$$S_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right)$$

There are 3 components for 2D and 6 for 3D flow.

**STRESS** Output of the components of stress tensor,  $\tau_{ij}$ , is obtained, where:

$$\tau_{ij} = \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \delta_{ij} \sum_k \frac{\partial u_k}{\partial x_k} \right) - \delta_{ij} p$$

Here  $\mu$  is viscosity and  $p$  is the pressure. There are 3 components for 2D and 6 for 3D flow.

**LIGHTHILL** Output of the components of “lighthill” stress tensor,  $L_{ij}$ , is obtained, where:

$$L_{ij} = -\mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \delta_{ij} \sum_k \frac{\partial u_k}{\partial x_k} \right) + \rho u_i u_j$$

Here  $\mu$  is the viscosity and  $\rho$  is the density. There are 3 components for 2D and 6 for 3D flow.

**plane** See Mode 1 specification.

**TABL** See Mode 1 specification.

**ADD** This modifier is assumed by default. Each command in this mode, unless an **OFF** modifier is present, is treated as an additional command that adds to the active commands.

**STAN** See Mode 1 specification.

**fname** See Mode 2 specification.

**plane** See Mode 1 specification.

**subrgn** See Mode 1 specification.

**STAT** See Mode 1 specification.

**NOST** See Mode 1 specification.

**NOTA** See Mode 1 specification.

<b>NARR</b>	See Mode 1 specification.
<b>WIDE</b>	See Mode 1 specification.
<b>TIME</b>	See Mode 1 specification.
<b>V<sub>freq</sub></b>	See Mode 1 specification.
<b>IMME</b>	See Mode 1 specification.
<b>NOW</b>	See Mode 1 specification.
<b>ONLY</b>	See Mode 1 specification.
<b>OFF</b>	Output for any previous command(s) for the same <b>subrgn</b> , same type ( <b>plane</b> or <b>TABLE</b> ) and same combination of <b>GRAD, VORT, STRUCTURE, STRAIN, STRESS</b> and <b>LIGHTHILL</b> modifiers is subsequently suppressed.

## EXAMPLES

---

**OUTPut:** VORTICITY and GRADients of velocity at the final stage to file 'VORTGRAD.FIL'

**OUTPut:** VORTICITY STRUCTure, STRESS and LIGHTHILL stress at frequency of 100 steps

**OUTPut:** in TABLE mode VORTICITY STRUCTure, STRESS and LIGHTHILL every 100 steps

**MODE 5:** Output of Partitioning for Species Variables

**SYNTAX**    **OUTP** {SECO} {PHAS} {Φ} [plane|TABL] [ADD] [STAN|fname] [subrgn]  
 [STAT|NOST|NOTA] [NARR|WIDE] [TIME] [V<sub>frq</sub>] [IMME|NOW|ONLY|OFF]

**SECO**        The output for the inventory (amount in storage at a element) of a species in phases other than the primary phase is generated. The output is computed as:

$$Q_{\Phi S} = Q_{\Phi} - Q_f \Phi$$

Here  $Q_{\Phi S}$  is the inventory of property in an element in phases other than the primary phase,  $Q_{\Phi}$  is the total inventory and  $\Phi$  is the species concentration in the primary phase for the element.

**PHAS**        A companion modifier that must be present to invoke this mode of the command.

**Φ**             **One or more** of the symbols that represent the variables for which output is desired. The valid symbols are the species symbols listed in Table 2.7.1. *A maximum of 32 symbols may be specified on a single command.* This output can only be obtained for species variables for which a transport equation is solved. *There is no default value; a valid symbol must be specified.*

**plane**        See Mode 1 specification.

**TABL**        See Mode 1 specification.

**ADD**         *This modifier is assumed by default.* Each command in this mode, unless an **OFF** modifier is present, is treated as an additional command that adds to the active commands.

**STAN**        See Mode 1 specification.

**fname**       See Mode 2 specification.

**plane**        See Mode 1 specification.

**subrgn**      See Mode 1 specification.

**STAT**        See Mode 1 specification.

**NOST**        See Mode 1 specification.

**NOTA**        See Mode 1 specification.

**NARR**        See Mode 1 specification.

**WIDE**        See Mode 1 specification.

**TIME**        See Mode 1 specification.

**V<sub>frq</sub>**        See Mode 1 specification.

**IMME**        See Mode 1 specification.

**NOW**         See Mode 1 specification.

**ONLY**        See Mode 1 specification.

**OFF**         Output for any previous command(s) for the same **subrgn**, same type (**plane** or **TABLE**) and same combination variable is subsequently suppressed.

**REMARKS**

This mode of the command is active only for the **PORFLOW®** Software Tool where some of the property may be present in secondary phases (solid and gas components of the porous matrix). The partitioning between different phases is specified by the **TRANSPORT**, **DISTRIBUTION** or **RETARDATION** commands.

The output from this mode of the command can be generated **only after** the start of the solution process. At the initial stage, before the solution process starts, the meaningful values of the inventory of variables may not be available. **This mode of the command operates independently of any other modes of the command and essentially adds to any other commands that are specified.**

**EXAMPLES**

**OUTP**ut: SECOndary PHASe partitioning for variables C and C2.to file 'SOLIDC.FIL'

**OUTP**ut: SECO PHAS for C and C2.to file 'SOLIDC.FIL' every 100 steps

**5.73 OXIDIZER**

**PURPOSE** To specify the proportion of inert substance to active substance in the oxidizer for reactive flows. This command is effective only for the ANSWER® Software Tool.

**SYNTAX** **OXID** {**N1**}

**N1** The molar ratio of inert to active substance in the oxidizer. The default value is 3.76.

**REMARKS**

---

The default oxidizer is "air" which consists of oxygen and nitrogen in a molar ratio of 3.76 moles of N<sub>2</sub> per mole of O<sub>2</sub>. Any other proportion of diluent nitrogen, including zero, may be set by this command.

**EXAMPLES**

---

**OXID**izer is pure oxygen

**OXID**izer with N<sub>2</sub>:O<sub>2</sub> = 2.5 by molar ratio

**5.74 PAUSE**

**PURPOSE** To cause a temporary halt in the calculations.

**SYNTAX** **PAUS**

**REMARKS** \_\_\_\_\_

Operator intervention is required to restart the calculation process. This command allows for operator action during interactive execution.

**EXAMPLES** \_\_\_\_\_

**PAUS**e and await operator action

5.75 PERIODIC

**PURPOSE** To select the periodic boundary option for boundary conditions of the domain.

**MODE 1:** Periodic Boundary Without Velocity Transformation

**SYNTAX** PERI {X|Y|Z}fname} [option]

**X|Y|Z** The boundary conditions for the corresponding x, y or z direction boundaries are determined automatically from the requirement of periodicity. A **BOUNDARY** command with a corresponding boundary index (X-, X+ or Y-, Y+, or Z-, Z+) must not be specified if this option is selected.

**fname** The name of the file which contains the information about the periodic boundary locations. This file must contain a list with one record for each pair of matching periodic surfaces (called “lower” and “upper”). Each record must consist of only 4 integers in the order:

1. Element number for the periodic surface at “lower” periodic boundary.
2. Surface number for the periodic surface (or face) of the element above.
3. Element number containing the periodic surface at “upper” periodic boundary.
4. Surface number for the periodic surface (or face) of the element above.

The definition of the “lower” or “upper” boundary is arbitrary. The element numbers that appear in the periodic list must consist only of the internal field elements. Boundary node numbers must not appear in this list. The surfaces are numbered from 1 to 4 for Quad and from 1 to 6 for Hex elements. For hybrid unstructured grids, the number of faces depends on the type of element. See **LOCATE** command with **PAIR** modifier and Section 3.4 for further details of the manner in which these surface numbers are assigned.

**option** Options selected for implementation of the **PERIODIC** command.

option	INTERPRETATION
<b>LINE</b>	For a structured grid, by default, an ADI cyclic solver is used in the periodic direction. If this modifier is present, then a linear form of the ADI solver is used. This modifier has no effect if the solver used does not have a cyclic option. Of the default solvers only the ADI has the cyclic option. Please check with ACRi for other solvers.
<b>CYCL</b>	For an unstructured grid, by default, a linear solver is used in the periodic direction. The periodicity is imposed explicitly by forcing the boundary values to be periodic. If this modifier is present, then the cyclic solver is used. The user must ensure that the unstructured grid is setup in a manner that the ADI “lines” in the periodic direction run unbroken from one boundary to the other. Otherwise the solution may be corrupted with unforeseen consequences
$\Phi$	One or more ( $\leq 54$ ) symbols that denote the variables, which are periodic (or not periodic if <b>OFF</b> modifier is specified) at the $\theta_1$ and $\theta_2$ boundaries. By default all variables are assumed to be periodic.
<b>OFF</b>	The periodic boundary conditions are not applied to the variables specified by the $\Phi$ modifier. All variable not so identified are assumed to be periodic.
<b>PRES</b>	By default it is assumed that the pressure at a periodic surface is also cyclic; that is there is no net pressure loss in the direction of periodicity. If this modifier is present, then any initial pressure difference between the matched elements of the periodic surfaces is maintained.

**EXAMPLES**

---

**PERI**odic boundaries in X

**PERI**odic boundaries in X, Y and Z

**PERI**odic boundary in X for U, V, W, T, P only

**PERI**odic boundary at X option OFF for P and FU variable

**PERI**odic boundary surface data on file: 'PERIODIC.DAT'

**PERI**odic boundary in Z direction with LINEAr solver

**PERI**odic boundary unstructured grid but use CYCLic solver; data on file: 'PERIODIC.DAT'

**MODE 2: Periodic Boundary With Radial to Cartesian Velocity Transformation****SYNTAX** PERI {X|Y|Z}fname {THET} [option] [ $\Psi$ , N<sub>VR</sub>, N<sub>VC</sub>]

**X|Y|Z** The boundary conditions for the corresponding x, y or z direction boundaries are determined automatically from the requirement of periodicity. A **BOUNDARY** command with a corresponding boundary index (X-, X+ or Y-, Y+, or Z-, Z+) must not be specified if this option is selected.

**fname** See Mode 1 specification.

**THET** This modifier denotes that the velocity components are Cartesian but the problem is periodic in the circumferential ( $\theta$ ) direction of a cylindrical coordinate system. At times a problem which is periodic in cylindrical (x, r,  $\theta$ ) coordinates is solved instead in a Cartesian (x, y, z) system. Then, the Cartesian (U, V, W) velocity components at the two periodic boundaries located at  $\theta=\theta_1$  and  $\theta=\theta_2$  obey the transformed periodicity relations:

$$U_2 = U_1$$

$$V_2 = V_1 \cos \Psi - W_1 \sin \Psi$$

$$W_2 = W_1 \cos \Psi + V_1 \sin \Psi$$

where

$$\Psi = \theta_2 - \theta_1$$

**option** See Mode 1 specification.

$\Psi$  The angle of periodicity,  $\Psi$ , in degrees. There is no default value, if the modifier **THET** is present, then a value must be specified.

**N<sub>VR</sub>** A number ( $1 \geq N_{VR} \geq 3$ ) that denotes the order of the radially directed component of velocity among the three velocity components (U, V, W). The default value is 2.

**N<sub>VC</sub>** A number ( $1 \geq N_{VC} \geq 3$ ) that denotes the order of the circumferentially directed component of velocity among the three velocity components (U, V, W). The default value is 3.

**EXAMPLES**

**PERI**odic boundaries in Z with THETA = 45 degrees

**PERI**odic boundaries in Y with THETA = 45 degrees v = 1, w = 2

**PERI**odic boundary in Z with THETA = 18 degrees for U, V, W, T, P only

**PERI**odic boundary in Z with THETA = 18 degrees but option OFF for P and FU variable

**PERI**odic boundary at THETA = 20 degrees surface data on file: 'PERIODIC.DAT'

**MODE 3:** Periodic Boundary With General Velocity Transformation

**SYNTAX** PERI {X|Y|Z}fname {TRAN|STAC} [option] [V<sub>1</sub>, ..., V<sub>9</sub>]

**X|Y|Z** The boundary conditions for the corresponding x, y or z direction boundaries are determined automatically from the requirement of periodicity. A **BOUNDARY** command with a corresponding boundary index (X-, X+ or Y-, Y+, or Z-, Z+) must not be specified if this option is selected.

**fname** See Mode 1 specification.

**TRAN** The velocity vector at the identified periodic boundaries is transformed according to:

$$\begin{bmatrix} U2 \\ V2 \\ W2 \end{bmatrix} = \begin{bmatrix} n1 & n2 & n3 \\ n4 & n5 & n6 \\ n7 & n8 & n9 \end{bmatrix} \cdot \begin{bmatrix} U1 \\ V1 \\ W1 \end{bmatrix}$$

**STAC** The velocity vector at the identified periodic boundaries is transformed according to the transformation specified in the most recent **STACK TRANSFORMATION** command.

**option** See Mode 1 specification.

**V<sub>1</sub>,..V<sub>9</sub>** The transformation matrix to compute the velocity components at the “higher” (say, at Z+) periodic boundary from those at the “lower” (say, at Z-) periodic boundary if the **TRAN** modifier is present. This information is ignored otherwise. There is no default value, if the modifier **TRAN** is present, then a set of 9 values must be specified.

## EXAMPLES

**PERI**odic boundaries in Z Velocity **TRAN**sformed by:

```
1      0      0
0      0     -1
0      1      0
```

**PERI**odic boundaries in Z Velocity **TRAN**sformed by:

```
1.0      0.      0.00.0      0.995004      -0.09983340.0      0.0998334
0.995004
```

**PERI**odic boundaries in Z Velocity by **STAC**k **TRAN**sformation

**PERI**odic boundary in X by **STAC**k **TRAN**sformation for U, V, W, T, P only

**PERI**odic boundary **TRAN**sformation as follows with surface data on file: 'PERIODIC.DAT'

```
1.0      0.      0.00.0      0.995004      -0.09983340.0      0.0998334
0.995004
```

**PERI**odic boundary by **STAC**k **TRAN**sformation surface data on file: 'PERIODIC.DAT'

## 5.76 PERMEABILITY

**PURPOSE** To specify permeability for a porous medium.

**MODE 1:** Constant Permeability

**SYNTAX** PERM {K} [subrgn]

**K** The numerical value (>0) for the permeability of the medium. The units are [L<sup>2</sup>]. The SI units are [m<sup>2</sup>]. There is no default value; a value must be specified.

**subrgn** The subregion for which the input is specified. See Section 3.4. If no subregion is specified, the entire computational domain is selected.

**APPLICABILITY**

---

This command is available only for the ANSWER® and PORFLOW® Software Tools.

**EXAMPLES**

---

PERMEABILITY for the medium is 1.E-6 square meter. For the entire domain  
PERMEABILITY for the medium is 1.E-6 square meter for ID=POROUS subdomain.

**MODE 2:** Permeability as a General Function

**SYNTAX** PERM {func[ξ]} {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub>|fname} [subrgn]

**func** One of the modifiers listed in Table 4.2.1 that denotes the functional form of the value of the permeability for the porous medium.

**ξ** One of the independent variables listed in Table 4.2.2. If no variable is specified, then the independent variable is assumed to be time.

**N<sub>1</sub>,...,N<sub>n</sub>** The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.

**fname** The name of the file containing the numerical values N<sub>2</sub> through N<sub>n</sub>. This option is available only for selected functions. See Section 3.3 for additional information.

**subrgn** The subregion for which the input is specified. See Section 3.4. If no subregion is specified, the entire computational domain is selected.

#### APPLICABILITY

---

This command is available only for the ANSWER® and PORFLOW® Software Tools. For ANSWER® Software Tool this command automatically triggers the inclusion of Brinkman-Forchheimer terms (BRINK command).

#### EXAMPLES

---

Generic examples for this command are given in Section 4.4. The command keyword (PERM) must replace the keyword used in these examples.

**MODE 3:** Permeability from Carman-Kozeny Equation

**SYNTAX** PERM {**CARM**} {**dp**} [**Cck**] [**subrgn**]

**CARM** The permeability is computed from the Carman-Kozeny relationship::

$$K = \frac{d_p^2 \theta^3}{C_{CK} (1-\theta)^2}$$

Where

**K** is the permeability of the medium [L<sup>2</sup>],

**dp** is the representative particle diameter[L],

**θ** is the porosity of medium,[0]

**CCK** is an empirical constant [0]

**dp** The representative particle diameter for the porous medium. Typically it represents the ratio of the volume to the surface area of the particles. There is no default value; a value must be given.

**Cck** The empirical constant for the relationship. The default value is 180.

**subrgn** The subregion for which the input is specified. See Section 3.4. If no subregion is specified, the entire computational domain is selected.

## EXAMPLES

**PERMEABILITY** from **CARMAN** relationship diameter =**0.002** meters for the entire domain

**PERMEABILITY** from **CARMAN** relationship diameter =**0.002** m for **ID=POROUS** subdomain.

**PERMEABILITY** from **CARMAN** relationship diameter =**0.002** m with Cck= **150**.

**5.77 PHASE**

**PURPOSE** To specify phase change option for multi-phase option. This command is effective for the **ANSWER®** and **PORFLOW®** Software Tools.

**MODE 1:** Phase change at a Fixed Temperature.

**SYNTAX** PHAS {EVAP | FREE} [T<sub>phase</sub>]

**EVAP** Evaporation temperature is specified. This is the default mode.

**FREE** Freezing temperature is specified.

**T<sub>phase</sub>** The temperature at which phase change occurs. If **FREE** modifier is specified; then it is the freezing temperature with a default value of 0 °C otherwise it is the evaporation temperature with a default value of 100 °C.

**EXAMPLES**

---

**PHASE** change at **100** Degrees Celcius

**PHASE** change is **FREEZING** at **0** Degrees Celcius

**MODE 1:** Phase Change (Evaporation) from Casius-Clapeyron Equation.

**SYNTAX** PHAS {CLAP} [T<sub>0</sub>, T<sub>1</sub>, P<sub>0</sub>, A]

**CLAY** Phase change occurs according to Clayperon’s vapor-pressure relation:

$$T_s = T_0 + \frac{T_1}{A - \ln(P/P_0)} ,$$

where, T<sub>s</sub> is the saturation temperature at pressure P, T<sub>0</sub> is a base temperature, P<sub>0</sub> is a base pressure and A is an empirical constant. This relation can also be inverted to compute, saturation pressure, P<sub>s</sub>, at any given temperature, T, according to:

$$P_s = P_0 \text{Exp} \left( A - \frac{T_1}{T - T_0} \right) ,$$

**T<sub>0</sub>** The base temperature T<sub>0</sub> of the vapor pressure equation above. The default value is 45 K.

**T<sub>1</sub>** The temperature T<sub>1</sub> of the vapor pressure equation. The default value is 3841.1954 K.

**P<sub>0</sub>** The pressure P<sub>0</sub> of the vapor pressure equation. The default value is 131.57894 pascal.

**A** The constant A of the vapor pressure equation. The default value is 18.3443.

**COMMNETS**

This relation is suitable for evaporation of a liquid. The default constants are tuned for water so that the computed vapor pressure is in good agreement with the experimental data in the 0°C to 350°C range as shown in the table below

**TABLE: VAPOR PRESSURE FOR WATER**

T		P <sub>s</sub> (pascals x 10 <sup>5</sup> )	
°C	K	Equation	Steam Tables
0	273.15	0.00594	0.00611
20	293.15	0.02309	0.02337
50	323.15	0.12260	0.12335
100	393.15	1.00539	1.0133
150	425.15	4.72619	4.7597
170	443.15	7.87248	7.9203
200	473.15	15.47723	15.5506
250	523.15	39.54853	39.776
300	573.15	84.61033	85.92
350	623.15	158.7037	165.37

**EXAMPLES**

**PHASE** Change according to default **CLAPEYRON's** relation

**PHASE** Change **CLAPEYRON's** equation: T0=45, T1=3841.1954, p0=131.57894, a1=18.3443

## 5.78 POROSITY

**PURPOSE** To define the material porosity of the porous matrix.

**MODE 1:** **Constant Porosity**

**SYNTAX** **PORO** { $\Theta_E$ } [ $\Theta_T$ ,  $\Theta_D$ ] [**ALWA**] [**subrgn**]

$\Theta_E$  The effective (or flow) porosity,  $\Theta_E$ . There is no default value; a value must be specified.

$\Theta_T$  The total porosity,  $\Theta_T$ . If no value is specified, total porosity is set equal to the effective porosity.

$\Theta_D$  The diffusional porosity,  $\Theta_D$ . If no value is specified, diffusional porosity is set equal to the effective porosity.

**ALWA** By default the **PORO** command is implemented **immediately and only once** – as soon as the command is encountered. If this modifier is present then the command is executed immediately as well as **repeatedly** at the beginning of each time step (or iterative step in steady state mode) of the solution procedure.

**subrgn** The subregion for which the input is specified. See Section 3.4. If no subregion is specified, the entire computational domain is selected.

**APPLICABILITY**

This command is currently available only for the **ANSWER®** and **PORFLOW®** Software Tools.

**REMARKS**

By default  $\Theta_E$ ,  $\Theta_T$  and  $\Theta_D$  are all set to unity for the computational domain. The user input through this command overwrites the initial default values.

For **ANSWER®** Software Tool only the effective porosity is relevant and hence the  $\Theta_T$  and  $\Theta_D$ , if specified, are ignored. Also for **ANSWER®** this command automatically triggers the inclusion of Brinkman-Forchheimer terms (**BRINK** command).

**EXAMPLES**

**POROSITY** for the material is **0.35**

**POROSITY** porosities: effective = **0.1**, total **0.2**, diffusive **0.15**

**POROSITY** porosities: **3\*0.15**

**POROSITY** porosities are **0.45, 1.50, 1.25** internal check

**MODE 2:** Porosity as a General Function**SYNTAX** **PORO** [**ONLY**] [**EFFE**] [**TOTA**] [**DIFF**] {func[ $\xi$ ]} { $N_1, N_2, \dots, N_n$ |fname} [**ALWA**] [**subrgn**]**ONLY** By default, the effective, total and diffusional porosity are all set to the specified value. If this modifier is present, then only the components specified by one or more of the **EFFE**, **TOTA**, or **DIFF** modifiers are set to the specified value.**EFFE** If **ONLY** modifier is also present, then the effective porosity is set to the specified value.**TOTA** If **ONLY** modifier is also present, then the total porosity is set to the specified value.**DIFF** If **ONLY** modifier is also present, then the diffusional porosity is set to the specified value.**func** One of the modifiers listed in Table 4.2.1 that denotes the functional form of the value of the porosity of the porous medium. $\xi$  One of the independent variables listed in Table 4.2.2. If no variable is specified, then the independent variable is assumed to be time. $N_1, \dots, N_n$  The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.**fname** The name of the file containing the numerical values  $N_2$  through  $N_n$ . This option is available only for selected functions. See Section 3.3 for additional information.**ALWA** By default the **PORO** command is implemented immediately and only once – as soon as the command is encountered. If this modifier is present then the command is executed immediately as well as **repeatedly** at the beginning of each time step (or iterative step in steady state mode) of the solution procedure.**subrgn** The subregion for which the input is specified. See Section 3.4. If no subregion is specified, the entire computational domain is selected.**APPLICABILITY** \_\_\_\_\_This command is currently available only for the **ANSWER®** and **PORFLOW®** Software Tools.**REMARKS** \_\_\_\_\_For **ANSWER®** Software Tool only the effective porosity is relevant and hence the mode selecting modifiers are ignored. Also for **ANSWER®** this command automatically triggers the inclusion of Brinkman-Forchheimer terms (**BRINK** command).More than one mode of porosity may be set by a single command. In the presence of the **ONLY** modifier, any combination of the modifiers **EFFE**, **TOTA** or **DIFF** may be present on the command. If the specified function is a constant then Mode 1 of the command is activated.**EXAMPLES** \_\_\_\_\_Generic examples for this command are given in Section 4.4. The command keyword (**PORO**) must replace the keyword used in these examples.

## 5.79 PRANDTL

**PURPOSE** To specify Prandtl number for the fluid. This command is effective only for the ANSWER® Software Tool.

**SYNTAX** PRAN [EFFE] { $\sigma$ }

**EFFE** If the modifier EFFECTIVE is present on the command line then the input is assumed to be for the effective Prandtl number,  $\sigma_e$ , otherwise it is assumed to be the molecular Prandtl number,  $\sigma$ , for the fluid.

$\sigma$  The effective or molecular Prandtl number (>0) for the fluid. The default value is 0.7.

**EXAMPLES**

---

PRANDtl number for fluid = 1.

PRANDtl number EFFEffective value = 0.5

## 5.80 PRECIPITATION

**PURPOSE** To specify the precipitation or uptake reactions for dissolved species. This command is effective only for the **PORFLOW®** Software Tool.

**MODE 1:** Precipitation Reaction with Constant or Variable Reaction Rate and Solubility.

**SYNTAX** **PREC** {idreac} {Φ} {Ψ} {ω, ξ} [LIMI, {ζ}] [TOLE, {ε}] [POSI|NEGA] [subrgn]

**idreac** A unique identifier for the precipitation (or uptake) reaction rate where the species Φ precipitates to generate species Ψ or the species Ψ is converted to species Φ by uptake. The reaction rate is computed as:

$$R_{\Phi} = \omega (\xi - \Phi) \quad (1)$$

The computed rate is algebraically added to the equation for Ψ and subtracted from the equation for Φ. The reaction rate units are the units of the species divided by time.

- Φ** The symbol for the species that is precipitated or re-adsorbed by the reaction.
- Ψ** The symbol for the species that is produced by precipitation or dissolved by uptake.
- ω** The constant or the symbol for a variable that specifies the precipitation/uptake reaction time constant (units of frequency) There is no default value; either constant or a symbol must be specified.
- ξ** The constant or the symbol that specifies the solubility limit for the species (units of Φ). There is no default value; either constant or a symbol must be specified.
- C<sub>s</sub>** The solubility limit for the species (units of Φ). There is no default value; a value must be specified.
- LIMI** If this modifier is present, then the reaction rate is assumed to be influenced by the presence of a limiter species denoted by the symbol, ζ. If the limiter is present above a certain threshold, the reaction rate is assumed to be zero.
- ζ** The symbol for the reaction limiter variable. If no tolerance (see **TOLE**) is specified then no reaction occurs if the value of the limiter variable exceeds 10<sup>-30</sup>. There is no default value; a symbol must be specified
- TOLE** If this modifier is present, then it must be followed by the threshold value above which no precipitation reaction occurs.
- ε** The threshold value of the limiter. There is no default value; a value must be specified
- POSI** The reaction rate is set to zero if Φ > C<sub>s</sub>. This limits the reaction only to uptake.
- NEGA** The reaction rate is set to zero if C<sub>s</sub> > Φ. This limits the reaction only to precipitation.
- subrgn** The subregion for which the **idreac** reaction is computed. If no subregion is specified, the reaction is computed for the entire computational domain. See Section 3.4.

**EXAMPLES**

**PRE**Cipitation reaction R1 from C1 to C2 with frequency = 1000; Solubility = 0.02

**PRE**Cipitation reaction R2 from C2 to C3 with time constant = 0.05; Solubility = 0.30

**PRE**Cipitation reaction R1 from C1 to C2 with solubility defined by C3 and a frequency = 1000

**PRE**Cipitation reaction R2 from C2 to C3 with C4 as solubility and time constant = 0.05

**MODE 2: Precipitation Reaction with Variable Solubility****SYNTAX** **PREC** {**ELEM**} {**idreac**} {**Φ**} {**Ψ**} {**ω**} {**ξ**} {**C<sub>Sat</sub>**} [**subrgn**]

**ELEM** Element mode of the precipitation reaction is activated. In this mode, the solubility of the isotope is decreased by the presence of other elements according to a modified Raoult's Law. With **C<sub>Sat</sub>** as the solubility of the pure element, the solubility, **C<sub>S</sub>**, in the presence of other elements is given by:

$$C_S = C_{Sat} \frac{(M_i^{liquid} + M_i^{solid})}{\sum_{j=1}^n (M_j^{liquid} + M_j^{solid})}$$

where  $M_i^{liquid}$  and  $M_i^{solid}$  are, respectively, the molar concentration of elements in the liquid and solid phases.

**idreac** A unique identifier for the precipitation (or uptake) reaction rate where the species **Φ** precipitates to generate species **Ψ** or the species **Ψ** is converted to species **Φ** by uptake. The reaction rate is computed as:

$$\text{Rate} = \omega (C_S - \Phi) .$$

The computed rate is algebraically added to the equation for **Ψ** and subtracted from the equation for **Φ**. The reaction rate units are the units of the species divided by time.

**Φ** The symbol for the species that is precipitated or re-adsorbed by the reaction.

**Ψ** The symbol for the species that is produced by precipitation or dissolved by uptake.

**ω** The constant or the symbol for a variable that specifies the precipitation/uptake reaction time constant (units of frequency) **There is no default value; either constant or a symbol must be specified.**

**ξ** The symbol for the variable that represents the sum of the molar concentrations of all the species that participate in reducing the solubility of the element; **the denominator in the above equation for C<sub>S</sub>.**

**C<sub>Sat</sub>** The solubility limit for the species (units of **Φ**) in its pure element form. **There is no default value; a value must be specified..**

**subrgn** The subregion for which the idreac reaction is computed. **If no subregion is specified, the reaction is computed for the entire computational domain.** See Section 3.4.

**REMARKS**

For this mode of input, it is preferable that all species concentrations be expressed in terms of moles. In this case the **SET SUM** command can be used to define the sum of molar concentrations, **ξ**. If the species concentrations are in terms of mass units, then **SET LINEAR** command must be used to define **ξ** with appropriate coefficients.

**EXAMPLES**

**PREC**ipitation **ELEM**ENT form: R1 for C1 and C2 with molar variable C3 frequency = 1000, csat=0.05

**PREC**ipitation **ELEM**ENT form: R1 for C1, C2 molar variable C3 omega=1000, csat=0.05 ID=REAC1

**PREC**ipitation **ELEM**ENT form: R1 for C1, C2, sum=C3, omega=1000, csat=0.05 **SELEC**Ted region

## 5.81 PRINT

**PURPOSE** To generate output of details of mass flow rate and statistical measures of flow variables.

**MODE 1:** Print Statistics for Flow and Selected Variable

**SYNTAX** PRIN [FLOW] [STAT] [Φ] [GEOM] [subrgn]

**FLOW** The mass inflow into and, outflow from, all inlets, outlets, open boundaries and sources are computed and written to the standard output device as soon as the command is encountered.

**STAT** The statistics related to the minimum, maximum and average value of specified variable(s) is computed and written to the standard output device as soon as the command is encountered. The variables for which the statistics are computed must be identified by the Φ modifier(s)

**Φ** One or more symbols that denote the dependent variables for which the **STAT** modifier is effective. The valid symbols are listed in Table 2.8.1-3. There is no default value.

**GEOM** The face areas and volume of the subregion identified on the command are printed on the output file.

**subrgn** The subregion for which the output of face areas and volume is required. If no subregion is specified, the entire computational domain is selected. See Section 3.4.

**EXAMPLES**

---

**PRINT** FLOW immediately

**PRINT** STATistics for U, V, T now

**PRINT** FLOW and STATistics for U, V, T now

**MODE 2: Integrated Averages for Selected Variables at Uniformly Spaced Locations**

**SYNTAX**    **PRIN** { $\Phi$ } {**AVER**|**PROF**} {**N1**} {coordinate} [**subrgn**] [**dir**] [**TOLE**= $V_{tol}$ ] [**NORM**] [**BASE**= $\Phi_{base}$ ] [**MASS**|**AREA**|**VOLU**] [**OUTP**|**fname**] [**V<sub>irq</sub>**] [**TIME**]

**$\Phi$**     One or more of the symbols that represent the variables for which output is desired. The valid symbols are listed in Tables 2.8.1-3 and 2.8.4. There is no default value.

**AVER**    The averages of the specified variable are computed at a number of specified locations.

**PROF**    Same as **AVER** modifier.

**N1**    The number of locations at which the averages are computed. It must be the first numerical value on the command. There is no default value; a numerical value (>0) must be specified.

**coordinate**    One of: **X**, **Y**, **Z**, **R** or **THETA** modifiers. It denotes the coordinate of locations at which the averages are computed. The **Z**, **R** or **THETA** can only be used for three-dimensional geometry. In this case is **R** = sqrt ( $y^2 + z^2$ ) and **THETA** = arctan ( $z/y$ ).

**subrgn**    The subregion for computations. If no subregion is specified, the entire domain is selected.

**dir**    The orientation index for the boundary of the sub-domain if averages are to be computed only at the boundary. See Section 3.5 for available choices. There is no default value.

**TOLE**    The coordinate tolerance for inclusion of the elements in averaging at a location. With  $S_i$  as the **coordinate** of the  $i$ th location, all elements that satisfy:  $S_i - V_{tol} \leq S < S_i + V_{tol}$  are included in averaging. By default, the tolerance is set equal to half the interval between successive uniformly spaced coordinates from 0 to 1. For example if **N1**=10, then interval will be 0.1 and  $V_{tol}$  will be set of 0.05. However if the modifier **TOLE** is present, then the tolerance is set to the user specified value  $V_{tol}$ .

**$V_{tol}$**     The user specified value of tolerance. This input must be specified only if the **TOLE** modifier is present and then it must immediately follow the modifier.

**STAT**    By default, if the **dir** modifier is present, then the computed values are the arithmetic, area-weighted and mass-flux-weighted mean values; otherwise the computed values are arithmetic, volume-weighted and mass-weighted mean values. If this modifier is present, then computed values include the arithmetic mean, minimum, maximum and standard deviation of the variable.

**NORM**    The output for the selected variable is normalized or non-dimensionalized as:

$$\Phi_{output} = \frac{\Phi_{computed} - \Phi_{base}}{\Phi_{norm} - \Phi_{base}}$$

where  $\Phi_{base}$  and  $\Phi_{norm}$  are normalizing values. By default these are set to the minimum and maximum values for the sub-domain selected by **subrgn** and **dir** modifiers. However other options are available as described below.

**BASE**    This modifier will automatically select the normalized (**NORM**) form of the output. If this modifier exists, then  $\Phi_{base}$  must be specified and, in this case, only one variable ( $\Phi$ ) must be specified on the command.

**$\Phi_{base}$**     The user specified value for  $\Phi_{base}$ . This input must be specified only if the **BASE** modifier is present and then it must immediately follow the modifier.

**MASS**    This modifier is significant only if the **BASE** modifier is specified. By default  $\Phi_{norm}$  is taken to be arithmetic mean of the values for the sub-domain. If this modifier is present, then  $\Phi_{norm}$  is the mass or mass-flux weighted mean of the variable.

**AREA**    This modifier is significant only if the **BASE** modifier is specified. By default  $\Phi_{norm}$  is the arithmetic mean of the values for the sub-domain. If this modifier is present, then  $\Phi_{norm}$  is the area-weighted mean in the presence of the **dir** modifier or volume-weighted mean otherwise.

**VOLU**    Same as the **AREA** modifier.

**OUTP**    The output is directed to the standard output file unit.

**fname**    The file name for output. If a file name is present then the output is directed to the named file,

otherwise the output is directed to the standard output device.

**V<sub>frq</sub>** The frequency (step or time interval) at which the output is written to the output device. See Section 3.7 for further details. **If V<sub>frq</sub> is specified, it must be the last numerical input on the command.** The default value is set so that output is obtained only at the end of simulations.

**TIME** By default, **V<sub>frq</sub>** is the frequency of output in terms of number of steps. If this modifier is present, then **V<sub>frq</sub>** is interpreted to be time interval between successive outputs.

## REMARKS

This command generates a series of profiles of integrated averages for a variable. For example, if the selected **coordinate** is **X**, the **subrgn** denotes a 3D sub-domain and there is no **dir** modifier, then this command will generate integrated averages across the yz planes of the sub-domain at **N1** uniformly spaced locations in the x-direction. If the **subrgn** and **dir** denote a xy plane, and the coordinate selected is **X**, then it generates integrated averages across the y-direction at **N1** locations along the x-coordinate of the plane.

This command with the **BASE** modifier can be used to generate the pattern and profile factors, which are commonly used in the aircraft and gas turbine industry to denote the variation of temperature at the outlet plane of the combustor. In this case, the variable  $\Phi$  should be the symbol **T** (for Temperature) and the base value,  $\Phi_{base}$ , should be specified as the average inlet temperature (commonly called T3). The  $\Phi_{norm}$  is automatically selected to be the computed arithmetic mean (generally called T4). If T4 is to be mass-weighted mean then the modifier **MASS** must also be present.

## EXAMPLES

**PRINT** Profile of T at 20 X locations! 20 values versus x for whole of the domain

**PRINT** PROFILE of U, V, T at 20 X locations! Multiple variables

**PRINT** AVERAge of U at 20 X locations for ID=OUTLET in X+ direction! Output for outlet plane.

**PRINT** PROFILE of STATistics for T at 20 X locations for ID=OUTLET in X+ direction! Min, max & stats

**PRINT** PROFILE of NORMalized T at 20 X locations for ID=OUTLET in X+ direction

**PRINT** AVERAge for T at 20 R for ID=OUTLET. In X+ direction BASE value = 375

**PRINT** AVERAge for T at 20 R for ID=SECTION1 BASE value = 375 MASS weighted

**PRINT** AVERAge for T at 20 Y locations with TOLErance=0.01 for SELEcted subregion in X+ direction

**PRINT** AVER for T at 20 Y locs with TOLErance=0.01 and BASE=375 for ID=OUTLET in X+ every 50 steps

**PRINT** AVER T 20 Y locs TOLErance=0.01, BASE=375, ID=OUTLET X+ at TIME interval of 0.05 units

**MODE 3: Integrated Averages for Selected Variables at Specified Locations**

**SYNTAX**    **PRIN** { $\Phi$ } {**AVER|PROF**} {**N<sub>Loc</sub>**} { $x_1, \dots, x_n$ } {**coordinate**} [**subrgn**] [**dir**] [**TOLE=V<sub>tol</sub>**] [**NORM**] [**BASE=Φ<sub>base</sub>**] [**MASS|AREA|VOLUME**] [**OUTP|fname**] [**V<sub>frq</sub>**] [**TIME**]

$\Phi$	One or more of the symbols that represent the variables for which output is desired. The valid symbols are listed in Tables 2.8.1-3 and 2.8.4. There is no default value.
<b>AVER</b>	The averages of the specified variable are computed at a number of specified locations.
<b>PROF</b>	Same as <b>AVER</b> modifier.
<b>N<sub>Loc</sub></b>	The number of locations at which the averages are computed. It must be the first numerical value on the command. There is no default value; a numerical value (>0) must be specified.
$x_1, \dots, x_n$	The coordinates of the locations at which averages are computed. There must be a total of <b>N<sub>Loc</sub></b> values and these must precede any other numerical input except <b>N<sub>Loc</sub></b> . The values must be specified in relative units. The relative units vary from 0 to 1, respectively, from the minimum to the maximum of the <b>coordinate</b> for the <b>subrgn</b> . There is no default value.
<b>coordinate</b>	Same as Mode 2 of the command.
<b>subrgn</b>	Same as Mode 2 of the command.
<b>dir</b>	Same as Mode 2 of the command.
<b>TOLE</b>	The coordinate tolerance, <b>V<sub>tol</sub></b> , for inclusion of the elements in averaging at a location. With $S_i$ as the <b>coordinate</b> of the $i^{\text{th}}$ location, all elements that satisfy: $S_i - V_{tol} \leq S < S_i + V_{tol}$ are included in averaging. By default, the tolerance is set equal to 0.01.
<b>V<sub>tol</sub></b>	Same as Mode 2 of the command.
<b>STAT</b>	Same as Mode 2 of the command.
<b>NORM</b>	Same as Mode 2 of the command.
<b>BASE</b>	Same as Mode 2 of the command.
$\Phi_{base}$	Same as Mode 2 of the command.
<b>MASS</b>	Same as Mode 2 of the command.
<b>AREA</b>	Same as Mode 2 of the command.
<b>VOLUME</b>	Same as the <b>AREA</b> modifier.
<b>fname</b>	Same as Mode 2 of the command.
<b>V<sub>frq</sub></b>	Same as Mode 2 of the command.
<b>TIME</b>	Same as Mode 2 of the command.

**EXAMPLES**

**PRINT** PROFILE of U, V, T at 5 X locations 0.1, 0.3, 0.4, 0.5, 0.9

**PRINT** PROFILE of STATistics for T at 4 Y values 0.1, 0.3, 0.5, 0.8 for ID=OUTLET in Y+ direction

**PRINT** PROFILE of NORMalized T at 4 X stations 0.1, 0.3, 0.5, 0.8 for ID=OUTLET in X+ direction

**PRINT** AVERAGE for T at 4 R values 0.1, 0.3, 0.5, 0.8 for ID=OUTLET in X+ direction BASE value = 375

**PRINT** AVERAGE for T at 4 R values 0.1, 0.3, 0.5, 0.8 ID=SECTION1 BASE value = 375 MASS weighted

**PRINT** AVERAGE for T at 3 R values 0.2, 0.4, 0.8 TOLERance=0.01 SELEcted region in X+ direction

**PRINT** AVER for T at 4 R 0.1, 0.3, 0.5, 0.8 TOLERance=0.01 BASE=375 ID=OUTLET in X+ frq= 50 steps

**PRINT** AVER T 3 Y 0.2, 0.4, 0.8 TOLERance=0.01, BASE=375, ID=OUTLET X+ at TIME frq 0.05 units

**MODE 4: Forces and Moments for a Selected Variable at a Specified Location****SYNTAX** **PRIN** [ $\Phi$ ] {**FORC**|**MOME**} [**subrgn**] [**dir**] [ $X_i$ ] [ $\Phi_0$ ] [ $V_{freq}$ ] [**TIME**] [**NOW**] [**OFF**] [**fname**] $\Phi$  The symbol for the variable for which the “force” or “moment” is to be computed. The valid symbols are listed in Tables 2.8.1-3. If no symbol is specified then pressure, P, is selected.**FORC** “Force” and “Moment” of a variable for the surface of a selected subregion are defined as:

$$F_i = \int (\Phi + \Phi_0) n_i dA$$

$$M_i = F_i \otimes (X_i - Y_i)$$

Where

 $F_i$  is the force vector for the  $\Phi$  variable, $\Phi_0$  is a reference datum for the  $\Phi$  variable, $n_i$  is the normal vector at the surface of the subregion,

A is the area of the surface,

 $M_i$  is the moment vector for the  $\Phi$  variable, $\otimes$  denotes the cross product of vectors, $X_i$  is the location of the pivot point around which the moment is computed, $Y_i$  is the point of intersection of the vector from  $X_i$  and the force vector such that the vector from  $X_i$  is normal to force vector. This is automatically computed.**MOME** Same as **FORCE** modifier.**subrgn** The subregion for computations. If no subregion is specified, the entire domain is selected.**dir** The orientation index for the boundary of the subregion for which the force is computed. See Section 3.5 for available choices. If no value is specified then the total (closed) boundary of the subregion is elected. In this case the computed force will be the net force on the selected body. $X_i$  The coordinates of the pivot point around which the moment of the force is computed. Two values must be specified for 2D and three for 3D geometry. If specified these must be the first set of numerical values on the command. The default value is zero. $\Phi_0$  Datum for the selected variable  $\Phi$ . If specified this must be the 3<sup>rd</sup> numerical value for 2D and 4<sup>th</sup> numerical value for 3D geometry. The default value is zero. $V_{freq}$  The frequency (step or time interval) at which the output is written to the output device. See Section 3.7 for further details. If specified this must be the 4<sup>th</sup> numerical value for 2D and 5<sup>th</sup> numerical value for 3D geometry. By default the output is obtained only at the end of simulations.**TIME** By default,  $V_{freq}$  is interpreted as the frequency of output in terms of number of steps. If this modifier is present, then it is interpreted as the time interval between successive outputs.**NOW** The force and moment are computed at the first step after the command.**OFF** Any previously specified command for the specified  $\Phi$  and **subrgn** is deactivated.**fname** The file name for output. A different output file may be specified for each command. If no file name is specified then the output is directed to the most recent previously specified file. If no previous file was specified, then the output is directed to a file with the same name as the Standard Output file but with the extension removed and replaced by “\_FORCE.TMP”. For example, if the Standard Output file is “PROBLEM.OUT”, then the file is named “PROBLEM\_FORCE.TMP”. A summary of output is also printed to the standard output device at the end of simulations.

**EXAMPLES**

---

**PRINT** FORCE for ID=BODY

**PRINT** FORCE of P for ID=BODY in X+ direction with moments around point (0., 2.354, 5.75)

**PRINT** FORCE of T for ID=BODY in X+ dir; around (0., 2.35, 5.7); add Pref = 1.E5 file='FORCE\_FIL

**PRINT** FORCE of U ID=BODY in Y+ dir with moments (0., 0.); add Pref = 1.E5 every 20 steps! [2D case](#)

**PRINT** FORCE ID=BODY in Z+ dir; pivot (0., 2.35, 5.7); Pref = 1.E5, every 20 steps

**PRINT** FORCE Z+ of ID=BODY; pivot (0., 2.35, 5.7); Pref = 1.E5, every TIME = 100 'FORCE\_OUT' NOW

**PRINT** FORCE of P OFF for Z+ direction of ID=BODY

**MODE 5:** Convective and Diffusive Fluxes for a Variable at Specified Boundaries

**SYNTAX** PRIN {FLUX} [Φ] [DETA] [option] [subrgn] [dir] [fname] [V<sub>frq</sub>] [TIME] [NOW] [OFF]

**FLUX** The convective and diffusive fluxes for the variable at specified boundaries are written to the output file. The output includes the mean, minimum and maximum values for the variable, and the flow rate at the boundary (or boundaries).

**Φ** One or more symbols for the dependent variable for which output is required. Up to 10 symbols may be specified per command. The valid symbols are listed in Table 2.7.1. If no symbol is specified then output is obtained for each active variable for which a transport equation is solved.

**DETA** By default only a summary of the total fluxes and variable values for each boundary is written to the output device. If this modifier is present, then output also contains the detail for each segment (element) of the boundary.

**option** The boundary type for which output is required. More than one option may be selected. If an option is specified, then the subrgn and dir modifiers are ignored. If no option is specified, then the averages are computed for the specified subrgn and dir modifiers.

option	INTERPRETATION
<b>INLE</b>	Boundaries specified by the <b>INLET</b> command are selected.
<b>OUTL</b>	Boundaries specified by the <b>OUTLET</b> command are selected.
<b>OPEN</b>	Boundaries specified by the <b>OPEN</b> command are selected.
<b>IO</b>	All boundaries specified by <b>INLET</b> , <b>OUTLET</b> or <b>OPEN</b> command are selected.
<b>WALL</b>	Walls specified by <b>WALL</b> or <b>BLOCK</b> command are selected.
<b>EXTE</b>	All external (or outer) boundaries of the computational domain are selected.
<b>ALL</b>	All of the above boundaries are selected.

**subrgn** The subregion for computations. This specification is ignored if any of the option modifiers is present. If no subregion is specified, the entire domain is selected.

**dir** The orientation index for the boundary for which the output is required See Section 3.5 for available choices. If no input is given, then the output is obtained for all boundaries of the subrgn. This specification is ignored if any of the option modifiers is present.

**fname** The file name for output. If a file name is present then the output is directed to the named file, otherwise the output is directed to the standard output device. The total number of open files in any simulation cannot exceed 64.

**V<sub>frq</sub>** The frequency (step or time interval) at which the output is written to the output device. See Section 3.7 for further details. By default the output is obtained only at the end of simulations.

**TIME** By default, V<sub>frq</sub> is interpreted as the frequency of output in terms of number of steps. In the presence of TIME this is the time intervals between successive outputs.

**NOW** A record of fluxes is written to the output device at the next time step when the equation for the dependent variable is solved. This is in addition to the output from the V<sub>frq</sub> specification.

**OFF** Any previously specified PRINT FLUX commands for the specified variables and subregion are disabled. New commands may be subsequently specified.

**EXAMPLES**

- PRINT FLUX for EXTErnal boundaries for all variables
- PRINT FLUX of U, V and T for INLET boundaries at every 100 steps
- PRINT FLUX of U and T for X- direction of ID=OBJECT at every TIME=20 units
- PRINT FLUX of T for X+ direction of currently SELEcted subregion at the end of simulations
- PRINT FLUX of T for OUTLET boundaries at every 100 steps on file "FLUX\_T\_OUTLET.FIL"
- PRINT FLUX of T for ALL boundaries NOW and at every TIME=20 units on file "FLUX\_T\_ALL.FIL"
- PRINT FLUX of T OFF for X- direction of ID=OBJECT

**MODE 6:** Convective and Diffusive Fluxes for a Variable at Specified Boundaries

**SYNTAX** PRIN {SHEA} [DETA] [option] [subrgn] [dir] [fname] [V<sub>frq</sub>] [TIME] [NOW] [OFF]

**SHEA** The shear stresses and  $y^+$  at specified boundaries are written to the output file. The  $y^+$  is the normalized distance to the node, which is meaningful for turbulent boundary layer type of flows. For laminar flow, it is reported as zero. The output also includes the normal distance from the boundary to the nearest fluid node and the mean velocity at the near boundary nodes. Strictly speaking, the output consists of the momentum gain or loss at the boundary and not the shear stress. However, in most instances where a boundary-layer type of flow occurs, the two can be considered to be equivalent.

**DETA** By default only a summary of the total fluxes and variable values for each boundary is written to the output device. If this modifier is present, then output also contains the detail for each segment (element) of the boundary.

**option** The boundary type for which output is required. More than one option may be selected. If an option is specified, then the **subrgn** and **dir** modifiers are ignored. If no option is specified, then the averages are computed for the specified **subrgn** and **dir** modifiers.

option	INTERPRETATION
<b>INLE</b>	Boundaries specified by the <b>INLET</b> command are selected.
<b>OUTL</b>	Boundaries specified by the <b>OUTLET</b> command are selected.
<b>OPEN</b>	Boundaries specified by the <b>OPEN</b> command are selected.
<b>IO</b>	All boundaries specified by <b>INLET</b> , <b>OUTLET</b> or <b>OPEN</b> command are selected.
<b>WALL</b>	Walls specified by <b>WALL</b> or <b>BLOCK</b> commands are selected.
<b>EXTE</b>	All external (or outer) boundaries of the computational domain are selected.
<b>ALL</b>	All of the above boundaries are selected.

**subrgn** The subregion for computations. This specification is ignored if any of the **option** modifiers is present. If no subregion is specified, the entire domain is selected.

**dir** The orientation index for the boundary for which the output is required See Section 3.5 for available choices. If no input is given, then the output is obtained for all boundaries of the **subrgn**. This specification is ignored if any of the **option** modifiers is present.

**fname** The file name for output. If a file name is present the output is directed to the named file, otherwise the output is directed to the standard output device. The total number of open files cannot exceed 64.

**V<sub>frq</sub>** The frequency (step or time interval) at which the output is written to the output device. See Section 3.7 for further details. By default the output is obtained only at the end of simulations.

**TIME** By default, **V<sub>frq</sub>** is interpreted as the frequency of output in terms of number of steps. In the presence of **TIME** this is the time interval between successive outputs.

**NOW** A record of fluxes is written to the output device at the next time step when the equation for the dependent variable is solved. This is in addition to the output from the **V<sub>frq</sub>** specification.

**OFF** Any previously specified **PRINT SHEAR** commands for the specified variables and subregion are disabled. New commands may be subsequently specified.

## EXAMPLES

**PRINT SHEAR** stress for all WALLS

**PRINT SHEAR** for EXTERNAL boundaries every 100 steps

**PRINT SHEAR T** for X- direction of ID=OBJECT at every TIME=20 units

**PRINT SHEAR** for X+ direction of currently SELEcted subregion at the end of simulations

**PRINT SHEAR** for OUTLET boundaries at every 100 steps on file "SHEAR\_T\_OUTLET.FIL"

**PRINT SHEAR** for ALL boundaries NOW and at every TIME=20 units on file "SHEAR\_T\_ALL.FIL"

**PRINT SHEAR OFF** for X- direction of ID=OBJECT

## 5.82 PROBLEM

**PURPOSE** To specify the general nature and type of problem to be solved.

**MODE 1:** Atmospheric Pressure Computation Mode

**SYNTAX** PROB {ATMO}

**ATMO** This command mode is applicable only to the ANSWER® Software Tool. It invokes the “atmospheric” mode of ANSWER®. By default the body force in the momentum equations is computed as  $(\rho^* \mathbf{g}_j)$  where  $\mathbf{g}_j$  is the body force vector component in the  $x_j$  direction. If this modifier is present, then the body force is computed as  $(\rho - \rho^*) \mathbf{g}_j$ . This effectively adds hydrostatic pressure  $(\rho^* \mathbf{g}_j x_j)$  to the computed pressure. In this case, the thermodynamic pressure,  $p$ , is related to the computed pressure by:

$$p = p_{\text{computed}} + \rho^* \sum_j (x_j^* - x_j) g_j$$

where  $x_j^*$  is the coordinate datum in the  $j^{\text{th}}$  direction which can be specified by the DATUM command. For this mode, all initial and boundary conditions for pressure must be specified in terms of the computed pressure,  $p_{\text{computed}}$ ; that is the pressure boundary values must be in terms of the thermodynamic pressure minus the hydrostatic pressure.

**REMARKS**

This command provides a convenient option for solving the problem with hydrostatic pressure. This is particularly convenient for atmospheric applications where often the boundary pressure is specified as a linear function of height from ground level.

**EXAMPLES**

**PROBLEM** to be solved with the ATMOSPHERIC mode of ANSWER®

**MODE 2:** Boussinesq Approximation for Natural Convection

**SYNTAX** PROB {BOUS} {ξ} {β} {ξ<sub>o</sub>}

**BOUS** This command mode is applicable only to the ANSWER® Software Tool. It invokes the “Boussinesq” approximation mode of ANSWER®. In this mode, a source terms for body force is added to momentum equations. The source term,  $S_i$ , for the  $i^{\text{th}}$  direction velocity component,  $u_i$ , is given by:

$$S_i = \beta (\xi - \xi_o) g_i$$

where  $g_i$  is the gravitational acceleration vector in the  $i^{\text{th}}$  direction and the other symbols are defined below. See comments below for further discussion.

**ξ** One, and only one, symbol to denote the variable which controls the body force. Most commonly, the body force is created by a variation of density due to temperature. In this case,  $\xi$  is the symbol T (for temperature). The valid symbols are listed in Table 2.8.1-3

**β** The numerical value for the coefficient for the body force defined as:

$$\beta = \frac{\partial \rho}{\partial \xi}$$

where  $\rho$  is the density of the fluid. If the working fluid is a gas and the perfect gas law applies, then  $\beta$  can be estimated from:

$$\beta = - \frac{\rho_o}{T_o}$$

where  $T_o$  is a reference temperature and  $\rho_o$  is a reference density. For air at atmospheric pressure and 300K,  $\beta \approx -0.004$  [Kg/(m<sup>3</sup>K)]. For water at 300K,  $\beta \approx -0.2$  [Kg/(m<sup>3</sup>K)].

**ξ<sub>o</sub>** The numerical value for the reference state at which there is no net body force in the fluid. Typically this should be equal to the mean value of  $\xi$  in the computational domain.

## REMARKS

This command provides a convenient way to incorporate body forces with Boussinesq assumption. In this case, the density for the fluid is assumed to be constant and the variation of density is accounted for only in the body force term which is specified as a linearized function of the controlling variable, say, temperature.

If the body force is a function of more than one variable (say, density is a function of both temperature and a species concentration) then the ALLOCATE command can be used to define a composite variable which is a function of these variables. For example if the source term is given by:

$$S_i = \{ \beta_1 (T - T_o) + \beta_2 (C - C_o) \} g_i$$

Then the ALLOCATE command can be used to specify a composite variable  $\xi$  with:

$$\xi = \beta_1 T + \beta_2 C; \quad \beta = 1.; \quad \xi_o = \beta_1 T_o + \beta_2 C_o$$

The composite variable,  $\xi$ , in turn can be defined by an appropriate SET command with the ALWAYS modifier so that it is continuously updated as T and C change.

## EXAMPLES

**PROBLEM** BOUSSINESQ approximation: function of T with beta = -0.004, T<sub>o</sub>=300

**MODE 3:** Other Geometry or Client Specific Options

**SYNTAX** **PROB** {**RADI**|**option**} [**attributes**] [**N<sub>1</sub>**, ..., **N<sub>4</sub>**]

**RADI** This command mode is applicable only to the **ANSWER®** Software Tool. If the geometry is Cartesian then, by default, the velocity components are also Cartesian. If this modifier is present, then it is assumed that the velocity components are in cylindrical radial system. In this case, the velocity component in the x-direction is identical to that in the Cartesian system, but the components in the y and z direction are assumed to be the radial and circumferential components, respectively.

**option** One of the special options implemented by **ACRi** in response to specific user requirements. Please call **ACRi** for the exact option(s), if any, available for your installation.

**N<sub>1</sub>, ..., N<sub>4</sub>** Numerical input relevant to the implemented option.

#### **REMARKS**

---

The purpose of this command is to provide users access to special options implemented by **ACRi** in response to user requests.

#### **EXAMPLES**

---

**PROBLEM** with RADial velocity in a geometry specified by Cartesian coordinates

**PROBLEM** special option for GEAE

**PROBLEM** special option for DOE to read GRID

**MODE 4:** Free Surface or Unconfined Flow

**SYNTAX** PROB {UNCO | FREE SURF} [dir [k<sub>r</sub>] | AREA]

**UNCO** This command mode is applicable only to the **PORFLOW®** Software Tool. It invokes the unconfined or free-surface mode **PORFLOW®** where the Richard's equation form of the pressure equation is solved. In this mode the hydraulic conductivity itself is a function of the pressure head or depth of saturation

**FREE** Same as **UNCO** but the modifier **SURF** must be simultaneously present.

**SURF** Same as **UNCO** but the modifier **FREE** must be simultaneously present.

**dir** The coordinate direction that is normal to the plane of the phreatic surface. This is the nominal "vertical" direction for the flow. This direction must be one of the **X, Y, Z, R** or **THETA** modifiers, which, respectively, denote the x, y, z, r or  $\theta$  coordinates. **By default, the y direction is assumed to be the direction of the normal to the phreatic surface for 2-D problems and the z direction for 3-D problems.** In this case, the effective hydraulic conductivity for the soil is computed as:

$$K_i^{\text{effective}} = S_f K_i^{\text{saturated}}$$

where  $K^{\text{effective}}$  is the hydraulic conductivity used in the governing equations,  $K^{\text{saturated}}$  is the value specified by the user and  $S_f$  is the local saturation of the soil. This relation leads to a hydraulic conductivity that is zero for soil above the water table. This is unsatisfactory if infiltration occurs at a boundary above the water table or if a source of flow is located above the water table. In this case, a better assumption is that the "vertical" component of hydraulic conductivity retains a minimum value. The above relation is then replaced by:

$$K_{\text{dir}}^{\text{effective}} = k_r K_{\text{dir}}^{\text{saturated}} ; S_f < 1$$

Where  $k_r$  is the relative conductivity for the dry soil and the subscript "dir" denotes the direction specified by **dir**.

**k<sub>r</sub>** The relative hydraulic conductivity if the modifier **dir** is specified. **The default value is 1.0.** The default value is satisfactory for most cases. However it does lead to a thickening of the interface (fringe) between the saturated and the dry soil. It will also lead to transport of contaminants from the water table to the dry soil since there is a finite "vertical" hydraulic conductivity at the water table if  $k_r > 0$ . In such cases, provided there is no infiltration or source above the water table, the  $k_r$  should be set to a small number or even zero.

**AREA** Unconfined areal mode is selected. In this mode, the equations are solved in a two-dimensional integrated form where the effect of the third direction appears as a dynamically variable depth of the water body.

## REMARKS

---

**By default, the upper and lower limits for the pressure head are automatically computed from the coordinates and datum specifications.** The default limits represent the phreatic surface when the surface reaches either the bottom of the aquifer or the top of the ground surface. However, the user may explicitly alter these limits. Alteration may be necessary, for example, when the bottom of the aquifer is not at the bottom of the grid system or when ponding occurs above the ground surface. In these cases, the lower value of the pressure head (representing the bottom of the aquifer) is specified as the P2 pressure field and the upper value (representing the top of the ponding surface) is specified as the P3 pressure field.

## EXAMPLES

---

**PROBLEM** is UNCONFINED

**PROBLEM** with FREE SURFACE; bedding plane normal to direction Y

**PROBLEM** with FREE SURFACE in X direction; dry soil  $k_r = 0.1$  of the saturated value

**PROBLEM** with FREE SURFACE in AREAL mode

**MODE 5:** Active Two or Multiple Phase with Variable Saturation

**SYNTAX** PROB {TWO|MULT}

**TWO** This command mode is applicable only to the **PORFLOW®** Software Tool. It invokes the active two-phase variable saturation mode. A pressure equation is solved for each of the two phases. The saturation of the 1<sup>st</sup> phase is determined from the capillary pressure (the difference between the 2<sup>nd</sup> and 1<sup>st</sup> phase pressure). The saturation of the 2<sup>nd</sup> phase is equal to one minus the saturation of the 1<sup>st</sup> phase.

**MULT** This command mode is applicable only to the **PORFLOW®** Software Tool. The general multi-phase variable saturation mode is selected. The number of pressure equations solved is determined from user input (see comments below). This mode is also automatically activated by a **MULTIPHASE** command. The saturation of the last phase is always obtained as one minus the sum of the saturation of the other phase.

#### REMARKS

---

For a typical vadose zone problem, involving water as the primary phase and air as the secondary phase, there are two phases but only 1 pressure equation is solved. It is assumed that the pressure of air phase is constant for this type of flow only 1<sup>st</sup> (water) phase is active; the 2<sup>nd</sup> (air) phase is passive.

If the 2<sup>nd</sup> phase is also active and its pressure is not constant, then a second pressure equation must be solved. In this case the **MULT** phase command with **TWO** modifier must be specified.

The next stage of complexity in multiphase flow is reached when there are two active phases of immiscible liquids (say, water and oil) and one passive phase (say, air). The pressure of the air phase is assumed constant and only 2 pressure equations, those for water and oil phase, are solved.

The most complex option involves three active phases. In this case 3 pressure equations, one for each phase, are solved.

With the exception of the two active phases, all other modes of solution can be automatically determined by **PORFLOW®** from the user input.

#### EXAMPLES

---

**PROBLEM** is TWO phase with water and oil as the liquid

**PROBLEM** is MULT phase with three phases

**MODE 6:** Phase Change with Freezing and Thawing

**SYNTAX** PROB {FREEZ} [V<sub>1</sub>]

**FREEZ** This command mode is applicable only to the **PORFLOW®** Software Tool. It invokes the Freezing/thawing phase-change mode of **PORFLOW®**. All 5 characters must be specified.

**V<sub>1</sub>** An empirical constant to determine the maximum rate of freezing or thawing from step to step. A small value results in gradual changes in freezing/thawing behavior; a large value results in more abrupt changes from one time step to another. The default value is 1000.

#### EXAMPLES

---

**PROBLEM** with phase change: FREEZing

**PROBLEM** with phase change: FREEZing rate=500

**MODE 7:** Phase Change with Evaporation and Condensation

**SYNTAX** PROB {EVAP}

**EVAP** The Evaporation/condensation phase-change mode is selected.

**APPLICABILITY** \_\_\_\_\_

This command is available only for the **ANSWER®** and **PORFLOW®** Software Tools.

**REMARKS** \_\_\_\_\_

The phase change is assumed to be in thermodynamic equilibrium and the phase saturation is computed from enthalpy equation.

**EXAMPLES** \_\_\_\_\_

**PROBLEM** with EVAPoration

**MODE 8:** Phase Change with Thermodynamic Non-Equilibrium

**SYNTAX** **PROB** {**EVAP**} [**DYNA**] [**FIXE**] [**C<sub>m</sub>**, **δT**, **α**, **R<sub>f</sub>**]

**EVAP** The Evaporation/condensation phase-change mode is selected.

**DYNA** The non-equilibrium mode of phase change is employed (see **REMARKS** below). The vapor formation rate is computed from:

$$m_v = C_m \Theta_E S^\alpha \frac{p_s - p_v}{p_g}$$

Where:

$m_v$  is the rate of evaporation or condensation,  
 $C_m$  is an empirical constant (see below),  
 $\Theta_E$  is the effective porosity  
 $S$  is the saturation fraction of the fluid,  
 $\alpha$  is an empirical exponent,  
 $p_s$  is the saturation pressure of the vapor at equilibrium,  
 $p_v$  is the actual thermodynamic pressure of the vapor, and  
 $p_g$  is the thermodynamic pressure of the gas phase.

**FIXE** The time constant, **C<sub>m</sub>**, for phase change is adjusted for transient flow to account for the changes in time step. Presence of the **FIXED** modifier results in **C<sub>m</sub>** being held at its specified value.

**C<sub>m</sub>** The time constant  $C_m$  in the equation above. The constant has units of [t<sup>-1</sup>] and represents the speed of phase change due to a balance between hydraulic conductivity (to replenish the fluid), thermal energy (to satisfy the enthalpy demand) and mass transfer (to restore equilibrium). The numerical value represents the limiting (slowest) process among these three. The default value is 0.0025.

**δT** Thermal tolerance for phase change. If the local temperature differs from the equilibrium value by less than **δT**, the effect of phase change on thermal energy balance is ignored. The default value is 0.01.

**α**  $\alpha$  is the exponent in the above equation. The default value is 0.0625.

**R<sub>f</sub>** The relaxation factor for the source term. The new source term at any stage is computed as a weighted average of the previous value and the newly computed value. The default value is 1.0.

#### APPLICABILITY

This command is available only for the **PORFLOW®** Software Tool.

#### REMARKS

Most phase change applications stay within the bounds of thermodynamic equilibrium. The exception may be extremely fast and explosive rate of phase change. This mode is computationally very sensitive and must be employed only if there are strong indications of non-equilibrium phase change. If the equation that determines the rate of evaporation and condensation is known, then the source terms may be introduced in the appropriate phase equations to control the phase change.

#### EXAMPLES

**PROBLEM** DYNAMIC EVAPORATION:  $C_m=0.01$ ,  $dt=0.02$ ,  $power=0.20$ ,  $RF=0.5$

5.83 PROPERTY

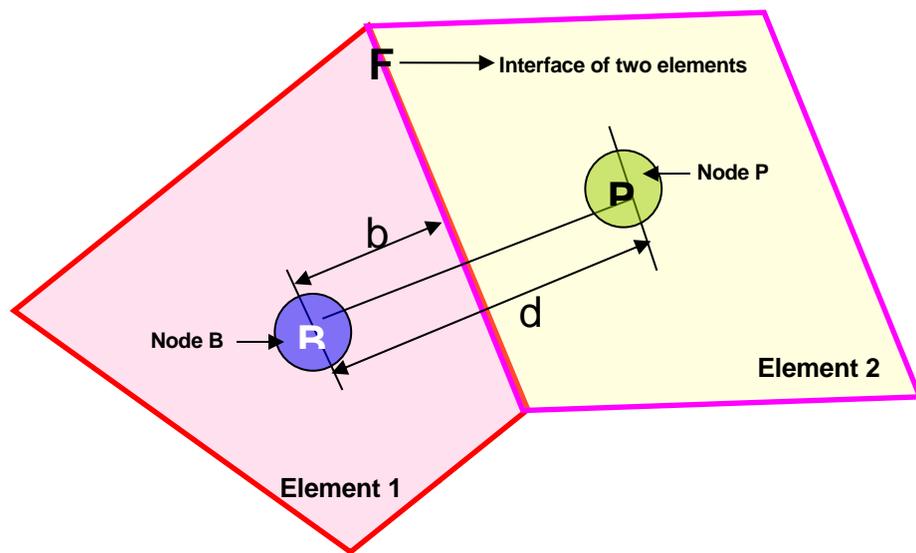
**PURPOSE** To specify the mode of calculation of fluid or material properties at an interface that lies between two elements where the properties are known at the element nodes.

**MODE 1:** The Interface Conductivity and Diffusivity Mode

**SYNTAX** PROP [ $\Phi$ ] [ARIT|HARM|GEOM|UPWI|MINI|MAXI|TRAV] [SATU] [ $\Lambda$ ]

$\Phi$  One or more symbols that represent corresponding variable(s) for which the input is effective. The valid symbols are listed in Table 2.7.1. If no symbol is specified, the input is assumed to be effective for all variables.

**ARIT:** The interface value is the weighted arithmetic mean of the values at nodes B and P. This is the default option for the ANSWER® Software Tool. With respect to figure below, the interface value of a property,  $\Gamma_f$ , is obtained from the node values,  $\Gamma_B$  and  $\Gamma_P$ , by:



$$\Gamma_F = \alpha \Gamma_P + \beta \Gamma_B$$

$$\alpha = \frac{b}{d}; \quad \beta = 1 - \alpha$$

Where b is the distance from the node B to the interface and d is the distance from node B to P, respectively.

**HARM:** The interface value is the weighted harmonic mean of the values at nodes B and P. This is the default option for the TIDAL® Software Tool. It is also the default option for the PORFLOW® Software Tool in the single-phase mode. The interface value of the property is computed as:

$$\Gamma_F = \frac{\Gamma_P \Gamma_B}{\alpha \Gamma_B + \beta \Gamma_P}$$

**GEOM:** The interface value is the weighted geometric mean of the values at nodes B and P which is computed as:

$$\Gamma_F = \Gamma_P^\alpha \Gamma_B^\beta$$

**UPWI:** The interface value is taken to be the upstream value. With V as the velocity vector directed

from B towards P; this value is given by:

$$\begin{aligned}\Gamma_F &= \Gamma_B \quad \text{if } V > 0 \\ &= \Gamma_P \quad \text{if } V < 0 \\ &= \alpha \Gamma_P + \beta \Gamma_B \quad \text{if } V = 0\end{aligned}$$

**MINI:** The interface is the minimum of the values at nodes B and P; that is:

$$\Gamma_F = \min(\Gamma_B, \Gamma_P)$$

**MAXI:** The interface is the maximum of the values at nodes B and P; that is:

$$\Gamma_F = \max(\Gamma_B, \Gamma_P)$$

**SATU** The interface value is computed from a harmonic average weighted with the local phase saturation. This modifier is effective only if the **HARMONIC** modifier is also present. This is the default option in **PORFLOW®** if the multiphase problem mode is invoked. With S denoting the phase saturation, the interface value is computed as:

$$\Gamma_F = \Gamma_P \Gamma_B \frac{\alpha S_P^{1/\lambda} + \beta S_B^{1/\lambda}}{\alpha S_P^{1/\lambda} \Gamma_B + \beta S_B^{1/\lambda} \Gamma_P}$$

**TRAV** The TRAVel option modifies the behavior of the ARITmetic, HARMonic and GEOMetric interpolation modes by switching the weighting factors in the interpolation formula.

**OFF** The OFF option works in conjunction with the TRAVel option to restore the default weighting factors for the interpolation.

**λ** The inverse of the power exponent of saturation if the **SATURATED HARMONIC** option is invoked; otherwise this input is ignored. The specified value must be an integer between 1 and 9. The default value is 2.

## REMARKS

---

This command is effective in **PORFLOW®** for computing the interface hydraulic conductivity, thermal conductivity and diffusivity of the mass species. In **ANSWER®**, this command is effective only if the conductivity or diffusivity for a governing variable is declared to be a vector (See **CONDUCTIVITY** command). In **TIDAL®** software it is effective only for the thermal and species equations.

## EXAMPLES

---

**PROP**erty at interface for all variables by arithmetic average  
**PROP**erty for U use GEOMetric mean  
**PROP**erty averages for T and C by UPWInd mean  
**PROP**erty for C by MAXImum method  
**PROP**erty at interface for all variables by arithmetic average

**MODE 2:**      **The Effective Fluid-Solid Matrix Property Mode**

**SYNTAX**      **PROP [Φ] [EFFE|TOTA]**

**Φ**              One or more symbols that represent corresponding variable(s) for which the input is effective. The valid symbols are those for the mass species listed in Table 2.7.1. If no symbol is specified, the input is assumed to be effective for all relevant variables. (See comments below for important information)

**EFFE**         This modifier is effective only for the **PORFLOW®** Software Tool. It modifies the interpretation of the 1<sup>st</sup> values given on the **TRANSPORT** commands. By default the 1<sup>st</sup> value is assumed to be the partition coefficient,  $k_d$ , for the species. If this modifier is present then this input is assumed to be the retardation factor,  $R_d$ .

**TOTA**         This modifier is effective only for the **PORFLOW®** Software Tool. **The use of the TOTAL modifier on this command has been superceded by the DISTRIBUTION command which provides more flexibility in selecting the inputs to which it applies. It is being retained here only for backward compatibility and its use is not recommended. See the DISTRIBUTION command for interpretation of this modifier.**

**REMARKS** \_\_\_\_\_

This mode of the **PROPERTY** command must be specified before the **TRANSPORT** command to which it applies; it will then apply to all **TRANSPORT** commands that follow till the next **PROPERTY** command.

This command mode may be combined with the Mode 1 of the command.

**EXAMPLES** \_\_\_\_\_

**PROP**erty in **EFFE**ctive mode for C

**PROP**erty in **EFFE**ctive mode for P by **HARMONIC SATURATED** based with power 4.

**PROP**erty C present in **TOTAL**: solid.

**5.84 QUIT**

**PURPOSE** To signify the termination of a sequence of problems.

**SYNTAX** **QUIT**

**REMARKS**

---

This command terminates a sequence of problems. The **FREEFORM™** command language does not process any input following this command. Thus, for a sequence of problems, an **END** command terminates each problem whereas the **QUIT** command terminates the entire segments of problems. An example of the use of this command is when, for example, there are, five problems in an input sequence but the user desires to solve only the first two. In this instance, a **QUIT** command may be inserted immediately after the **END** command for the second problem. The input for the other three problems will then be ignored.

**EXAMPLES**

---

**QUIT** Termination of sequence. No more problems to solve.

**5.85 RADIATION**

**PURPOSE** To activate thermal radiation calculations and specify radiation constants. This command is effective only for the **ANSWER®** Software Tool.

**MODE 1:** **Radiation Approach and Thermal Source Term**

**SYNTAX** **RADI [VIEW] [PASS]**

**VIEW** By default the radiation flux is computed from the 6-flux radiation model. If this modifier is present, then the radiation flux is computed from the View Factor approach. While the 6-flux radiation approach is available as a standard option with the Professional Versions of **ACRi** Software, the View Factor approach is available only by special arrangement and under a consulting contract with **ACRi**.

The 6-flux approach is best suited for “participating” media where the absorption and/or scattering phenomena dominate radiative transport. This is the case, for example, for combustion systems with high concentrations of gases that absorb and re-emit (such as CO<sub>2</sub> and H<sub>2</sub>O) and contain scattering particles (such as soot). For “non-participating” media (such as air), this approach is not well suited. In these cases the View Factor approach is more appropriate.

**PASS** By default the net radiation flux contributes to the thermal source in the enthalpy equation. If this modifier is present, then the effect of radiative heat flux is ignored in the enthalpy equation. For all practical purposes this is equivalent to a non-participating fluid medium.

**REMARKS**

---

The modifiers of this mode can be combined with any of the other modes.

**EXAMPLES**

---

**RADI**ation by **VIEW** factor approach.

**RADI**ation computations in **PASS**ive mode

**RADI**ation computations by **VIEW** factor method in **PASS**ive mode

**MODE 2:**      **Radiation Computation Frequency**

**SYNTAX**      **RADI {FREQ} [V<sub>freq</sub>]**

**FREQ**              The frequency of computation of radiation fluxes.

**V<sub>freq</sub>**              The frequency in terms of number of steps of the solution process. The numerical value must be >0; otherwise the input is ignored and the default or any previously specified value is used. The default value is 10.

#### **EXAMPLES**

---

**RADI**ation **FREQ**uency of computations is 20 steps.

<b>MODE 3:</b>	<b>Specification of Gas Absorptivity</b>
<b>SYNTAX</b>	<b>RADI {ABSOR} {func[ξ]} {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub>   fname} [subrgn]</b>
<b>ABSOR</b>	The absorptivity for radiation is specified. It is to be noted that all 5-characters of the modifier <b>ABSOR</b> must be present, since the 4 character modifier <b>ABSO</b> is used to denote the absolute value of the function (see Section 4.3). By default gas absorptivity is computed from the built-in formulae, which take account of gas and soot constituents.
<b>func</b>	One of the modifiers listed in Table 4.2.1 that denotes the functional form of gas absorptivity. If no function is specified then the value is assumed to be constant.
<b>ξ</b>	One of the independent variables listed in Table 4.2.2. If no variable is specified, then the independent variable is assumed to be time.
<b>N<sub>1</sub>,...,N<sub>n</sub></b>	The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.
<b>fname</b>	The name of the file containing the numerical values <b>N<sub>2</sub></b> through <b>N<sub>n</sub></b> . This option is available only for selected functions. See Section 3.3 for additional information.
<b>subrgn</b>	The sub region for which the input is specified. See Section 3.4. If no sub region is specified, the entire computational domain is selected.

**REMARKS**

The absorptivity is denoted by the symbol "**ARAD**" in ACRi software. The **SET** command with **ARAD** modifier may also be used to set the value of gas absorptivity. This symbol can also be used to obtain output of gas absorptivity.

**EXAMPLES**

Generic examples for this command are given in Section 4.4. The command keyword (**RADI**) with modifier **ABSOR** must replace the keyword used in these examples.

<b>MODE 4:</b>	<b>Specification of Surface Emissivity</b>
<b>SYNTAX</b>	<b>RADI {EMIS} {func{ξ}} {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub> fname} [[subrgn] [dir] WALL] [EXTE INTE BLOC]</b>
<b>EMIS</b>	The emissivity of a surface is specified. By default the emissivity for all surfaces is set to unity. However if this command is present, and no value is specified, then the default value is set to 0.8.
<b>func</b>	One of the modifiers listed in Table 4.2.1 that denotes the functional form of surface emissivity. If no function is specified then the value is assumed to be constant.
<b>ξ</b>	One of the independent variables listed in Table 4.2.2. If no variable is specified, then the independent variable is assumed to be time.
<b>N<sub>1</sub>,...,N<sub>n</sub></b>	The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.
<b>fname</b>	The name of the file containing the numerical values N <sub>2</sub> through N <sub>n</sub> . This option is available only for selected functions. See Section 3.3 for additional information.
<b>subrgn</b>	The sub region for which the input is specified. See Section 3.4. If no sub region is specified, the entire computational domain is selected.
<b>dir</b>	Orientation index for the boundary to which the input is applied. See Section 3.5 for available choices. If <b>dir</b> is omitted, then all surfaces of the selected <b>subrgn</b> are selected.
<b>WALL</b>	If this modifier is present then any specification of <b>subrgn</b> and <b>dir</b> is ignored and the input is applied to walls of the computational domain. The input is applied to all the walls unless one of the modifiers <b>EXTE</b> , <b>INTE</b> or <b>BLOC</b> is present.
<b>EXTE</b>	If this modifier is present along with the <b>WALL</b> modifier, then the input is applied only to the exterior walls of the domain.
<b>INTE</b>	If this modifier is present along with the <b>WALL</b> modifier, then the input is applied only to the interior walls of the domain defined with the <b>WALL</b> command.
<b>BLOC</b>	If this modifier is present along with the <b>WALL</b> modifier, then the input is applied only to the walls of the domain defined with the <b>BLOCK</b> command.

#### EXAMPLES

---

Generic examples for this command are given in Section 4.4. The command keyword (**RADI**) with modifier **EMIS** must replace the keyword used in these examples.

**MODE 5:**      **Radiation Constants**

**SYNTAX**      **RADI** [SCAT= $V_{scatter}$ ] [STEF|BOLT= $\sigma$ ]

**SCAT**            The scattering coefficient for the gas is specified. The default value is 0.01.

$V_{scatter}$         The numerical value for the Scattering coefficient. It must be  $\geq 0$ .

**STEF**            Stefan-Boltzmann constant is specified. The default value is 5.669E-8

**BOLT**            Stefan-Boltzmann constant is specified. The default value is 5.669E-8

$\sigma$               The numerical value for the Stefan-Boltzmann constant. It must be  $> 0$ .

#### **EXAMPLES**

---

**RADI**ation SCATtering coefficient is 0.02

**RADI**ation STEFAn-Boltzmann constant is 4.76E-13 in fps units.

**RADI**ation SCATtering coefficient= 0.02, STEFAn= 4.76E-13 in fps units.

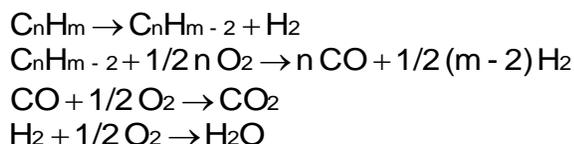
## 5.86 REACTION

**PURPOSE** To select the nature and mode of control of chemical reaction for reactive flows.

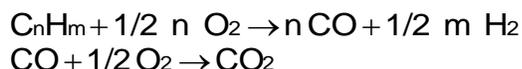
**MODE 1:** **The Built-In 2 or 4-Step, 8 Species Reaction Mechanism.**

**SYNTAX** **REAC [TWO] [KINE | PROD | HYDR | USER | OFF]**

**TWO** By default, the reaction mechanism is composed of four steps:



If this modifier is present, then the reaction mechanism is assumed to be:



See Section 3.5 and comments below for further details.

**KINE** By default, the reaction rate for the built-in reaction mechanism is taken to be the smaller of the chemical kinetics (Arrhenius) reaction rate and the eddy break up (EBU) rate to account for turbulence. If this modifier is present, then the EBU model is deactivated and the reaction rate is that given by the chemical kinetics.

**PROD** This modifier is used if the user selects the explicit reaction mode (Mode 2) but still wants to employ the equilibrium algebraic relations to determine the mass concentrations of O<sub>2</sub>, H<sub>2</sub>O, CO<sub>2</sub>, and N<sub>2</sub> from the solved species and conserved variables. This option assumes the reactants are only pure hydrocarbons and oxygen or air.

**HYDR** The reaction is for a hydrocarbon fuel but the user explicitly specifies the equilibrium relations for those chemical species for which a transport equation is not solved.

**USER** Reaction is assumed to be explicitly specified by the user (see Mode 2). The default 4-step reaction is switched off.

**OFF** Same as the **USER** modifier.

**REMARKS**

The default reaction systems consist of the 2 steps and 7 species or 4 steps and 8 species described in ANSWER® Theory Manual. The 2 step system, the gas phase consists of 7 chemical species: C<sub>n</sub>H<sub>m</sub>, CO, H<sub>2</sub>, O<sub>2</sub>, H<sub>2</sub>O, CO<sub>2</sub>, and N<sub>2</sub>. For the 4-step system, an 8<sup>th</sup> species, C<sub>n</sub>H<sub>m-2</sub>, is added. The default fuel is C<sub>10</sub>H<sub>19</sub>. This can be changed with the **FUEL** command. The default Arrhenius reaction constants are given in Table 3.5.1; these can be changed by the **ARRHENIUS** command. The EBU constants can be changed by the **EBU** command.

**EXAMPLES**

**REACTIVE** flow with default options

**REACTION** with TWO-step mechanism default options

**REACTION** process is KINETically controlled

**REACTION** process; use default PRODUct relations; explicit reactions supplied later

**REACTION** for HYDRocarbon fuel; next commands explicitly supply the reaction details

**REACTION** specified by USER. Explicit reactions of Mode 2 will be specified.

**MODE 2:** User-Specified Chemical Reaction Rate.

**SYNTAX** REAC {idreac} {Φ} [EBU] [TEMP] [subrgn] [option] {Z<sub>0</sub>, E, α, β, γ} [θ]

**idreac** A unique identifier for the reaction rate. The reaction rate is computed as:

$$\text{Rate} = Z_0 \exp\left(-\frac{E}{R_u T}\right) \Phi_1^\alpha \Phi_2^\beta \Phi_3^\gamma T^\theta$$

where Z<sub>0</sub> is the pre-exponential constant, E is the activation energy, R<sub>u</sub> is the universal gas constant, (E/R<sub>u</sub> is the activation temperature), T is the temperature in absolute units, α, β, γ and θ are exponents and Φ's are the reactant concentrations. The Z<sub>0</sub> is generally reported in the literature in molar units where the Φ's are molar concentrations. The actual computed Φ's in ACRi Software may be in mass, molar, volumetric or user specified units. Therefore the specified value of Z<sub>0</sub> may need to be converted to appropriate units internally.

**Φ** One or more symbols that specify the chemical species participating in the reaction as reactants. A maximum of three species can be specified.

**EBU** The reaction is limited by the eddy breakup (EBU) rate (see EBU command).

**TEMP** By default N2 specifies the activation energy, E (e.g. J/kg-mole). If this modifier is present, then N2 is the activation temperature (E/R<sub>u</sub>) in units of Kelvin.

**subrgn** The subregion for which the idreac reaction is computed. If no subregion is specified, the reaction is computed for the entire computational domain. See Section 3.4.

**option** The modifier for interpretation of numerical input

option	INTERPRETATION
<b>MASS</b>	By default it is assumed that the computed Φ's are in mass units and that Z <sub>0</sub> is given in molar units. Therefore Z <sub>0</sub> is internally converted to appropriate mass units. If this modifier is present, then no unit conversion is performed. It is assumed that Z <sub>0</sub> is specified in appropriate units for the computed Φ's.
<b>CGS</b>	By default, Z <sub>0</sub> and E are assumed to be in specified SI (kg-mole - m <sup>3</sup> - J - K) units. If this modifier is present, then Z <sub>0</sub> and E are assumed to be in the cgs (gm-mole - cm <sup>3</sup> - cal - K) units and the activation energy in calories. The specified values are converted to SI units on the assumption that the computed Φ's are mass concentrations.
<b>LIFE</b>	Z <sub>0</sub> is in units of time [t] representing the half -life of the reaction. In this case E must be the activation temperature [T].
<b>DECA</b>	Z <sub>0</sub> is in units of inverse of time [t <sup>-1</sup> ] representing the decay rate of the reaction. In this case E must be the activation temperature [T].

**Z<sub>0</sub>** The pre-exponential constant, Z<sub>0</sub>. The specified value may be internally converted to appropriate units based on the option modifier. For example, in the CGS system, units are cm<sup>3n-3</sup>/moles<sup>n-1</sup>-s, where n = α+β+γ.

**E** Activation energy, E, (J/kg-mole or cal/gm-mole) or activation temperature, E/R<sub>u</sub>, (Kelvin).

**α, β, γ** Stoichiometric coefficients or exponents α, β, γ and γ for the species specified by Φ's. There is no default value. One value must be specified for each identified reactant.

**θ** Exponent of the temperature, θ, in the reaction rate equation; it is the 4<sup>th</sup>, 5<sup>th</sup> or 6<sup>th</sup> numerical value depending on the order of the reaction (1, 2, or 3 reactants). The default value is 0.

**EXAMPLES**

---

**REAC**tion R1 preexp=1.960E+18, activ\_e=2.48E+04 FU ^ 0 and O2 ^ 0.5  
**REAC**tion R1 FU, O2 1.960E21 2.062E8 0.0 0.5 1.07 CGS EBU TEMP SELEcted subregion  
**REAC**tion R2 CH, O2, FU: 5.395E16, 2.50E04 0.0 0.9 1.18 -0.57 TEMP for ID=RGN1 active  
**REAC**tion R1 MASS based preexp=1.50, activ\_T= 600 FU ^ 1 and O2 ^ 1  
**REAC**tion R1 MASS FU, O2 1.50 0. 1 0.5 1.07 EBU! No effect of temperature  
**REAC**tion R2 DECAy frequency FU: 2.00, 1000 1.0 for SELEcted region  
**REAC**tion R2 between C1 & C2 Half Life = 0.01 Active\_T=0., exponents: 1 & 1 ID=RGN1

**MODE 3:** Fast-Chemistry Reaction Mechanism

**SYNTAX** REAC {FAST} {N<sub>1</sub>, N<sub>2</sub>} {N<sub>3</sub>, ..., N<sub>n</sub> | fname}

**FAST** The two-parameter PDF fast-chemistry reaction model is used. In this mode, two differential equations are solved for the mean and variance of the mixture fraction. The values of the mean mixture fraction and variance at each node are then used to determine the properties (temperature, species concentrations, and density) at the node by a lookup table procedure. The lookup table consists of the properties as a function of the mixture fraction mean and variance. The table is constructed by assuming that the mean and variance describe the PDF (usually a beta-function) of the properties at each point of the domain. The properties for each mean and variance value in the table are then determined by convoluting the corresponding PDF with the properties, which are assumed to be a function of the mixture fraction and are determined by chemical equilibrium considerations. This model is based on the approach described by Jones and Whitelaw (1982).

**fname** The name of the file containing the fast-chemistry lookup table. The total number of values in the table must equal N<sub>1</sub> times N<sub>2</sub>.

**N<sub>1</sub>** The number of rows of data in the fast-chemistry lookup table. Each row represents the properties (temperature and species concentrations) for a given value of the mean and the variance. *There is no default value.*

**N<sub>2</sub>** The number of columns of data in the fast-chemistry lookup table. The first column represents the mean, the second column the variance, and each of the others represents a property. *There is no default value*

**N<sub>3</sub>,...,N<sub>5</sub>** These represent the numerical values in the table. If the file name is specified, then these values are read from the file; otherwise N<sub>1</sub> times N<sub>2</sub> values must be specified with the command. *There is no default value.*

## EXAMPLES

---

**REA**ction FAST chemistry; lookup table has 1000 rows and 5 columns on file 'FAST\_CHEM'  
**REA**ction FAST chemistry 1000 rows & 5 columns values!!!! 5000 values must be specified)

**MODE 4:** Empirical Constants for Fast-Chemistry Reaction

**SYNTAX** REAC {FAST} {CONS} {N<sub>1</sub>, N<sub>2</sub>}

**FAST** The two-parameter PDF fast-chemistry reaction model is used.

**CONS** The empirical constants required for the fast-chemistry reaction are specified.

**N<sub>1</sub>** The constant C<sub>g1</sub> (>0) for fast-chemistry reaction (see Jones and Whitelaw, 1982). The default value is 2.

**N<sub>2</sub>** The constant C<sub>g2</sub> (>0) for fast-chemistry reaction (see Jones and Whitelaw, 1982). The default value is 2.

#### EXAMPLES

---

REACTION FAST chemistry CONStants: cg1 = 3.0

REACTION FAST chemistry CONStants: cg1 = 2.0 and cg2 = 3.0

REACTION FAST chemistry CONStants: cg1 = 2.0 and cg2 = 3.0

**MODE 5:**      **Disable Previously Specified Reaction**

**SYNTAX**      **REAC {idreac} [subrgn] {OFF}**

**idreac**      A unique identifier for the reaction rate.

**subrgn**      Same as Mode 2.

**OFF**      Any previously specified reaction with **idreac** identifier for the specified subregion is disabled.  
A new specification may follow.

#### **EXAMPLES**

---

**REAC**tion R1 for subdomain ID=R1DMn switched OFF

**REAC**tion R2 for SELEcted subdomain OFF

## 5.87 READ

**PURPOSE** To read the archive file for basic problem information and initial conditions.

**MODE 1:** Read a Generic Archive File

**SYNTAX** READ {fname} [fmt] [SKIP|ONLY= $\Phi$ ] [STAR] [N1]

**fname** The name of the file from which the input is obtained. 3.3 for more information. There is no default file name; a file name must be specified.

**fmt** The modifier "FORMATTED" or "UNFORMATTED" that define the nature of the data in the restart file. If this specification is omitted, the file is assumed to be formatted (see Section 2.10).

**SKIP** If this modifier is present then any explicitly specified variables, if present on the file, are ignored at time of reading of the file. If this modifier is present, then the **ONLY** modifier must not be present on the command.

**ONLY** If this modifier is present then only the explicitly specified variables, if present on the file, are read from the file. If this modifier is present, then the **SKIP** modifier must not be present on the command.

**$\Phi$**  One or more of the character strings listed in Table 2.8.1-3 that represent the corresponding variables on the file that is not to be read from the file. This input is ignored if the **SKIP** modifier is not present.

**STAR** If this modifier is present, the calculation step number for the current simulation is set to 0; otherwise, it is computed by reference to the step number at which the data was archived. This modifier is effective only if the file is read at the start of simulations. If simulation is already in progress, then this modifier is ignored and the active current step number of simulation is retained.

**N1** The data set number ( $\geq 0$ ) to be read from the archive file. If no data-set number is specified, then the first set is read from the archive file. As explained in Section 2.9, a data set in this context consists of several records. The default value is 1.

**REMARKS**

If the file being read was generated from a steady state simulation (where  $t=10^{30}$  by default) then the value of time read from the file is ignored. The value of time from the file is also ignored if the file is read after the computation has started (that is: a **SOLVE** command was given before the **READ** command).

However, if the file is read at the start of the computations, then the simulation time is set to the value read from the file. This allows a continuity of time from the last record read from the file. The time read from a file replaces the time used in an analysis if and only if the **READ** statement occurs before any **SOLVE** command and the file was generated via a transient analysis. Thus all file times from steady state simulations are ignored and all file times from transient analyses for **READ** commands that appear after a **SOLVE** command are ignored. Note that the **TIME** command following the **READ** command can reset the time used in an analysis.

**EXAMPLES**

**READ** from archive file 'PROBLEM.SAV'

**READ** record number 3 from file 'PROBLEM.SAV'

**READ** from 'EXAMPLE1.SAV'

**READ** record number 5 from 'EXAMPLE2.SAV' in FORMatted mode

**READ** record # 5 from 'EXAMPLE2.SAV' and STARt step count from now

**READ** record # 5 from 'EXAMPLE2.SAV' but SKIP variables U and MTYP

**MODE 2:**      **Read a Restart File**

**SYNTAX**      **READ {REST} [fname] [STAR]**

**REST**          A restart file is read in “unformatted” mode to continue the computations from a previous simulation. This file must have been generated previously by a **SAVE** command with **RESTART** modifier. If this restart option is used, then it is recommended that the input command file must be identical to the file that generated this particular restart file except for changes in, or presence of, **READ**, **SAVE**, **OUTPUT** and **SOLVE** commands. **Any other changes in, say, boundary conditions or geometry may lead to unpredictable results.**

**fname**          The name of the file from which the input is obtained. See Sections 2.4 and 3.3 for more information. **The default file name is the input file name but with the extension “\_RESTART.TMP” attached to the file name.** For example if the standard input file name is “MYCASE” or “MYCASE.INP” then the default file name will be “MYCASE\_RESTART.TMP”.

**STAR**          If this modifier is present, the calculation step number for the current simulation is set to 0; otherwise, it is computed by reference to the step number at which the data was archived.

#### **EXAMPLES**

---

**READ** REStart data from 'STAGE\_1.FIL

**READ** REStart file with default file name

**READ** REStart data from file 'STAGE\_1.FIL'; **START** step count again

## 5.88 REFERENCE

**PURPOSE** To specify the reference values of selected constants and variables.

**MODE 1:** Specify Reference Value for a Variable

**SYNTAX** REFE { $\Phi$ |TEMP} [ABSO|FREEZ] [N1]

**$\Phi$**  The symbol for the variable to which the specified input applies. The valid symbols are listed in Table 2.8.1-3

**TEMP** The reference value for temperature,  $T^*$ , is specified. The default value is 0.

**ABSO** The absolute base for the temperature scale,  $T_a$ , is specified. This modifier is effective only if the modifier **T** or **TEMP** is also simultaneously present. The default value is 273.15.

**FREEZ** The freezing point for the fluid is specified for phase change operations. The modifier is effective only if the modifier **T** or **TEMP** is also simultaneously present. Only the **PORFLOW®** Software Tool currently uses this input. The default value is 0.

**N1** The reference value of the corresponding variable specified by  $\Phi$  or another modifier.

**EXAMPLES**

REFE P is 2.5E5!!! N/m^2

REFE T = 30 deg C

REFE T = 60 F

REFE T ABSOLute base = 459.67 R

REFE ABSOLute TEMPerature base = 459.67 R

REFE FREEZing TEMPerature is = 0.0 C

**MODE 2:** Specify Absolute or Thermodynamic Pressure Mode

**SYNTAX** REFE {P} {ABS|THER}

**ABSO** In the ANSWER® Software Tool, by default for incompressible flow, the computed pressure is modified by subtracting the computed pressure at a reference node; that is (See REFE P NODE mode of command):

$$P = P_{\text{Computed}} - P_{\text{Reference\_Node}}$$

If this modifier is present then the reference node pressure is not subtracted. This mode is also automatically invoked for the compressible (COMP Command) mode is used or if P is explicitly specified through one or more BOUN commands.

**THER** By default the pressure in PORFLOW® is normalized to length units which are obtained by dividing the thermodynamic pressure with  $\rho \cdot g$ . If this modifier is present, then the pressure is assumed to be in thermodynamic units. All input, such as initial and boundary conditions, must be specified in thermodynamic units.

## EXAMPLES

---

REFERENCE P in ABSOLUTE mode

**MODE 3:** Specify Constants and Options for Pressure Related Choices

**SYNTAX** REFERENCE {P} {N1}

**P** The specified input pertains to the pressure equation.

**N1** In both ANSWER® and PORFLOW® Software Tool, the pressure is computed with reference to a datum, p\*. In the presence of this modifier, N1 is the reference datum pressure. This input can also be provided by the GAS P or GAS PRES command.

#### EXAMPLES

---

REFERENCE P = 10000 for this case.

REFERENCE P = 1

**MODE 4:** Location of Reference Pressure

**SYNTAX** REFERENCE {P} {NODE|COEF} {N1, ..., Nn}

**P** The specified input pertains to the pressure equation.

**NODE** In the ANSWER® Software Tool, for incompressible flow, the absolute value of pressure does not appear in the governing equations. Only relative pressure is computed from the continuity constraint. By default, the computed pressure is chosen to be relative to the pressure at the first inlet node or the first open node in the computational domain. This means that the computed pressure at the selected node is set equal to zero. If **NODE** modifier is present then pressure is set to zero at the grid node or element specified by **N1** through **Nn**.

For compressible mode of the ANSWER® software and for the other ACRi software packages (PORFLOW®, TIDAL®, etc.), this information is ignored unless pressure coefficient is computed and the **COEF** mode of this command is not invoked to explicitly specify the reference pressure location.

**COEF** The location of the reference value for computation of the pressure coefficient is specified. The pressure coefficient,  $C_p$ , is defined as:

$$C_p = \frac{P - P_{ref}}{0.5 \rho V^2}$$

In the equation above,  $P$  is the computed pressure,  $P_{ref}$  is the pressure at the specified location,  $\rho$  is the density and  $V$  is the fluid speed. By default the location of the reference pressure is assumed to be the same as that of the datum (see **NODE**).

**N1, ..., Nn** The location of the reference pressure in terms of grid indices for structured grid or element number for unstructured grid.

## EXAMPLES

---

REFERENCE NODE for P at (I=5, J=10, K=3)

REFERENCE location for P COEFFicient is (I=1, J=5, K=2)

**5.89 REGENERATION**

**PURPOSE** To specify regeneration rate of one species from another in the decay chain. This command is effective only for the **PORFLOW®** Software Tool.

**SYNTAX** **REGE** { $\Phi$ } { $\Psi$ } [ $V_{reg}$ ]

$\Phi$  The symbol for the species that is generated by the decay of another species (denoted by  $\Psi$ ) in the decay chain. The valid symbols are listed in Table 2.7.1.

$\Psi$  The symbol for the species that is decaying to generate the species denoted by  $\Phi$ . The valid symbols are listed in Table 2.7.1.

$V_{reg}$  The factor to convert the decay rate of  $\Psi$  to the source for the decay product  $\Phi$ . For species in mass units this number is between 0 and 1. However if the species are in user-defined units (such as curies) then this number may take on any value. **The user must ensure that the value supplied is consistent with the defined units of  $\Phi$  and  $\Psi$ .** If no value is specified, then the factor is assumed to be equal to 1.

**APPLICABILITY**

---

This command is available only for the **PORFLOW®** Software Tool.

**EXAMPLES**

---

**REGE**neration of C2 from decay of species C; converted fraction is 1.0 (total conversion)

**REGE**neration of C2 from C; conversion factor = 1000!! (C1 in kg but C2 in g units)

**REGE**neration of C2 from C; conversion factor = 60!! To account for different units of C1 and C2

## 5.90 RELAX

**PURPOSE** To specify relaxation factors for iterative solution of the matrix of equations.

**SYNTAX** **RELA** [ $\Phi=N_1, \Phi=N_2, \dots, \Phi=N_n$ ]

$\Phi$  One or more of the symbols for which the relaxation parameters are specified. The valid symbols are those listed in Table 2.7.1 plus **RHO** and **VIS**.

$N_1, N_2, \dots, N_n$   $\Theta$  is the relaxation factor as shown in the balance equation for the variable denoted by the symbol  $\Phi$ . If no value is specified, then any previous specification stays in effect. If only one value is specified, then it is applied to all specified variables. If more than 1 value is specified, then the number of values must equal the number of symbols on the command. The numerical value must be larger than 0 and less than 2. The default value is set to 0.5 for steady state flows and 1.0 for transient flows.

$$\Phi^{k+1} = \Theta \Phi^{\text{computed}} + (1 - \Theta) \Phi^k$$

Where:

$\Phi^k$  is the value at the  $k^{\text{th}}$  step (most recent previous value)

$\Phi^{\text{computed}}$  is the value computed from the numerical process at the  $k+1^{\text{st}}$  step, and

$\Phi^{k+1}$  is the value stored as the new value at the  $k+1^{\text{th}}$  step.

**REMARKS**

This command is effective in both steady state and transient modes of solution. The relaxation factor affects the convergence of the numerical solution. If the solution shows instability, a relaxation factor with a value less than unity may help obtain a stable solution. Alternatively, if the convergence rate is too slow, a value greater than unity may result in more rapid convergence. A value larger than 2 will lead to exponentially unstable growth of the solution.

In the steady state modes the relaxation factor affects the values of the variable during outer-loop step-to-step iterations. In the transient mode, a relaxation factor is used during execution of the inner loop in which multiple iterations (**N2** on **CONVERGENCE** command) are used at each time step. A more complete discussion of the role of the relaxation parameter is given in standard textbooks (for example, Varga, 1962).

**EXAMPLES**

**RELAXATION** factor for **P = 0.7**

**RELAXATION** factors: **T = 1.2, C = 0.9**

**RELAXATION** factors: **T = 0.7, C = 0.9, C2 = 0.7**

**RELAXATION** factors: **P= 0.1, P2= 0.1, T= 0.1, S = 0.5**

**RELAXATION** factor: **P, P2, T, S** is equal to **0.5** for all

**RELAXATION** factor **0.7** for all

## 5.91 RENAME

**PURPOSE** To rename the output variables listed in Table 2.8.1-3.

**SYNTAX** **RENA** { $\Phi$ =name}

$\Phi$  The current symbol for the output variable that is to be renamed.

**name** A character string that defines the new symbol and name for the variable currently denoted by the symbol,  $\Phi$ . The name must be enclosed in single or double quotes and must be a character string no longer than 64 characters. The first four characters of this name will be used as the new symbol for all subsequent references to this renamed variable.

### EXAMPLES

---

```
// Rename "U" to "XDIR Velocity of Fluid Flow" for all future reference
```

```
RENAme U = 'XDIR Velocity of fluid flow'
```

```
// Rename "T" to "TEMPERATURE (THERMODYNAMIC)"
```

```
RENAme T = 'TEMPERATURE (THERMODYNAMIC)'
```

```
// Rename "FF" to "TC99 Technetium 99 in solution"
```

```
RENAme FF = "TC99 Technetium 99 in solution"
```

**5.92 RETARDATION**

**PURPOSE** To specify the retardation coefficient for the governing differential equations. This command is effective only for the **PORFLOW®** Software Tool.

**APPLICABILITY** \_\_\_\_\_

This command is available only for the **PORFLOW®** Software Tool.

**REMARKS** \_\_\_\_\_

This command is exactly identical to the **STORAGE** command. The term “retardation” is normally applied only to the transport of chemical species whereas the term “storage coefficient” applies to any of the governing transport equations.

**EXAMPLES** \_\_\_\_\_

See the **STORAGE** command

## 5.93 SAVE

**PURPOSE** To write the archival and post-processing file for restart, plotting and archiving purposes.

**MODE 1:** Archival and Post-Processing Data in ACRi Block Format

**SYNTAX** SAVE [ $\Phi$ ] [BLOC|TABL] [ADD] [fname] [fmt] [GEOM|COMP] [DATA] [REPL|SEQU] [subrgn] [TIME] [V<sub>freq</sub>] [IMME|NOW|ONLY|OFF]

$\Phi$  **One or more** of the symbols that represent the variables for which output is desired. The valid symbols are listed in Table 2.8.1-3. A maximum of 64 symbols may be specified on a single command. The output for listed variables is produced in the order of specification. If no symbols are specified, and the **ADD** modifier is not present, then a suitable default set is automatically selected after the 1<sup>st</sup> **SOLVE** command is encountered. The default set includes the variables for which equations are solved, and some important supporting variables based on the nature of the problem.

**BLOC** The output for each variable is written in a block format. One record is written for each variable. The variables are written in the order in which they appear on the command. If the grid is structured then the variable is written in the manner of a FORTRAN DO loop (over the grid indices I, J, K). If the grid is unstructured, then the record for the variable is sequential over element numbers starting with the 1<sup>st</sup> element. This is the default mode.

**TABL** The file is written as a table of values with one column per variable. The first column contains the node number, the 2<sup>nd</sup> the material type and the next 2 or 3 columns contain the (x, y or x, y, z) coordinates. These are followed by up to 12 (for 3D) or 13 (for 2D) columns of the values of the specified variables. If more variables are specified, then supplemental tables are generated for the remaining variables in the same format.

**ADD** One **SAVE** command is activated by default. If the **ADD** modifier is present, then the specified command is added to the list of active **SAVE** commands. On the other hand, if the **ADD** modifier is not specified, then the default command is modified or replaced by user specified command.

**fname** The name of the file to which the output is directed. See Section 3.3 for additional information. The default file name is the Standard Output Unit name (see **OUTPUT** Command) but with the extension changed to "SAV" for the **BLOCK** and "\_TABLE.SAV" for the **TABLE** format file. For example, if the Standard Output Unit file name is "MYCASE" or "MYCASE.OUT", then the default file names will be "MYCASE.SAV" and "MYCASE\_TABLE.SAV", respectively.

If a file name is specified on an **SAVE** command with an **ADD** modifier, then this file is uniquely attached to a unit for output from that command and any subsequent **SAVE** command that specifies the **same file name**. The file name specification is case sensitive and the case must be consistent, otherwise some operating systems may report an error. See Section 3.3 for additional information.

Output from **all SAVE** command without an **ADD** modifier is directed to a **single common file** (with a default name as stated above). If any such command specifies a file name, then that file becomes the common output file for all commands without the **ADD** modifier. If a new file name is subsequently specified, then the previous file is closed and all subsequent output from **all SAVE** commands without an **ADD** modifier is directed to the new file.

**fmt** The modifier "**FORMATTED**" or "**UNFORMATTED**", which defines the nature of the data in the archive file. By default, the file is formatted.

**GEOM** By default the archive file contains problem geometry and grid connectivity variables plus variables specified by the user or active default variables (see  $\Phi$  above). If **GEOMETRY** modifier is present, then **only** the problem geometry and grid connectivity information is written to the file; the  $\Phi$  variables are omitted.

**COMP** If this modifier is present, then the problem geometry and grid connectivity information is not written to the file; **only** the  $\Phi$  variables are written. If both modifiers are specified, then **COMPACT** modifier takes precedence over the **GEOMETRY** modifier.

**DATA** By default standard **ACRi** file header containing information about the problem and nature of data appears at the start of the file. If **DATA** modifier is present then the file header

information is omitted.

- REPL** If multiple records are written due to the **V<sub>frq</sub>** input or the **NOW** modifier, then by default each new record is appended to the existing file. If this modifier is present, then the old records are replaced by the new record.
- SEQU** If **SEQU** modifier is present, then sequential files are written at a frequency determined by the **V<sub>frq</sub>** input and the **NOW** modifier. A 4 digit numerical index (from 0001 to 9999) is appended to the file name for each file written in the order determined by **V<sub>frq</sub>** and **NOW** input. Any extension specified with the file name is retained without change. For example if the file name is "MYDATA.SAV", then the names for sequential files, in order, will be: "MYDATA0001.SAV", "MYDATA0002.SAV", and so on. If more than 9999 files are generated, then the subsequent data will be appended at the end of the 9999<sup>th</sup> file.
- subrgn** This modifier is effective only for the **TABLE** mode of the command. It specifies the subregion for which the output is required. If no subregion is specified, then entire computational domain is selected. See Section 3.4.
- TIME** By default, **V<sub>frq</sub>** is the frequency of output in terms of number of steps. If this modifier is present, then **V<sub>frq</sub>** is interpreted to be time interval between successive outputs.
- V<sub>frq</sub>** The frequency (step or time interval) at which the output is written to the output device. See Section 3.7 for further details. The specified value is ignored if it is zero or negative. The default value is set so that output is obtained only at the end of simulations.
- IMME** The output is produced immediately. This modifier is necessary only before the 1<sup>st</sup> **SOLVE** command is given. It then enables the output of the current value of specified variable(s) immediately. After the 1<sup>st</sup> **SOLVE** command, the output is always produced as soon as the command is encountered and this modifier is redundant. The symbol(s) for the output variable(s) must be specified with this modifier since the default output symbols are not available till after the 1<sup>st</sup> **SOLVE** command.
- NOW** If this modifier is specified before the 1<sup>st</sup> **SOLVE** command, then the output is produced just before the 1<sup>st</sup> step of the solution but after all the initial and boundary conditions have been processed. Thus, this modifier provides the output of the initial conditions for the variable(s) at the start of the solution process. After the 1<sup>st</sup> **SOLVE** command. The output is always produced as soon as the command is encountered; this modifier is redundant.
- ONLY** If this modifier is specified then the output is produced only once at the step number or time specified by the **V<sub>frq</sub>**. This modifier is ignored if **NOW** or **IMMEDIATE** modifier is present.
- OFF** Output for any previous command(s) for the same **subrgn** and type (**BLOCK** or **TABLE**) is subsequently suppressed.

## REMARKS

---

If no **SAVE** command is specified then a command is automatically invoked to produce output for active variables at the end of simulations. The active variables consist of all variables for which the equations are solved plus other variables that are of interest to the problem being solved. Successive commands may be used to accommodate changing output requirements.

**Any modifiers or numerical input specified on an **SAVE** command without the **ADD** modifier stay active for all commands without the **ADD** modifier unless changed by a subsequent command.** For example, any output frequency specified on one such command will stay in effect till it is replaced by new value on a subsequent command. The **fmt**, **GEOM**, **COMP** and **DATA** modifiers can only be specified before the actual writing of the file has started.

## EXAMPLES

---

**SAVE** every 20 steps  
**SAVE** U, V, W on file 'DEMO.PLT' in UNFormatted mode  
**SAVE** U, V, P IMMEDIATEly on file 'PRESOLVE.SAV'  
**SAVE** U, V, P, K and L every 100 steps  
**SAVE** U, V, P, K and L every 100 steps by REPLacing old records  
**SAVE** U, V, P at TIME interval = 0.2 units  
**SAVE** U, T and K NOW and every 20 steps

**SAVE** ADD output of GEOMetry data to file 'GEOM.SAV'  
**SAVE** in COMPact form only on file 'VARIABLES.SAV'  
**SAVE** in COMP form with DATA file 'DATAONLY.SAV'  
**SAVE** in COMP only on file 'DATA.SAV' in SEQUential mode every 100 steps  
**SAVE OFF** for all variables  
**SAVE TABLE** of U, V, W on file 'TABLE.ARC'  
**SAVE** U, V, P, K and L in TABLE format every 100 steps  
**SAVE** in TABLE format at TIME interval of 1.25 years  
**SAVE** in TABLE format at TIME interval of 1.25 years and REPLace every time  
**SAVE TABLE** of U, V, P for subregion on most recent LOCAt command  
**SAVE** U in TABLE for subregion ID=DMNTable at TIME interval = 0.2 hours  
**SAVE TABLE** of T NOW at TIME interval of 20 years  
**SAVE OFF** for TABLE option  
**SAVE TABLES** SEQUentially every 100 steps on file 'SEQUENCE.TBL'

<b>MODE 2:</b>	<b>Output of Properties and Coefficients for the Differential Equations</b>
<b>SYNTAX</b>	<b>SAVE {Φ} {[COND DIFF] [STOR] [SOUR] [MATR] [RESI]} [BLOC TABL] [ADD] [fname] [fmt] [GEOM COMP] [DATA] [REPL SEQU] [subrgn] [TIME] [V<sub>freq</sub>] [IMME NOW ONLY OFF]</b>
<b>Φ</b>	<b>One, and only one,</b> of the symbols that denotes the dependent variable for which a differential equation is solved. The valid symbols are listed in Table 2.7.1. <b>There is no default value; a valid symbol must be specified.</b>
<b>COND</b>	The conduction or diffusion coefficient for the variable is printed to the output file. If the conduction or diffusion coefficient is a tensor, then the output is produced for each component of the tensor.
<b>DIFF</b>	Same as <b>COND</b> .
<b>STOR</b>	The storage coefficient for the variable is printed to the output file.
<b>SOUR</b>	The net source (algebraic sum of source and sink) for the variable is printed to the output file.
<b>MATR</b>	The matrix coefficients for the variable are printed to the output file. The coefficients are printed in the order of the forcing function (rhs of the matrix), the diagonal coefficient and the influence coefficients for each of the neighboring elements.
<b>RESI</b>	The residue of the matrix equation or the governing differential equation is printed to the output device. If the output is requested at an intermediate stage of computation, then the matrix residue ( $B_j - A_{ij} X_i$ ) is printed. If the output is requested at the final stage of computations, then the residue of the governing differential equation is printed.
<b>BLOC</b>	See Mode 1 specification.
<b>TABL</b>	See Mode 1 specification.
<b>ADD</b>	<b>This modifier is assumed by default.</b> Each command in this mode, unless an <b>OFF</b> modifier is present, is treated as an additional command that adds to the active commands.
<b>fname</b>	The file name to which the output is directed. Since the <b>ADD modifier is assumed by default</b> , the filename, if specified, is considered unique for the command and any subsequent <b>SAVE</b> command that specifies the <b>same file name</b> . <b>By default, the output is directed to the same file that is used for Mode 1 output of SAVE commands without the ADD modifier.</b> See Section 3.3 for additional information.
<b>fmt</b>	See Mode 1 specification.
<b>GEOM</b>	See Mode 1 specification.
<b>COMP</b>	See Mode 1 specification.
<b>DATA</b>	See Mode 1 specification.
<b>REPL</b>	See Mode 1 specification.
<b>SEQU</b>	See Mode 1 specification.
<b>subrgn</b>	See Mode 1 specification.
<b>TIME</b>	See Mode 1 specification.
<b>V<sub>freq</sub></b>	The frequency (step or time interval) at which the output is written to the output device. <b>In this mode of the command, there is no default value. A value must be specified or the modifiers NOW or IMMEDIATE must be present for an output to be obtained.</b> See <b>REMARKS</b> below.
<b>IMME</b>	The output is produced at the start of computations. In this mode of input, this modifier is equivalent to the <b>NOW</b> modifier because the coefficients and components of the transport equation are not available till the 1 <sup>st</sup> <b>SOLVE</b> command is given and the computation process has started.
<b>NOW</b>	See Mode 1 specification.
<b>ONLY</b>	See Mode 1 specification.
<b>OFF</b>	Output for any previous command(s) for the same <b>subrgn</b> , same type ( <b>BLOCK</b> or <b>TABLE</b> )

and same combination of **COND**, **DIFF**, **STOR**, **SOUR**, **MATR**, and **RESI** modifiers is subsequently suppressed.

#### REMARKS

---

This mode of the **SAVE** command provides the ability to monitor the coefficients and contribution of the various components of the transport equation. Any combination of the modifiers **COND**, **DIFF**, **STOR**, **SOUR**, **MATR**, and **RESI** may be specified. **At least one of these must be specified to trigger this mode of the command.**

This command mode triggers output for quantities which are computed (and needed) only during the solution of the equations. Most of these quantities are not allocated permanent storage but are computed on as needed basis. Therefore this output is possible only during the solution of the equation for the corresponding variable. **This command must therefore be given before the last SOLVE command.** Also a frequency (**V<sub>frq</sub>** with or without the **ONLY** modifier) or **NOW** modifier **must be specified** for the output to be generated.

**Multiple commands may be used to obtain output for different variables.** Output is generated after all the coefficients for the solution matrix have been computed. These commands essentially form a queue and are executed for each variable in order as the solution for that variable at the specified step is initiated.

**This mode of the SAVE command may be combined with Mode 3 of the command. That is, the modifiers of Mode 2 may be simultaneously specified with those of Mode 3 on the same command.**

#### EXAMPLES

---

**SAVE:** for U and P of DIFFusion coefficients at the final stage

**SAVE:** for T DIFFusion, coefficients NOW and the end to file = 'DIFFUSION.T'

**SAVE:** for T STORage and MATRix coefficients NOW (at the next step)

**SAVE:** for T CONDUction, STORage, SOURce, MATRix and RESIDUe at step number 52 ONLY

**SAVE:** for C: SOURce at step number every 52 steps

**SAVE:** in TABLE format of MATRIX for T and P at frequency of 75 steps

**SAVE:** of RESIdue for T at the end

<b>MODE 3:</b>	<b>Output of Convective, Diffusive and Total Flux for each Element</b>
<b>SYNTAX</b>	<b>SAVE</b> { $\Phi$ } { <b>FLUX</b> } [ <b>CONV</b> ] [ <b>DIFF</b> ] [ <b>TOTA</b> ] [ <b>AREA</b> ] [ <b>BLOC TABL</b> ] [ <b>ADD</b> ] [ <b>fname</b> ] [ <b>fmt</b> ] [ <b>GEOM COMP</b> ] [ <b>DATA</b> ] [ <b>REPL SEQU</b> ] [ <b>subrgn</b> ] [ <b>TIME</b> ] [ <b>V<sub>frq</sub></b> ] [ <b>IMME NOW ONLY OFF</b> ]
$\Phi$	<b>One, and only one,</b> of the symbols that denotes the dependent variable for which a differential equation is solved. The valid symbols are listed in Table 2.7.1. <b>There is no default value; a valid symbol must be specified.</b>
<b>FLUX</b>	The total flux for each face of the elements of the computational domain is printed. The total flux is the sum of the convective and diffusive components.
<b>CONV</b>	The convective flux for each face of the elements of the computational domain is printed.
<b>DIFF</b>	The diffusive flux for each face of the elements of the computational domain is printed.
<b>TOTA</b>	The sum total of the convective and diffusive fluxes for each face of the elements of the computational domain is printed. <b>This modifier is assumed by default if the <b>CONV</b> or <b>DIFF</b> modifiers are not present.</b>
<b>AREA</b>	The computed flux is divided by the projected area of the face for output.
<b>BLOC</b>	See Mode 1 specification.
<b>TABL</b>	See Mode 1 specification.
<b>ADD</b>	<b>This modifier is assumed by default.</b> Each command in this mode, unless an <b>OFF</b> modifier is present, is treated as an additional command that adds to the active commands.
<b>fname</b>	The file name to which the output is directed. Since the <b>ADD</b> modifier is assumed by default, the filename, if specified, is considered unique for the command and any subsequent <b>SAVE</b> command that specifies the <b>same file name</b> . <b>By default, the output is directed to the same file that is used for Mode 1 output of <b>SAVE</b> commands without the <b>ADD</b> modifier.</b> See Section 3.3 for additional information.
<b>fmt</b>	See Mode 1 specification.
<b>GEOM</b>	See Mode 1 specification.
<b>COMP</b>	See Mode 1 specification.
<b>DATA</b>	See Mode 1 specification.
<b>REPL</b>	See Mode 1 specification.
<b>SEQU</b>	See Mode 1 specification.
<b>subrgn</b>	See Mode 1 specification.
<b>TIME</b>	See Mode 1 specification.
<b>V<sub>frq</sub></b>	The frequency (step or time interval) at which the output is written to the output device. <b>In this mode of the command, there is no default value. A value must be specified or the modifiers <b>NOW</b> or <b>IMMEDIATE</b> must be present for an output to be obtained.</b> See <b>REMARKS</b> below.
<b>IMME</b>	The output is produced at the start of computations. In this mode of input, this modifier is equivalent to the <b>NOW</b> modifier because the coefficients and components of the transport equation are not available till the 1 <sup>st</sup> <b>SOLVE</b> command is given and the computation process has started.
<b>NOW</b>	See Mode 1 specification.
<b>ONLY</b>	See Mode 1 specification.
<b>OFF</b>	Output for any previous command(s) for the same <b>subrgn</b> , same type ( <b>BLOCK</b> or <b>TABLE</b> ) and same combination of <b>FLUX</b> , <b>CONV</b> , <b>DIFF</b> and <b>TOTAL</b> modifiers is subsequently suppressed.

**REMARKS**

---

This mode of the **SAVE** command provides the output of the convective, diffusive and total fluxes at the faces of the elements. Any combination of the choice of fluxes may be specified. **At least one of these must be specified to trigger this mode of the command.**

This command mode triggers output for quantities which are computed (and needed) only during the solution of the equations. These quantities are not allocated permanent storage but are computed on as needed basis. Therefore this output is possible only during the solution of the equation for the corresponding variable. **This command must therefore be given before the last SOLVE command.** Also a frequency (**V<sub>frq</sub>** with or without the **ONLY** modifier) or **NOW** modifier **must be specified** for the output to be generated.

**Multiple commands may be used to obtain output for different variables.** Output is generated after all the fluxes required for solution of the transport equation have been computed. These commands essentially form a queue and are executed for each variable in order as the solution for that variable at the specified step is initiated.

**This mode of the SAVE command may be combined with Mode 2 of the command. That is, the modifiers of Mode 2 may be simultaneously specified with those of Mode 3 on the same command.** Also, if the output of both the diffusion coefficients and the diffusive fluxes is desired through such a single command, then the modifier **DIFF** must be specified twice or both the **COND** and **DIFF** modifiers must appear simultaneously on such a command.

**EXAMPLES**

---

**SAVE:** FLUX for T for each element at the final stage! Output to default Mode 1 file

**SAVE:** CONVective FLUX for T NOW (at the next step) to file 'CFLUX.TMP'

**SAVE:** CONVective and DIFFusive FLUX for T at step number 50 ONLY on 'CD\_File.50'

**SAVE:** CONVective, DIFFusive and TOTAL FLUX divided by AREA for T every 50 steps

**SAVE:** FLUX divided by AREA for T at step# 52 ONLY for ID=SUBREGION

**SAVE:** in TABLE mode FLUX for T for ID=SUBREGION at end of simulations to "FLUX.TBL"

**MODE 4:** Output of Special Derived Flow Based Variables

**SYNTAX** **SAVE** **{**[GRAD] [VORT] [STRUCTURE] [STRAIN] [STRESS] [LIGHTHILL]} [BLOC|TABL] [ADD] [fname] [fmt] [GEOM|COMP] [DATA] [REPL|SEQU] [subrgn] [TIME] [Vfrq] [IMME|NOW|ONLY|OFF]

**GRAD** The tensor components of the gradients of velocity,  $\phi_{ij}$  are output, where:

$$\phi_{ij} = \frac{\partial u_i}{\partial x_j}$$

Here  $u_i$  is the  $i^{\text{th}}$  component of velocity and  $x_j$  is the  $j^{\text{th}}$  coordinate. The gradient consists of 4 components for 2D and 9 for 3D.

**VORT** Output of the components of vorticity vector,  $\omega$ , is obtained, where:

$$\omega_1 = \frac{\partial u_3}{\partial x_2} - \frac{\partial u_2}{\partial x_3}; \quad \omega_2 = \frac{\partial u_1}{\partial x_3} - \frac{\partial u_3}{\partial x_1}; \quad \omega_3 = \frac{\partial u_2}{\partial x_1} - \frac{\partial u_1}{\partial x_2}$$

There are 3 components of vorticity for 3D flow but only one for 2 D flow.

**STRUCTURE** Output of a “vorticity” structure variable,  $\Omega$ , is obtained. It is defined as:

$$\begin{aligned} \Omega &= -\frac{\partial u_1}{\partial x_2} \frac{\partial u_2}{\partial x_1} \quad \text{for 2D} \\ &= -\frac{\partial u_1}{\partial x_2} \frac{\partial u_2}{\partial x_1} - \frac{\partial u_2}{\partial x_3} \frac{\partial u_3}{\partial x_2} - \frac{\partial u_3}{\partial x_1} \frac{\partial u_1}{\partial x_3} \quad \text{for 3D} \end{aligned}$$

**STRAIN** Output of the components of strain tensor,  $S_{ij}$ , is obtained, where:

$$S_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right)$$

There are 3 components for 2D and 6 for 3D flow.

**STRESS** Output of the components of stress tensor,  $\tau_{ij}$ , is obtained, where:

$$\tau_{ij} = \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \delta_{ij} \sum_k \frac{\partial u_k}{\partial x_k} \right) - \delta_{ij} p$$

Here  $\mu$  is viscosity and  $p$  is the pressure. There are 3 components for 2D and 6 for 3D flow.

**LIGHTHILL** Output of the components of “lighthill” stress tensor,  $L_{ij}$ , is obtained, where:

$$L_{ij} = -\mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \delta_{ij} \sum_k \frac{\partial u_k}{\partial x_k} \right) + \rho u_i u_j$$

Here  $\mu$  is the viscosity and  $\rho$  is the density. There are 3 components for 2D and 6 for 3D flow.

**BLOC** See Mode 1 specification.

**TABL** See Mode 1 specification.

**ADD** This modifier is assumed by default. Each command in this mode, unless an **OFF** modifier is present, is treated as an additional command that adds to the active commands.

**fname** The file name to which the output is directed. Since the **ADD** modifier is assumed by default, the filename, if specified, is considered unique for the command and any subsequent **SAVE** command that specifies the **same file name**. By default, the output is directed to the same file that is used for Mode 1 output of **SAVE** commands without the **ADD** modifier. See Section 3.3 for additional information.

**fmt** See Mode 1 specification.

**GEOM** See Mode 1 specification.

**COMP** See Mode 1 specification.

**DATA** See Mode 1 specification.

<b>REPL</b>	See Mode 1 specification.
<b>SEQU</b>	See Mode 1 specification.
<b>subrgn</b>	See Mode 1 specification.
<b>TIME</b>	See Mode 1 specification.
<b>V<sub>frq</sub></b>	See Mode 1 specification.
<b>IMME</b>	See Mode 1 specification.
<b>NOW</b>	See Mode 1 specification.
<b>ONLY</b>	See Mode 1 specification.
<b>OFF</b>	Output for any previous command(s) for the same <b>subrgn</b> , same type ( <b>plane</b> or <b>TABLE</b> ) and same combination of <b>GRAD, VORT, STRUCTURE, STRAIN, STRESS</b> and <b>LIGHTHILL</b> modifiers is subsequently suppressed.

#### EXAMPLES

---

**SAVE:** VORTICITY and GRADients of velocity at the final stage to file 'VORTGRAD.FIL'

**SAVE:** VORTICITY STRUCTure, STRESS and LIGHTHILL stress every 100 steps to "VFLOW.FIL"

**SAVE:** in TABLE mode VORTICITY STRUCTure, STRESS and LIGHTHILL every 100 steps

**MODE 5:** Output of Partitioning for Species Variables

**SYNTAX** **SAVE** {SECO} {PHAS} {Φ} [plane|TABL] [ADD] [STAN|fname] [subrgn] [STAT|NOST]  
[NARR|WIDE] [TIME] [V<sub>frq</sub>] [IMME|NOW|ONLY|OFF]

**SECO** The output for the inventory (amount in storage at an element) of a species in phases other than the primary phase is generated. The output is computed as:

$$Q_{\Phi S} = Q_{\Phi} - Q_f \Phi$$

Here  $Q_{\Phi S}$  is the inventory of property in an element in phases other than the primary phase,  $Q_{\Phi}$  is the total inventory and  $\Phi$  is the species concentration in the primary phase for the element.

**PHAS** A companion modifier that must be present to invoke this mode of the command.

**Φ** **One or more** of the symbols that represent the variables for which output is desired. The valid symbols are the species symbols listed in Table 2.7.1. **A maximum of 32 symbols may be specified on a single command.** This output can only be obtained for species variables for which a transport equation is solved. **There is no default value; a valid symbol must be specified.**

**plane** See Mode 1 specification.

**TABL** See Mode 1 specification.

**ADD** **This modifier is assumed by default.** Each command in this mode, unless an **OFF** modifier is present, is treated as an additional command that adds to the active commands.

**STAN** See Mode 1 specification.

**fname** See Mode 2 specification.

**plane** See Mode 1 specification.

**subrgn** See Mode 1 specification.

**NARR** See Mode 1 specification.

**WIDE** See Mode 1 specification.

**TIME** See Mode 1 specification.

**V<sub>frq</sub>** See Mode 1 specification.

**IMME** See Mode 1 specification.

**NOW** See Mode 1 specification.

**ONLY** See Mode 1 specification.

**OFF** Output for any previous command(s) for the same **subrgn**, same type (**plane** or **TABLE**) and same combination variable is subsequently suppressed.

**REMARKS**

This mode of the command is currently active only for the **PORFLOW®** Software Tool where some of the property may be present in secondary phases (solid and gas components of the porous matrix). The partitioning between different phases is specified by the **TRANSPORT**, **DISTRIBUTION** or **RETARDATION** commands.

The output from this mode of the command can be generated **only after** the start of the solution process. At the initial stage, before the solution process starts, the meaningful values of the inventory of variables may not be available. **This mode of the command operates independently of any other modes of the command and essentially adds to any other commands that are specified.**

**EXAMPLES**

**SAVE:** SECOndary PHASe partitioning for variables C and C2.to file 'SOLIDC.FIL'

**SAVE:** SECO PHAS for C and C2.to file 'SOLIDC.FIL' every 100 steps

**MODE 5:** Output of Running Average Value of Variables

**SYNTAX** SAVE {AVER} {Φ} {fname} {V<sub>freq</sub>} [V<sub>Start</sub>] [V<sub>Stop</sub>] [UNFO] [DIAG]

- AVER** The average of a variable is written to the file. For steady state this average is the simple mean whereas for transient simulations a time integral is computed. The file is written in the block format of the **SAVE** command but without any geometry or header information.
- Φ** From 1 to 10 symbols that represent the variables for which output is desired. The valid symbols are listed in Table 2.8.1-3. Multiple commands may be specified if output is desired for more than 10 variables.
- fname** The name of the file to which the output is directed. See Section 3.3 for additional information. A file name must be specified; there is no default file name
- V<sub>freq</sub>** The step or time interval for which the running average is computed. The value is interpreted to be in time units if transient mode of solution is active otherwise it is interpreted to be the step interval. There is no default value; a value must be specified..
- V<sub>Start</sub>** The starting time (for transient simulation) or step number (for steady state simulation) at which the average is started.. By default the output starts with the 1<sup>st</sup> computational step.
- V<sub>Stop</sub>** The starting time (for transient simulation) or step number (for steady state simulation) at which the average is stopped.. By default the output continues to the end of the computations.
- UNFO** By default the file is generated in the formatted mode. If this modifier is present, then the file is in unformatted mode.
- DIAG** If this modifier is present, then the a line of diagnostic information consisting output record number, step number and time of output, is written to the Standard Output Unit.

## EXAMPLES

---

**SAVE AVERAGE** of **U** and **T** on "PROB\_AVG.FIL" for an interval of **10** steps

**SAVE AVERAGE** of **T** on "PROB\_AVG\_T.FIL" at time interval of **1.5**; from **5.7** to **8.5** with **DIAG**nostic

**MODE 7:** Restart File for Later Continuation of Simulations

**SYNTAX** SAVE {REST} [fname] [TIME] [V<sub>frq</sub>] [NOW]

**REST** A special file is generated to restart the computations at a later time. It is always in machine-specific “unformatted” mode. This file is generated in addition to other archive files that may be generated by other **SAVE** commands.

This file can be read only by a **READ** command with **RESTART** modifier to restart the computations for the same identical problem. Only the **SOLVE** command may be modified for the new simulations; all other input data must be identical to the run that generated the file. **RESTART** file should not be used to start a different problem or to restart the same problem with different input commands. In these latter cases, the required variables should be explicitly saved by Mode 1 of the **SAVE** command. If multiple commands are given, and a new file name is not specified, then any previous data on the file is replaced by the new data.

**fname** The name of the file to which the output is directed. See Section 3.3 for additional information. The default file name is the output file name (see **OUTPUT** Command) but with the extension “\_RESTART.TMP” attached to the file name. For example if the standard output file name is “MYCASE” or “MYCASE.OUT”, then the default file name will be “MYCASE\_RESTART.TMP”.

**TIME** By default, the **V<sub>frq</sub>** is interpreted to be the frequency in terms of number of steps. If this modifier is present, then **V<sub>frq</sub>** is interpreted to be time interval between successive outputs.

**V<sub>frq</sub>** The frequency (step or time interval) at which the restart file specified by **'fname'** is written. See Section 3.7 for further details. If no value is specified and if **NOW** modifier is not present, then the output is obtained only at the end of simulations. If a value (> 0) is specified, then the file is written at the specified frequency and also at the end of simulations. The restart file always contains only one data record. The data on the file is replaced each time the file is written.

**NOW** A restart file is immediately written. If **V<sub>frq</sub>** modifier is not present, then the file is written only once, otherwise the file is replaced at the specified frequency.

## EXAMPLES

---

### SAVE REST

**SAVE REST**art file! xxx\_RESTART.TMP file generated at end of simulations; xxx=output file without extension

**SAVE REST**art file as 'STAGE\_1.FILE'! File STAGE\_1.FILE will be generated at end of simulations

**SAVE REST**art NOW! xxx\_RESTART.TMP file will be immediately generated.

**SAVE REST**art every 10 steps on file named 'REST.FIL'! Above command will generate REST.FIL every 10 steps and also at the end of simulations.

**SAVE REST**art NOW and replace every 10 steps on file named 'REST.FIL'! Above command will generate REST.FIL file immediately, and then replace it every 10 steps and also at the end of simulations.

**5.94 SCALE**

**PURPOSE** To allow internal scaling of the specified input according to the equation:

$$Q_{in} = a_1 * Q + a_2$$

where  $Q_{in}$  is the internal representation of a quantity,  $Q$  is the value specified by the user, and  $a_1$  and  $a_2$  are user-specified constants.

**SYNTAX** **SCAL** {**N1**} [**N2**]

**N1** Multiplier,  $a_1$ , in the scaling equation automatically set to 1 at the end of each application. The default value is 1.

**N2** Addend,  $a_2$ , in the scaling equation automatically set to 0 at the end of each application. The default value is 0.

**REMARKS**

---

This command must be used immediately before the command that contains the data to be scaled. It can be used for internal scaling in conjunction with some of the modes of the **COORDINATE**, **LOCATE**, **SET** and **SOURCE** commands. See those commands for further details.

**EXAMPLES**

---

**SCALE** multiply by 3.3 and add 10.

**SCALE** multiply by 0.3048

**5.95 SCHMIDT NUMBER**

**PURPOSE** To specify the Schmidt number for the fluid. This command is effective only for the ANSWER® Software Tool.

**SYNTAX** **SCHM** [**EFFE**] [ $\Phi=V_1, \Phi=V_2, \dots, \Phi=V_n$ ]

**EFFE** If the modifier **EFFECTIVE** is present then the input is assumed to be for the effective Schmidt number; otherwise it is assumed to be the molecular Schmidt number for the fluid.

$\Phi$  One or more symbols to denote the variables for which the Schmidt numbers are specified. The valid symbols are those listed in Table 2.7.1 with the exception of the U, V, W, P and HT. If no symbol is specified, then the input is applied to all the mass transfer variables.

$V_1, V_2, \dots, V_n$  Schmidt numbers (>0) for the dependent variables denoted by the symbols on the command. If no value is specified, then any previous specification stays in effect. If only 1 value is specified, then it is applied to all specified variables. If more than 1 value is specified, then the number of values must equal the number of symbols on the command. The default value is 0.7.

**COMMETNS**

---

This input is only effective for the ANSWER® Software Tool.

**EXAMPLES**

---

**SCHM**id number for fluid = 1.

**SCHM**id number for FF = 0.5, FU = 0.7, CH = 0.7

**SCHM**id number **EFFE**ctive value for FF = 0.5

**5.96 SCREEN**

**PURPOSE** To control the echo of diagnostic output obtained from the **DIAGNOSTIC** command to the CRT device.

**SYNTAX** **SCRE** [**OFF** | **ON**] [**WIDE**]

**OFF** The diagnostic output to the CRT device is turned off.

**ON** The diagnostic output to the CRT device is turned on. This is the default option. This option may also be used to turn on the CRT output, which was previously turned off.

**WIDE** By default a maximum of 5 selected diagnostic variables (**DIAGNOSTIC** command) are echoed to the CRT device. If this modifier is present, then the maximum is set to 9.

**REMARKS**

---

This command allows the user to monitor progress of the numerical simulation on the CRT device. It does not affect the diagnostic output to the standard output device.

**EXAMPLES**

---

**SCRE**en echo for diagnostic output to be turned OFF

**SCRE**en ON! This is the default option

**SCRE**en in WIDE format

**SCRE**en ON in WIDE format! Turn on previously off screen output

**5.97 SELECT**

**PURPOSE** To locate and identify a subregion or material type in the domain of computation for later input by one of the other commands.

**REMARKS** \_\_\_\_\_

This command is identical in all respects to the **LOCATE** command.

**EXAMPLES** \_\_\_\_\_

See **LOCATE** command

## 5.98 SET

**PURPOSE** To set the value of a field variable as a function of space, time or other variables.

**MODE 1:** Specification of Field Variable

**SYNTAX** SET { $\Phi$ } [func[ $\xi$ ]] { $N_1, N_2, \dots, N_n$ |fname} [operation] [mod] [ALWA] [subrgn] [STAC]

**$\Phi$**  One, and only one, symbol that denotes the variable for which the values are specified. The valid symbols include those listed in Table 2.8.1-3 plus the user defined variables (ALLOCATE command) and the real type variables defined in the initialization file (xxxINIT.ACR). There is no default value; a symbol must be specified.

**func** One of the modifiers listed in Table 4.2.1, which denotes the functional form of the dependent variable. If no function is specified, the value is assumed to be constant.

**$\xi$**  One of the independent variables listed in Table 4.2.2. If no variable is specified, the independent variable is assumed to be time.

**$N_1, \dots, N_n$**  The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.

**fname** The name of the file containing the numerical values  $N_2$  through  $N_n$ . This option is available only for selected functions. See Section 3.3 for additional information.

**operation** Available modifiers are REPLACE, ADD, SUBTRACT, MULTIPLY, DIVIDE, ABS, ABSOLUTE, POSITIVE and NEGATIVE. See Section 4.3 for further details

**ALWA** By default the SET command is implemented immediately and only once – as soon as the command is encountered. If this modifier is present then the command is executed immediately as well as repeatedly at the beginning of each time step (or iterative step in steady state mode) of the solution procedure.

**subrgn** The subregion for which the input is specified. If no subregion is specified, then entire computational domain is selected. Subregion may be specified with the FIEL or dir modifiers. See Section 3.4 for further details.

**STAC** By default  $\Phi$  is computed from the value of  $\xi$  at the same location. If this modifier is present, then  $\Phi$  is computed from  $\xi$  at a location previously specified by the STACK LOCATE command.

**REMARKS**

The primary purpose of the SET command is to specify the value of a dependent variable. Combinations of ALLOCATE and SET commands can be used to define complex multi-variate functions by specifying the appropriate operation to be performed during each intervening step.

**It should be noted** that if the SET command sets the value of a variable that is governed by a transport equation then, at the next step, the value will be recomputed from the governing equation. If it is intended to set this variable permanently then the SET command must be followed by a FIX command. The actual process of setting such values is as follows. At the start of a step, before any transport equation is solved, all values that are specified by SET commands are computed. It is to be noted that between computing the value from a SET command and the actual solution of the equation, if the value of the specific variable is required by another equation or another variable or another process (such as boundary conditions and source) then the value as computed by SET command will be used till the value is recomputed from the governing transport equation. A FIX command ensures that the value does not change during solution of the transport equation.

If the subrgn is specified with a dir modifier (see Section 3.4) then the SET command may be effectively used as an alternative to the BOUNDARY command.

**EXAMPLES**

Generic examples for this command are given in Section 4.4. The keyword SET must replace the keyword used in these examples.

**MODE 2:** Specification of a Variable on a Node by Node Basis.

**SYNTAX** SET { $\Phi$ } {**NODE|STRU|UNST|fname**} [**V<sub>1,..V<sub>m</sub></sub>**] [**subrgn**] [**EXTR**]

**$\Phi$**  A symbol to denote the variable for which the values are specified. The valid symbols include those listed in Table 2.8.1-3 plus the supplementary integer or real type field variables that are defined in the initialization file (xxxINIT.ACR). A symbol must be specified.

**NODE** The input is specified in a node-by-node manner. By default it is assumed that the input values are given in the **STRUCTURED** mode if the **subrgn** was defined with a **LOCATE** command in terms of grid indices or in the **UNSTRUCTURED** mode otherwise.

**STRU** The input is read in the manner of nested implied FORTRAN DO loops:

Read (NUNIT, \*) (((VAR (I, J, K), I = ILO, IHI), J = JLO, JHI), K = KLO, KHI)

Where **NUNIT** is an internally assigned unit number, **VAR** is the variable denoted by  **$\Phi$** , and **ILO, IHI, JLO, JHI, KLO, KHI** define, respectively, the starting and ending grid index values for the **subrgn** for a structured grid. For 2D grid, **KLO** and **KHI** are set to unity.

**UNST** The input is read in the manner of a simple implied FORTRAN DO loop:

Read (NUNIT, \*) (VAR (M), M = MLO, MHI)

Where **NUNIT** is an internally assigned unit number, **VAR** is the variable denoted by  **$\Phi$** , and **MLO and MHI** are the starting and ending element numbers for the **subrgn**. The order of the elements is the same as the one specified (or implied) by the corresponding **LOCATE** command which can be examined by including a file name on the **LOCATE** command.

**fname** The name of the file for numerical values unless the values are directly specified by **V<sub>1....V<sub>m</sub></sub>** below. See Section 3.3 for additional information.

**V<sub>1....V<sub>m</sub></sub>** The values of the variable at each node selected by the **subrgn** and **dir** modifiers. These values must be present unless **fname** is present. In this case, one of the modifiers, **NODE, STRU, or UNST** must also be specified.

**subrgn** The subregion for which the input is specified. If no subregion is specified, then entire computational domain is selected. This input is ignored for the supplementary variables. For these, the number of values specified on the file must match the total number of elements in the selected variable.

**EXTR** If this modifier is present, then it is assumed that the numerical values are only supplied for the internal field nodes. If the subdomain (**subrgn**) defined is such that it contains any exterior boundary nodes, then the boundary nodes are put equal to the field values.

**FIEL** See Mode 1 specification.

**dir** See Mode 1 specification.

## REMARKS

---

It is highly recommended that the user examine the input data (for example by running the data set with **SOLVE OFF** command) to make sure that the input has been properly interpreted.

## EXAMPLES

---

**SET T** from file 'ALLVAL' input for the entire domain of computation

**SET T** for SELEcted region from 'ACTIVAL' input for active subregion

**SET T** in ID=RGN1 region 'RGN1VAL' input for identified subregion

**SET FC** – a supplementary flux variable for the entire field from file: 'VALUES.FC'

**SET MTYP** – a supplementary material type index (integer) for the entire field from file: 'VALUES.IZ'

**MODE 3: Specification of a Variable on a Node by Node Basis as a Function of Time****SYNTAX** SET { $\Phi$ } {NODE|STRU|UNST} {TIME} {fname} [LINE|STEP] [subrgn] [FIEL] [dir]

**$\Phi$**  A symbol to denote the variable for which the values are specified. The valid symbols include those listed in Table 2.8.1-3 plus the supplementary integer or real type field variables that are defined in the initialization file (xxxINIT.ACR). A symbol must be specified.

**NODE** The input is specified in a node-by-node manner. By default it is assumed that the input values are given in the **STRUCTURED** mode if the **subrgn** was defined with a **LOCATE** command in terms of grid indices or in the **UNSTRUCTURED** mode otherwise.

**STRU** The input is read in the manner of nested implied FORTRAN DO loops:

Read (NUNIT, \*) ((VAR (I, J, K), I = ILO, IHI), J = JLO, JHI), K = KLO, KHI)

Where **NUNIT** is an internally assigned unit number, **VAR** is the variable denoted by  **$\Phi$** , and **ILO, IHI, JLO, JHI, KLO, KHI** define, respectively, the starting and ending grid index values for the **subrgn** for a structured grid. For 2D grid, **KLO** and **KHI** are set to unity.

**UNST** The input is read in the manner of a simple implied FORTRAN DO loop:

Read (NUNIT, \*) (VAR (M), M = MLO, MHI)

Where **NUNIT** is an internally assigned unit number, **VAR** is the variable denoted by  **$\Phi$** , and **MLO and MHI** are the starting and ending element numbers for the **subrgn**. The order of the elements is the same as the one specified (or implied) by the corresponding **LOCATE** command which can be examined by including a file name on the **LOCATE** command.

**fname** The name of the file for numerical values. See Section A.1.1 for file format. A file name must be specified; the input cannot be read directly from the command. The number of values for each record at each time step must equal the number of elements for the subdomain; any extra values present in the record are ignored.

**LINE** The time step interpolation is linear; this is the default option.

**STEP** The time step interpolation is step function (Equation 4.1.4).

**subrgn** The subregion for which the input is specified. If no subregion is specified, then entire computational domain is selected. This input is ignored for the supplementary variables. For these, the number of values specified on the file must match the total number of elements in the selected variable.

**FIEL** See Mode 1 specification.

**dir** See Mode 1 specification.

**REMARKS**

It is highly recommended that the user examine the input data (for example by running the data set with **SOLVE OFF** command) to make sure that the input has been properly interpreted.

**EXAMPLES**

**SET T** from file 'ALLVAL' input for the entire domain of computation

**SET T** for SELEcted region from 'ACTIVAL' input for active subregion

**SET T** in ID=RGN1 region 'RGN1VAL' input for identified subregion

**SET FC** – a supplementary flux variable for the entire field from file: 'VALUES.FC'

**SET MTYP** – a supplementary material type index (integer) for the entire field from file: 'VALUES.IZ'

**MODE 4:** Node by Node Input for a Variable Every Step from a Sequential File

**SYNTAX** SET { $\Phi$ } {**NODE|STRU|UNST**} {**SEQU**} {**fname**} [**subrgn**] [**FIEL**] [**dir**]

**$\Phi$**  A symbol to denote the variable for which the values are specified. The valid symbols include those listed in Table 2.8.1-3 plus the supplementary integer or real type field variables that are defined in the initialization file (xxxINIT.ACR). A symbol must be specified.

**NODE** The input is specified in a node-by-node manner. By default it is assumed that the input values are given in the **STRUCTURED** mode if the **subrgn** was defined with a **LOCATE** command in terms of grid indices or in the **UNSTRUCTURED** mode otherwise.

**STRU** The input is read in the manner of nested implied FORTRAN DO loops:

Read (NUNIT, \*) (((VAR (I, J, K), I = ILO, IHI), J = JLO, JHI), K = KLO, KHI)

Where **NUNIT** is an internally assigned unit number, **VAR** is the variable denoted by  **$\Phi$** , and **ILO, IHI, JLO, JHI, KLO, KHI** define, respectively, the starting and ending grid index values for the **subrgn** for a structured grid. For 2D grid, **KLO** and **KHI** are set to unity.

**UNST** The input is read in the manner of a simple implied FORTRAN DO loop:

Read (NUNIT, \*) (VAR (M), M = MLO, MHI)

Where **NUNIT** is an internally assigned unit number, **VAR** is the variable denoted by  **$\Phi$** , and **MLO and MHI** are the starting and ending element numbers for the **subrgn**. The order of the elements is the same as the one specified (or implied) by the corresponding **LOCATE** command which can be examined by including a file name on the **LOCATE** command.

**SEQU** The input is in a sequential time step interpolation is linear; this is the default option.

**fname** The name of the file for numerical values. See Section A.1.1 for file format. A file name must be specified; the input cannot be read directly from the command. The number of values for each record at each time step must equal the number of elements for the subdomain; any extra values present in the record are ignored.

**LINE** The time step interpolation is linear; this is the default option.

**STEP** The time step interpolation is step function (Equation 4.1.4).

**subrgn** The subregion for which the input is specified. If no subregion is specified, then entire computational domain is selected. This input is ignored for the supplementary variables. For these, the number of values specified on the file must match the total number of elements in the selected variable.

**FIEL** See Mode 1 specification.

**dir** See Mode 1 specification.

#### REMARKS

---

It is highly recommended that the user examine the input data (for example by running the data set with **SOLVE OFF** command) to make sure that the input has been properly interpreted.

#### EXAMPLES

---

**SET T** from file 'ALLVAL' input for the entire domain of computation

**SET T** for SELEcted region from 'ACTIVAL' input for active subregion

**SET T** in ID=RGN1 region 'RGN1VAL' input for identified subregion

**SET FC** – a supplementary flux variable for the entire field from file: 'VALUES.FC'

**SET MTYP** – a supplementary material type index (integer) for the entire field from file: 'VALUES.IZ'

**MODE 5: Specification of Multiple Variables on a Node-by-Node Basis from a Table**

**SYNTAX**     **SET**     $\{\Phi_1, \dots, \Phi_n\}$  {**NODE|STRU|UNST** | **fname**}  $\{V_1, \dots, V_{nm}\}$  [**subrgn**] [**FIEL**] [**dir**]

$\Phi_1, \dots, \Phi_n$      Symbols to denote the variables that are to be set by the command. The symbols include all those listed in Table 2.8.1-3 plus the user defined variables (**ALLOCATE** command) and the real type variables defined in the initialization file. **In addition a special symbol “SKIP” can be specified to by-pass the input of any of the columns.** Any symbol may occur multiple times. There is no default value appropriate symbols must be specified.

**NODE**            The  $\Phi$  variables are set individually for each of the nodes of a subregion. It is assumed that the input is in a tabular format with **n** columns (1 for each variable) and **m** rows (1 for each node of the subregion). By default it is assumed that the input values are given in the **STRUCTURED** mode if the **subrgn** was defined with a **LOCATE** command in terms of grid indices or in the **UNSTRUCTURED** mode otherwise.

**STRU**            The input is read in the manner of nested implied FORTRAN DO loops:

**Read (NUNIT, \*) (((VAR (L, I, J, K), L=1,n), I=ILO, IHI), J=JLO, JHI), K=KLO, KHI)**

Where **NUNIT** is an internally assigned unit number, **VAR** is the set of variables denoted by  $\Phi$ 's, **n** denotes the number of variables, and **ILO, IHI, JLO, JHI, KLO, KHI** define, respectively, the starting and ending grid index values for the **subrgn** for a structured grid. For 2D grid, **KLO** and **KHI** are set to unity.

**UNST**            The input is read in the manner of a simple implied FORTRAN DO loop:

**Read (NUNIT, \*) (VAR (L, M), L=1,n), M = MLO, MHI)**

Where **NUNIT** is an internally assigned unit number, **VAR** is the set of variables denoted by  $\Phi$ 's, **n** denotes the number of variables, and **MLO** and **MHI** are the starting and ending element numbers for the **subrgn**. The order of the elements is the same as the one specified (or implied) by the corresponding **LOCATE** command which can be examined by including a file name on the **LOCATE** command.

**fname**            The name of the file (see Section 3.3) from which  $V_1$  through  $V_{nm}$  are read.

$V_1, \dots, V_{nm}$     The table of values that pertains to the variable values at each node selected by the **subrgn** and **dir** modifiers. If **n** symbols are specified on the command and there are **m** nodes in the subregion, then **n** times **m** values must be specified in **n** columns and **m** rows.

**subrgn**           See Mode 1 specification.

**FIEL**            See Mode 1 specification.

**dir**             See Mode 1 specification.

**EXAMPLES**

**SET** by **NODE** following variables for ID=SUBRGN1! There are 3 nodes in this region)

U	V	T
10	-5	100
-10	+5	150
-10	+10	200

**SET** by **NODE** following variables for ID=SUBRGN1! There are 3 nodes in this region)

SKIP	SKIP	U	V	SKIP	T
0.2	0.2	10	-5	10.20	100
3.5	0.6	-10	+5	12.50	150
10	1.0	-10	+10	2.00	200

**SET** by **NODE** for ID=MIDDLE at X- boundary! 3 boundary nodes

(SKIP, SKIP, SKIP)	U	V	T
(0.2, 0.2, 0.2)	10	-5	100
(3.5, 0.6, 1.0)	-10	+5	150
(10., 1.0, 2.0)	-10	+10	200

**SET** by **NODE** ID=MIDDLE T, U, V SKIP from file 'SPATIAL!' File with 4 x #of nodes values

**MODE 6:** Specification of a Variable as a Linear Sum of other Variables

**SYNTAX**    **SET**    { $\Phi$ } {**SUM**|**LINE**} [**MASS**] { $\xi_1, \dots, \xi_n$ } [ $a_1, \dots, a_n$ ] [ $a_0$ ] [**option**] [**mod**] [**ALWA**] [**subrgn**] [**FIEL**]

$\Phi$             See Mode 1 specification.

**SUM**            The variable  $\Phi$  is computed from:

$$\Phi = \sum_n \xi_n$$

**LINE**            The variable  $\Phi$  is computed from:

$$\Phi = \sum_n a_n \xi_n + a_0$$

**MASS**            By default, the  $\xi$ 's (see definition below) on the right hand side of the defining equation are assumed to be the field variables. If this modifier is present, then  $\xi$ 's are replaced by the total mass of the corresponding  $\xi$  variable. The total mass is equal to the quantity that appears in the accumulation term of the governing differential equation for  $\xi$  (see Chapter 2) multiplied by the volume of the element. In the presence of this modifier only those  $\xi$ 's can appear on the right hand side for which differential equations are solved.

$\xi_1, \dots, \xi_n$     Symbols to denote the variables, or their masses, which contribute to the linear sum as in the above equation. The valid symbols include those listed in Table 2.8.1-3 plus the variables defined by the user (**ALLOCATE** command) or the real type variables defined in the initialization file (xxxINIT.ACR). There is no default value; appropriate symbols must be specified. No more than 10 symbols can be specified.

$a_1, \dots, a_n$     The numerical constants and coefficients for the linear function. These values must be specified if the **LINEAR** modifier is present. With the **SUM** modifier these values must not be specified. There is no default value; appropriate number of values must be specified.

$a_0$               The datum for the linear function if **LINEAR** modifier is specified. The default value is 0.

**option**           See Mode 1 specification.

**mod**              See Mode 1 specification.

**ALWA**            See Mode 1 specification.

**subrgn**           See Mode 1 specification.

**FIEL**            See Mode 1 specification.

**EXAMPLES**

**SET** T as SUM of X + Y + U

**SET** T as SUM of X + Y + U ALWAYS for ID=T1DOMAIN

**SET** T as LINEAR function 1. \* X + 2. \* Y -0.5 \* U + 5. ALWAYS for ID=T1DOMAIN

**MODE 7:** Specification of a Variable as a Square or Square Root Sum of other Variables

**SYNTAX** SET {Φ} {SQUA|ROOT} {ξ<sub>1</sub>, ..., ξ<sub>n</sub>} [a<sub>1</sub>, ..., a<sub>n</sub>] [a<sub>0</sub>] [option] [mod] [ALWA] [subrgn] [FIEL]

**Φ** See Mode 1 specification.

**SQUA** The variable Φ is computed from:

$$\Phi = \sum_n \xi_n^2 \quad \text{or} \quad \Phi = \sum_n a_n \xi_n^2 + a_0$$

**ROOT** The variable Φ is computed from:

$$\Phi = \sqrt{\sum_n \xi_n^2} \quad \text{or} \quad \Phi = \sqrt{\max\left[\left(\sum_n a_n \xi_n^2 + a_0\right), 0\right]}$$

**ξ<sub>1</sub>, ..., ξ<sub>n</sub>** Symbols to denote the variables that contribute to the linear sum as in the above equation. The valid symbols include those listed in Table 2.8.1-3 plus the variables defined by the user (ALLOCATE command) or the real type variables defined in the initialization file (xxxINIT.ACR). There is no default value; appropriate symbols must be specified. No more than 10 symbols can be specified.

**a<sub>1</sub>, ..., a<sub>n</sub>** The numerical constants and coefficients for the function. Either all coefficients must be omitted (in which case the function without coefficients is used) or all must be specified.

**a<sub>0</sub>** The datum for the function. The default value is 0.

**option** See Mode 1 specification.

**mod** See Mode 1 specification.

**ALWA** See Mode 1 specification.

**subrgn** See Mode 1 specification.

**FIEL** See Mode 1 specification.

## EXAMPLES

SET T as SQUARE sum of X, Y and Z

SET T as ROOT of sum of SQUAREs of X, Y and U ALWAYS for ID=T1DOMAIN

SET T as SQUARE sum of 1. \* X -1 \* Y and 5. \* Z

SET T as ROOT of sum of SQUAREs of 1. \* X, 2. \* Y and 3 \* U + 4. ALWAYS for ID=T1DOMAIN

**MODE 8:** Specification of Variables as Distance-based Interpolated Function from a Table

**SYNTAX** SET {DIST} [LINE|SQUA|CUBE|FOUR|EIGH] {Φ<sub>1</sub>, ..., Φ<sub>n</sub>} {Nsets} {V<sub>1</sub>,...V<sub>m</sub>|fname}  
 [DOMA=V<sub>domain</sub>|NEAR=V<sub>near</sub>] [option] [mod] [ALWA] [subrgn] [FIEL]

**DIST** The Φ<sub>n</sub> variables are computed from a inverse distance based interpolation function (here ξ<sub>i</sub> is i<sup>th</sup> value in the specified table for the corresponding variable, x<sub>k,i</sub> are its coordinates and N is the power to which the distance is raised):

$$\Phi = \sum_i \frac{\xi_i / r_i^N}{1 / r_i^N} ; \quad r_i^2 = \sum_k (x_k - x_{k,i})^2$$

**LINE** The inverse distance based interpolation is used (N=1).

**SQUA** The inverse distance squared interpolation is used (N=2). This is the default option.

**CUBE** Interpolation is based on distance raised to the 3<sup>rd</sup> power (N=3).

**FOUR** Interpolation is based on distance raised to the 4<sup>th</sup> power (N=4).

**EIGH** Interpolation is based on distance raised to the 8<sup>th</sup> power (N=8).

**Φ<sub>1</sub>, ..., Φ<sub>n</sub>** Symbols to denote the variables that are to be computed from the distance-based interpolation. There is no default value; appropriate symbols must be specified. No more than 7 symbols can be specified.

**Nsets** The number of sets of values in the table. Each set must consist of (x, y; for 2D) or (x, y, z; for 3D) and as many values as the symbols specified above. There is no default value.

**V<sub>1</sub>,...V<sub>m</sub>** The table of values that pertains to the coordinates of the locations and the variable values at those locations. If the table is viewed to consist of a number of columns and rows, then each row must contain, in order, (x, y; for 2D) or (x, y, z; for 3D) and n values for Φ<sub>1</sub> through Φ<sub>n</sub>. The number of rows of the table must equal Nsets.

**fname** The name of the file (see Section 3.3) from which V<sub>1</sub>,...V<sub>m</sub> are read.

**DOMA** The interpolation is restricted to the specified ξ<sub>i</sub> that lie within the distance specified by V<sub>domain</sub>. In the formula above any values where r<sub>i</sub> > V<sub>domain</sub> are excluded from the formula. If no values within the specified domain are found, then the nearest value is selected.

**V<sub>domain</sub>** The maximum distance that specifies the extent of the domain of influence in the interpolation formula. V<sub>domain</sub> must be the last numerical value on the command.

**NEAR** The interpolation is restricted to the nearest V<sub>near</sub> number of ξ<sub>i</sub> determined by r<sub>i</sub>. If this modifier is present and V<sub>near</sub> is omitted or is < 1, then the nearest specified ξ<sub>i</sub> is selected.

**V<sub>near</sub>** The number of nearest values to be included in the above interpolation formula. Default value is 1. V<sub>near</sub> must be the last numerical value on the command.

**option** See Mode 1 specification.

**mod** See Mode 1 specification.

**ALWA** See Mode 1 specification.

**subrgn** See Mode 1 specification.

**FIEL** See Mode 1 specification.

**EXAMPLES**

**SET** by DISTANCE interpolation from specified values at 3 stations (Two-dimensional)

(X, Y)	U	V	T
(0.2, 0.2)	10	-5	100
(3.5, 0.6)	-10	+5	150
(10, 1.0)	-10	+10	200

**SET** by LINEAR DISTANCE interpolation for ID=MIDDLE from values at 3 stations (Three-dimensional)

(X, Y, Z)	U	V	T
(0.2, 0.2, 0.2)	10	-5	100

(3.5, 0.6, 1.0)	-10	+5	150	
(10, 1.0, 2.0)	-10	+10	200	

**SET** by DISTANCE interpolation for ID=MIDDLE from values at 3 stations (Three-dimensional)

(X, Y, Z)	U	V	T	
(0.2, 0.2, 0.2)	10	-5	100	
(3.5, 0.6, 1.0)	-10	+5	150	
(10, 1.0, 2.0)	-10	+10	200	DOMAIN of influence=0.5

**SET** by DISTANCE interpolation for ID=MIDDLE from 5 values

(X, Y, Z)	U	V	T	
(0.2, 0.2, 0.2)	10	-5	100	
(3.5, 0.6, 1.0)	-10	+5	150	
(10, 1.0, 2.0)	-15	+10	200	
(10, 2.0, 2.0)	-20	+15	250	
(10, 5.0, 2.0)	-10	+10	300	interpolate from NEAREST 3

**SET** by DISTANCE interpolation for ID=MIDDLE from 5 sets for U V T on file='INTER.FIL', NEAR=3  
**SET** by DISTANCE ID=MIDDLE 3 sets of U V T file 'SPATIAL' ALWAYS ADD POSITIVE to existing

**MODE 9:** Specification of Values from Total Inventory

**SYNTAX** SET {Φ} {INVENT} [subrgn] {Q} [VOLU | UNIF | SCAL] [ADD]

**Φ** A symbol that denotes the variable for which the values are specified. This input mode can only be used for the mass species and heat transfer variables for which a differential equation is solved. The valid symbols include those listed in Table 2.7.1, except the pressure variables, plus the user defined variables (ALLOCATE command) for which a differential equations a solved. There is no default value; a symbol must be specified.

**INVENT** The total inventory or amount of the property, Φ, is specified. The inventory, Q, for a subregion is defined as:

$$Q = \sum_{\text{subrgn}} \alpha_i \Phi_i \delta V_i$$

Here Q the subscript “i” denotes the values for an element, α is the accumulation coefficient for the transport equation, and δV is the volume of the element. The summation is taken over all elements of the subregion.

**subrgn** The subregion for which the input is specified. If no subregion is specified, then entire computational domain is selected.

**Q** The specified inventory (> 0) of the property, Φ.

**VOLU** The inventory is distributed so that the amount added to each element is proportional to the volume of the element. This is the default option. The new value is computed as:

$$\Phi_i = \theta \Phi_i^0 + \frac{1}{\alpha_i} \frac{Q}{\sum_{\text{subrgn}} \delta V_i}$$

Here Φ<sub>i</sub><sup>0</sup> is the existing value of variable, and θ = 0 by default but is set to 1 if modifier ADD is present.

**UNIF** The inventory is distributed so that the increment in Φ is uniform and constant:

$$\Phi_i = \theta \Phi_i^0 + \frac{Q}{\sum_{\text{subrgn}} \alpha_i \delta V_i}$$

**SCAL** The inventory is distributed proportional to the existing value of Φ:

$$\Phi_i = \theta \Phi_i^0 + \frac{Q \Phi_i^0}{\sum_{\text{subrgn}} \alpha_i \Phi_i^0 \delta V_i}$$

Since the old value of Φ appears in the denominator, this modifier should be used only if the variable is positive definite; that is all values are greater than zero.

**ADD** By default the specified inventory replaces any existing value; that is the constant θ = 0 in the above equation. If this modifier is present, then the inventory is added to the existing value; that is θ = 1.

**REMARKS**

This command can be used to set the values only in the interior elements of the computational domain. The external boundary nodes are excluded. If this command is used in the middle of a solution in progress, it net effect is to inject (or withdraw, if replaced amount is greater than the existing amount) property into the domain. The balance for the property will then show a net flux disparity.

**EXAMPLES**

SET INVENTORY of mass species C to 10 kg

SET **INVENTORY** of **FF** to **10** kg; distribute **UNIFormly** over **SELEcted** region  
SET **T INVENTORY** in **ID=RGN1** is **1.E6** units; **SCALE** with existing value

**MODE 10:** Transfer of Inventory From One Subregion to Another

**SYNTAX** SET {Φ} {INVENT} {ID=subrgn1} {ID=subrgn2} [VOLU | UNIF | SCAL] [β] [FRAC | TIME] [ALWA | OFF]

**Φ** A symbol that denotes the variable for which the values are specified. This input mode can only be used for the mass species and heat transfer variables for which a differential equation is solved. The valid symbols include those listed in Table 2.7.1, except the pressure variables, plus the user defined variables (ALLOCATE command) for which a differential equation is solved. There is no default value; a symbol must be specified.

**INVENT** The inventory of the property is transferred from a donor to a receptor region. See the previous mode of the command for a more complete definition.

**ID=subrgn1** The donor region from which the inventory is transferred to the receptor region. There is no default value. A valid subregion name with an ID modifier must be specified. The total amount of inventory, Q, which is transferred from the donor region, is computed as:

$$Q = \lambda \sum_{\substack{\text{subrgn1} \\ \text{(Donor)}}} \alpha_i \Phi_i^o \delta V_i$$

Here superscript 'o' denotes the existing value of the variable Φ, subscript "i" denotes the value for an element, α is the accumulation coefficient for the transport equation, δV is the volume of the element, and λ is a constant. The summation is taken over all elements of the donor subregion. The constant, λ, is determined as follows:

- λ = 1 by default if β is not specified.
- λ = min (β, 1) if β is specified but TIME is not specified
- λ = min (β δt, 1) where δt is the time step if β and TIME are specified

Values of the variable for all elements in the donor region are decreased to preserve mass balance for the variable. The modified value, i.e. the remaining property, for element 'i' is computed as:

$$\Phi_i = (1 - \lambda) \Phi_i^o$$

**ID=subrgn2** The receptor region to which the inventory is transferred. There is no default value. A valid subregion name with an ID modifier must be specified.

**VOLU** The donor inventory is distributed so that the amount added to each receptor element is proportional to the volume of the element. This is the default option. The new value is computed as

$$\Phi_i = \Phi_i^o + \frac{1}{\alpha_i} \frac{Q}{\sum_{\substack{\text{subrgn2} \\ \text{(Receptor)}}} \delta V_i}$$

**UNIF** The inventory is distributed so that the increment in Φ is uniform and constant:

$$\Phi_i = \Phi_i^o + \frac{Q}{\sum_{\substack{\text{subrgn2} \\ \text{(Receptor)}}} \alpha_i \delta V_i}$$

**SCAL** The inventory is distributed proportional to the existing value of Φ:

$$\Phi_i = \Phi_i^o + \frac{Q \Phi_i^o}{\sum_{\substack{\text{subrgn2} \\ \text{(Receptor)}}} \alpha_i \Phi_i^o \delta V_i}$$

Since the old value of Φ appears in the denominator, this modifier should be used only if the variable is positive definite; that is all values are greater than zero.

- $\beta$**  A constant ( $\beta \geq 0$ ) that determines how much of the current inventory is transferred from the donor to the receptor region. The specified value must not be negative. If no value is specified, then the total amount is transferred.
- FRAC**  $\beta$  specifies the fraction of the current inventory in the donor region that is transferred. Limits are imposed so that ( $0 \leq \beta \leq 1$ ); that is the minimum transferred amount is zero and the maximum is equal to the total amount in the donor region at any time. This is the default interpretation of  $\beta$ .
- TIME**  $\beta$  specifies the fraction per unit time of the current inventory that is transferred. The actual fraction transferred is equal to  $\beta \delta t$  where  $\delta t$  is the time step. Limits are imposed dynamically so that ( $0 \leq \beta \delta t \leq 1$ ).
- ALWA** By default, the specified amount is transferred only once, that is, the first time the command is encountered. If the **TIME** modifier is present then the specified amount is transferred at each step starting with the first time the command is encountered.
- OFF** If this modifier is present then any previous **SET INVE** command that was specified by the **ALWA**s modifier for the same  $\Phi$ , and the same donor and receptor subregions is disabled.

### REMARKS

---

Inventory can be transferred only between interior elements of the computational domain. The external boundary nodes are excluded. Since the command essentially transfers the property from one region to another, there is no net effect on the total amount of property in the domain. If total inventory is transferred from one region to another, then the variable in the donor region will instantaneously fall to zero.

This command can be used, for example, to approximate the transfer of a contaminant from an originally un-collapsed containment area (such as a vault) to a smaller area after collapse without revising the mesh. It provides an approximate method of simulating the settling of a contaminated region without having to deal with the additional complexity of the physics of consolidation.

Another example of its use is that of leakage of a contaminant from one region to another through a structure (such as a duct or pipe) that is not explicitly modeled by the grid but the leakage rate is determined from other considerations.

### EXAMPLES

---

**SET INVE**ntory of C from ID=DONOR to ID=RECEPTOR! Total amount transferred  
**SET INVE**ntory of C from ID=DONOR to ID=RECEPTOR FRACTION=0.5! 50% transferred  
**SET INVE**ntory of C from ID=DONOR to ID=RECEPTOR 0.1 TIME ALWAYS! 0.1\*dt every time step  
**SET INVE**ntory of C from ID=DONOR to ID=RECEPTOR OFF! previous command off

**MODE 11:** Computation of Gradient of a Variable

**SYNTAX** SET {GRAD} {Ψ} {Φ} {ξ} [fnc] [BLOC] [FIEL] [ALWA|OFF]

**GRAD** The Gradient of the specified variable is computed as:

$$\Psi = \frac{\partial\Phi}{\partial\xi} \quad \text{or} \quad \Psi = \text{function}\left(\frac{\partial\Phi}{\partial\xi}\right)$$

The function, if any, is defined by the **fnc** modifier as given below.

**Ψ** The symbol for the variable where the computed gradient values are stored. This must be a valid symbol for one of the default field variables of ACRi Software or it must be allocated through the **ALLOCATE** command.

**Φ** Symbol for one of the previously defined field variables listed in Table 2.7.1. There is no default value; a variable must be specified.

**ξ** One of the symbols: X, Y, r or Z that denotes the coordinate with respect to which the gradient is computed. The symbol r can only be used if the specified coordinate system is cylindrical. There is no default value; a value must be specified.

**fnc** The modifier for function evaluation

fnc	INTERPRETATION
<b>ABS</b> or <b>ABSO</b>	The absolute value for the computed gradient is taken.
<b>POSI</b>	The negative values for the computed gradient are set of zero
<b>NEGA</b>	The positive values for the computed gradient are set of zero
<b>SQUA</b>	The function is computed as the square of the gradient
<b>ROOT</b>	The function is computed as the square root of the gradient; negative values are set to 0.

**BLOC** If this modifier is present then the value of the gradient at the blocked nodes see (**BLOCK** command) is set to zero. Otherwise the computed value is retained.

**FIEL** By default the gradient is computed throughout the computational domain including the field and the boundary nodes. If this modifier is present, then only the field values are computed.

**ALWA** By default the **SET** command is implemented immediately and only once – as soon as the command is encountered. If this modifier is present then the command is implemented immediately as well as repeatedly at the beginning of every time step (or iterative step in steady state mode) of the solution procedure.

**OFF** Any previous **SET** command with **GRAD** modifier for the specified variable is deactivated.

**EXAMPLES**

- SET DTDX = GRAD of T with respect to X
- SET DTDR = ABSOLUTE value of GRAD of T with respect to R compute ALWAYS
- SET PHIR = GRAD of T with respect to R only once
- SET DTDY = GRAD of T with respect to Y for FIELd nodes only with BLOC value set to zero
- SET DTDY = POSITIVE values of GRAD of T with Y with BLOC = 0
- SET DTDY = SQUARE ROOT of GRAD of T with Y with BLOC = 0
- SET DTDY OFF

**MODE 12:** Specification of Multiple Variables from Table of Values at Vertices**SYNTAX** SET {VERT} { $\Phi_1, \dots, \Phi_n$ } { $V_1, \dots, V_{(n+1)m}$ |fname} [subrgn] [FIEL] {dir|BOUN}

**VERT** The  $\Phi_n$  variables are set individually for each of the external boundary nodes of a computational domain from a given set of values at the vertices. It is assumed that the input is in a tabular format with **n** columns (1 for each variable) and **m** rows (1 row for each boundary vertex). The input for the variables must be provided for **all the vertices** required to set the values for the identified sub domain. However, additional vertices of the domain may be present in the input file.

**$\Phi_1, \dots, \Phi_n$**  Symbols to denote the variables that are to be set by the command. The symbols include all those listed in Table 2.8.1-3 plus the user defined variables (**ALLOCATE** command) and the real type variables defined in the initialization file. **In addition a special symbol "SKIP" can be specified to by-pass the input of any of the columns.** Any symbol may occur multiple times. There is no default value; appropriate symbols must be specified.

**$V_1, \dots, V_{nm}$**  The table of values in the format of columns and rows. Each row consists of a vertex number followed by **n** values – one for each of the specified symbols. If **n** symbols are specified on the command and there are **m** boundary vertices, then **(n+1)** times **m** values must be specified in **n+1** columns and **m** rows.

**fname** The name of the file (see Section 3.3) from which  **$V_1$**  through  **$V_{(n+1)m}$**  are read.

**subrgn** The subregion for which the input is specified. **If no subregion is specified, then entire computational domain is selected.**

**FIEL** See Mode 1 specification.

**dir** The input is applied only to the nodes at the subregion boundary where the outward normal matches the specified modifier. See Section 3.5 for available choices.

**BOUN** If this modifier is present then the values at all boundary nodes of the specified subregion are set by this command.

**EXAMPLES**

**SET** boundary from VERTEX locations! 12 boundary vertices for a 4 by 3 by 3 grid

#	SKIP	P	T	SKIP	C
1	0.2	1.0	-5	10.20	100
2	0.6	0.0	+5	12.50	150
3	1.0	-1.0	+10	2.00	200
4	0.2	1.0	-5	10.20	100
5	0.6	0.0	+5	12.50	150
6	1.0	-1.0	+10	2.00	200
7	0.2	1.0	-5	10.20	100
8	0.6	0.0	+5	12.50	150
9	1.0	-1.0	+10	2.00	200
10	0.2	1.0	-5	10.20	100
11	0.6	0.0	+5	12.50	150
12	1.0	-1.0	+10	2.00	200

**SET** boundary P T SKIP C from VERTEX data on file 'VERTEX.VAL!' Table of vertex values on file

**MODE 13: Special User Empirical Function for Colloidal Settling**

**SYNTAX** SET {ZALU} {Φ}[ξ] {x<sub>0</sub>, y<sub>0</sub>, [z<sub>0</sub>]} {q<sub>0</sub>} {t<sub>start</sub>, t<sub>end</sub>} [C<sub>0</sub>,C<sub>1</sub>,C<sub>2</sub>,C<sub>3</sub>] [LIMI] [subrgn] [FIEL] [ALWA] [FREQ [NOW]] [ONLY] [TIME]=V<sub>frq</sub>

**ZALU** This modifier indicates that a special user function is invoked. This function was developed by Dr. Zaluski for the deposition rate of a specific iron colloidal solution. It expresses the deposition rate, q, (in kg/kg of soil) of the colloids in terms of multi-variate function of time, t (seconds), distance, d (m) and modulus of the Darcy velocity, U (m/s). The specific function is:

$$q = \xi \max [(C_0 + C_1 t + C_2 d + C_3 U), 0.0] \quad \text{if } t_{\text{start}} < t \leq t_{\text{end}} \quad \text{and } S_f \geq 0.99$$

$$= 0 \text{ otherwise}$$

The default values of the constants are: C<sub>0</sub> = 4.0581x10<sup>-2</sup>, C<sub>1</sub>= 2.88x10<sup>-7</sup>, C<sub>2</sub>=-2.294x10<sup>-2</sup>, C<sub>3</sub>= 19.68013. The q is further modified to ensure that the total amount present in the field does not exceed the total released up to that point. The total amount in the field and the total injected are computed as:

$$Q_{\text{total}} = \sum q (1-\theta) \rho_s \delta V ; \quad Q_{\text{inj}} = q_0 \min [(t - t_{\text{start}}), (t_{\text{end}} - t_{\text{start}})]; \quad t > t_{\text{start}}$$

In these relations, ρ<sub>s</sub> is the particle density of solid, θ is the porosity and δV is the volume of the element; the summation is carried out over all elements of the sub-domain if the modifier **LIMIT** is present then q is corrected according to:

$$q_{\text{final}} = q \min \left[ \frac{Q_{\text{inj}}}{Q_{\text{total}}}, 1.0 \right]$$

If the modifier **LIMIT** is not present, then full correction is applied so that:

$$q_{\text{final}} = q \frac{Q_{\text{inj}}}{Q_{\text{total}}}$$

**Φ** Symbol for one of the previously defined field variables that will contain the computed deposition rate. **There is no default value; a variable must be specified.**

**ξ** The independent variable that multiplies the computed value of q. The variable **ξ** can be computed from an equation or specified by other means. **If the symbol ξ is not specified, then it is replaced by unity.**

**x<sub>0</sub>, y<sub>0</sub>, [z<sub>0</sub>]** The coordinates of the point of injection of the colloid. Two numerical values must be specified for 2D and 3 for 3D geometry. **There is no default value**

**q<sub>0</sub>** The injection rate of the colloids. The units must be consistent with the units of the correlation equation given above.

**t<sub>start</sub>, t<sub>end</sub>** The start and end time for the injection of the colloids.

**C<sub>0</sub>,C<sub>1</sub>,C<sub>2</sub>,C<sub>3</sub>** The constants of the deposition relation. Default values are given above.

**LIMI** If this modified is present, then the computed q is corrected only if the total computed amount exceeds the injected amount. **See the equations above.**

**subrgn** The subregion for which the colloidal deposition is computed. **If no subregion is specified, then entire computational domain is selected.**

**FIEL** See Mode 1 specification.

**ALWA** **By default the computation is performed only once when the command is given.** If this modifier is present, then the computation is performed at every step.

**FREQ** The computation is performed at the frequency specified by **V<sub>frq</sub>**.

**NOW** The computation is performed immediately and at the frequency specified by **V<sub>frq</sub>**.

- ONLY** The computation is performed only once at the step number or time specified by the **V<sub>frq</sub>**.
- TIME** By default, **V<sub>frq</sub>** is the frequency of output in terms of number of steps. If this modifier is present, then **V<sub>frq</sub>** is interpreted to be time interval between successive outputs.
- V<sub>frq</sub>** The frequency (step or time interval) at which the computation is performed.

#### REMARKS

---

This is a very specific and limited-applicability empirical function developed by Dr. Marek Zaluski ([marek.zaluski@mse-ta.com](mailto:marek.zaluski@mse-ta.com)) and his colleagues for the deposition rate of an iron colloidal solution. It should only be used for type of application for which it was develop, and then only within the range of conditions for which it was developed.

#### EXAMPLES

---

**SET ZALUSKI** function for COLLIOD at (0.1,0.5) q=0.55, tstart=10, tend=500 ALWAYS  
**SET ZALUSKI** COLLIOD at (0.1,0.5,0.4) q=0.55, tstart=10, tend=500 FREQ=5 in TIME units  
**SET ZALU CO** (0.1,0.5,0.4) q=0.55, ts=10, te=50, c0=0.1, c1=0.005, c2=0.1, c3=1.2 FREQ=5 TIME  
**SET ZALU CO** multiply by C (0.1,0.5,0.4) q=0.55, ts=10, te=50 ALWAYS

**MODE 14:**     Disable Previously Specified SET Commands

**SYNTAX**     **SET**   {Φ} {OFF} [subrgn]

**Φ**             See Mode 1 specification.

**OFF**          Previous **SET** commands for the specified variable in the identified subregion are deactivated. A new specification may follow.

**subrgn**       Identifier for the subregion that appeared previously in a **SET** command for the current variable, Φ.

#### **EXAMPLES**

---

**SET** commands for T for OFF for currently SELEcted subregion

**SET** commands OFF for T for ID=MIDDLE

**5.99 SOLID**

**PURPOSE** To specify the conjugate heat transfer mode for the **ANSWER®** Software Tool.

**SYNTAX** **SOLI**

**APPLICABILITY** \_\_\_\_\_

This command is available only for the **ANSWER®** Software Tool.

**REMARKS** \_\_\_\_\_

By default the thermal transport in the solid is ignored in the **ANSWER®** Software Tool. If this command is present, the thermal equation accounts for both the fluid and the solid phases. The properties of the solid must be specified with the appropriate **CONDUCTIVITY**, **DENSITY**, **POROSITY** and **SPECIFIC HEAT** commands.

**EXAMPLES** \_\_\_\_\_

**SOLI**: thermal transport to be accounted for.

## 5.100 SOLVE

**PURPOSE** To initiate solution of the governing equations and to select the transient or steady state mode of solution.

**MODE 1:** Transient Solution Mode With Manual Time Step

**SYNTAX** SOLV [ $\Phi$ ] [ADD] [MANU] [ABSO] [ $t_{Max}$ , dt,  $\alpha$ ,  $dt_{Max}$ ] [ $N_{Step}$ ]

**$\Phi$**  By default, an appropriate set of governing equations is solved based on the user-specified input. The default selection includes any variable that occurs on a **BOUNDARY** or **SOURCE** command and the basic flow and thermal variables that occur on a **SET** command.

The user may override the default selection by explicitly specifying one or more  $\Phi$  symbols to select the corresponding dependent variable for which the governing equations are solved. The valid symbols are listed in Table 2.7.1. See the **ADD** modifier for more details.

**ADD** If this modifier is present, then the variables specified by  $\Phi$  are added to the list of default variables for which a governing equation is solved. In the absence of the **ADD** modifier, the governing equations are solved **ONLY** for the variables specified by  $\Phi$ .

**MANU** The time step is determined according to the user-specified input of **dt**,  **$\alpha$**  and  **$dt_{Max}$** . This is the default mode.

**ABSO** By default the  **$t_{Max}$**  value below is assumed to be the incremental time period and is added to the current time to determine the total time for simulations. If this modifier is present, then  **$t_{Max}$**  is interpreted to be the time in absolute units.

**$t_{Max}$**  The incremental time period (**>0**) (default) or the absolute time (**ABSO** modifier; **>0**) for solution of the governing equations. The default value is 0.

**dt** The time step (**>0**) at the start of the current set of calculations. The time step may be changed during the calculations or by a subsequent **SOLVE** command. The default value is  **$t_{Max}/1000$** .

**$\alpha$**  The geometric ratio multiplier (**>0**) for the time step. Each successive time step is multiplied by this value until a maximum value specified by  **$dt_{Max}$**  is reached. The default value is 1.

**$dt_{Max}$**  The maximum permissible time step. The default value is  $10^{30}$ .

**$N_{Step}$**  The maximum number of time steps. The default value is 99 999 999.

## EXAMPLES

---

**SOLVE** for 50 years in steps of 2

**SOLVE** for 50 hours, initial step=0.2 hr, increase by 1.1, max=10 hr

**SOLVE** 1.E6 yrs, DT=1, fac=1.1, dtmax=1000, max steps=1000

**SOLVE** for 10 days, **MANUAL** mode from now on! Previously an **AUTO** mode was used.

**SOLVE** for **T** only for 10 days in steps of 0.2

**SOLVE** for **P2**, **T** and **C3** only for 10 days in steps of 0.2

**MODE 2:** Transient Solution Mode with Flux Based Automatic Time Step**SYNTAX** SOLV [ $\Phi$ ] [ADD] {AUTO} [PASS] [ABSO] [ $t_{Max}$ , dt,  $\alpha$ ,  $dt_{Max}$ ,  $dt_{Min}$ ,  $\beta$ ,  $N_{Step}$ ] **$\Phi$**  See Mode 1.**ADD** See Mode 1.**AUTO** The time step is automatically determined so that the specified primary convergence constraint is achieved for each time step (See **CONVERGENCE** command with **FLOW** modifier). The time step is increased if the number of iterations to converge is less than 1/2 of the maximum number of iterations specified on the **CONVERGENCE** command with **FLOW** modifier and decreased if it is larger than 4/5<sup>th</sup> of the maximum number of iterations.**PASS** With the **AUTO** modifier, the time step may be adjusted either dynamically or passively. This modifier disables the dynamic mode; only passive adjustment takes place. In the dynamic adjustment mode, if the solution fails to converge to a specified tolerance in the specified number of iterations (see above), the time step is decreased and the solution process is repeated from the previous time. In the passive adjustment mode, the time step is adjusted only from the next time step; the currently computed values are retained.**ABSO** See Mode 1. **$t_{Max}$**  See Mode 1.**dt** See Mode 1. **$\alpha$**  The geometric ratio multiplier (>0) for the time step. Each successive time step is multiplied by this value until a maximum value specified by  **$dt_{Max}$**  is reached. The default value is 1.01 **$dt_{Max}$**  See Mode 1. **$dt_{Min}$**  The minimum permissible time step. With the **AUTO** modifier, it is very important to specify a reasonable lower bound for the time step. If too low a value is specified, computer resources may be wasted. Alternatively, too high a value may lead to unacceptable numerical errors. The default value is  $10^{-10}$ . **$\beta$**  The geometric ratio divisor (>0) for the time step. Each successive time step is decreased by this factor if the number of iterations to convergence is larger than the threshold specified in the **CONVERGENCE** command. The default value is 2. **$N_{Step}$**  The maximum number of time steps. The default value is 99 999 999.**REMARKS**

The **AUTOMATIC** solution mode is currently available only with the **PORFLOW®** Software Tool. It works quite efficiently for single phase (saturated mode) flow problems. However, with multiphase problems, this mode may lead to excessive computation time unless the residual convergence criterion has been properly set. It is recommended that for such problems initially a manual time step be employed till a satisfactory solution behavior has been established.

**EXAMPLES**SOLVE for 10 days in **AUTOMATIC** modeSOLVE **AUTO** 10 days DT=0.2, alpha=1.3, dtmax=1, dtmn=1.E-5, beta=1.1, Nmax=1000SOLVE for T only for 10 days in **AUTO** mode

**MODE 3:** Transient Solution Mode with Automatic Time Step Based on CFL-Type of Limit

**SYNTAX** SOLV [ $\Phi$ ] [ADD] {CFL} [ABSO] [ $t_{Max}$ , dt,  $\alpha$ , dt<sub>Max</sub>, dt<sub>Min</sub>, R<sub>max</sub>,  $\beta$ , N<sub>Step</sub>]

$\Phi$  See Mode 1.

ADD See Mode 1.

**CFL** The time step is automatically determined so that largest normalized matrix coefficient for any of equations is smaller than the prescribed threshold value (Runchal 2004); that is:

$$R = \max_{\text{over all } i} \frac{\max_{|i \neq j} |A_{ij}|}{A_{jj}} \leq R_{\max}$$

where  $A_{ij}$  are the coefficients of the solution matrix:  $A_{ij} X_j = B_i$

This criterion is numerically equivalent to the CFL criterion except that the matrix terms also include the contribution from the diffusion terms. For the same reason, it is also more versatile than the CFL criterion in that it incorporates the elements of both the CFL and the “diffusion” limit in a single criterion, which is numerically convenient to compute. It essentially complies with the oft-stated heuristic limit for matrix stability so that none of the errors for a linear system shall grow with an iterative process.

**ABSO** See Mode 1 of command.

$t_{Max}$  See Mode 1 of command.

dt See Mode 1 of command.

$\alpha$  See Mode 1 of command.

dt<sub>Max</sub> See Mode 1 of command.

dt<sub>Min</sub> The minimum permissible time step. It is very important to specify a reasonable lower bound. If too low a value is specified, computer resources may be wasted. Alternatively, too high a value may lead to unacceptable numerical errors. The default value is 10<sup>-10</sup>.

R<sub>max</sub> The cut-off threshold value, R<sub>max</sub>, for the CFL parameter. If the computed value of R exceeds the threshold value then new time step is computed from:

$$dt_{\text{new}} = dt_{\text{old}} / \lambda; \quad \lambda = \max\left(\frac{R}{R_{\max}}, \beta\right)$$

where  $\beta$  is defined below. The default value is 0.8

$\beta$  The minimum geometric ratio divisor (>0) for the time step. The default value is 1.1.

N<sub>Step</sub> The maximum number of time steps. The default value is 99 999 999.

**REMARKS**

The **CFL** solution mode is currently available only with the **ANSWER®** Software Tool. It works quite efficiently for most flow problems. However, due to non-linearity of the governing equations, complex sources and boundary conditions, there is no guarantee that any automatic procedure will prove economic under all circumstances. It is recommended that in case of doubt, initially a manual time step be employed till a satisfactory solution behavior has been established.

**EXAMPLES**

SOLVE for 10 days in CFL mode

SOLVE CFL 10 sec DT=0.2, alpha=1.0, dtmax=1, dtmin=1.E-5, Rmax=0.5 beta=1.5 Nmax=1000

SOLVE for T only for 10 seconds in CFL automatic mode with DT=0.2

**MODE 4:** Transient Solution with Automatic Time Step for TIDAL Software Tool**SYNTAX** SOLV [ $\Phi$ ] [ADD] {AUTO} [ABSO] [ $t_{Max}$ , dt,  $\alpha$ ,  $dt_{Max}$ ,  $dt_{Min}$ ,  $N_{Step}$ ] $\Phi$  See Mode 1.**ADD** See Mode 1.**AUTO** The time step is automatically determined so that the time step stays within the computed value of the CFL limit for the flow.**ABSO** See Mode 1. $t_{Max}$  See Mode 1.**dt** The starting value of the time step which will be replaced by the computed value if is larger than the computed value $\alpha$  The geometric ratio multiplier (**>0**) for the time step. Each successive time step is multiplied by this value until the maximum value (**dt<sub>Max</sub>**) is reached. The default value is 1.01**dt<sub>Max</sub>** The maximum permissible time step. The default value is  $10^{30}$ .**dt<sub>Min</sub>** The minimum permissible time step. With the **AUTO** modifier, it is very important to specify a reasonable lower bound for the time step. If too low a value is specified, computer resources may be wasted. Alternatively, too high a value may lead to unacceptable numerical errors. The default value is  $10^{-10}$ .**N<sub>Step</sub>** The maximum number of time steps. The default value is 99 999 999.**REMARKS**

The **AUTOmatic** solution mode is currently available only with the **TIDAL®** Software Tool. It works quite efficiently for most flow problems. However often, for problems without run-up and draw-down, larger time can be used. In the case, the multiplier ( $\alpha$  above) can be larger than unity. For such problem the multipliers of 2 to 5 can be used for most problems.

**EXAMPLES**SOLVE for 3.E7 seconds in **AUTO** modeSOLVE **AUTO** 3.E7 seconds DT=100, Factor =2.0

**MODE 5:** Transient Solution with Time Step as a Function of Time

**SYNTAX** SOLV [ $\Phi$ ] [ADD] [ABSO] { $t_{Max}$ } {FUNC}{ $N_1, N_2, \dots, N_n$ |fname} [MAX=dt<sub>Max</sub>, MIN=dt<sub>Min</sub>, STEP=N<sub>Step</sub>]

$\Phi$  See Mode 1.

**ADD** See Mode 1.

**ABSO** See Mode 1.

$t_{Max}$  See Mode 1.

**FUNC** The presence of this modifier implies that the time step is computed as a function of time.

$N_1, \dots, N_n$  The numerical constants and coefficients for the selected function of time. See Section 4.4 for more details. There are no default values for this input.

**fname** The name of the file containing the numerical values  $N_2$  through  $N_n$ . This option is available only for selected functions. See Section 3.3 for additional information.

**MAX** The presence of this modifier implies that the maximum time step for the computations is being specified.

**dt<sub>Max</sub>** The maximum permissible time step. The numerical value must immediately follow the **MAX** modifier. The default value is  $10^{30}$ .

**MIN** The presence of this modifier implies that the minimum time step for the computations is being specified.

**dt<sub>Min</sub>** The minimum permissible time step. The numerical value must immediately follow the **MIN** modifier. The default value is  $10^{-10}$ .

**STEP** The presence of this modifier implies that the maximum number of time steps for the computations is being specified.

**N<sub>Step</sub>** The maximum number of time steps. The numerical value must immediately follow the **STEP** modifier. The default value is 99 999 999.

**EXAMPLES**

**SOLVE** for 300 seconds with **FUNCTION**: dt= **LINEAR 0.2 +0.01** \* time

**SOLVE** for 300 seconds with **FUNCTION**: dt= **LINEAR 0.2 +0.01** \* time **MAX=1.0**

**MODE 5:**      **Steady State Solution Mode**

**SYNTAX**      **SOLV {**STEA**} [**Φ**] [**ADD**] [**N<sub>step</sub>**, **N<sub>step\_min</sub>**]**

**STEA**            The equations are solved in their steady state mode. That is, the storage or accumulation term in the governing transport equation is set to zero.

**Φ**                See Mode 1.

**ADD**            See Mode 1.

**N<sub>step</sub>**            The maximum number of iterative steps for solution of the matrix. **The default value in this instance is 99999999.**

**N<sub>step\_min</sub>**        The minimum number of iterative steps to be performed on the matrix. **The default value is 100.**

#### **EXAMPLES**

---

**SOLVE** in **STEADY** state mode: maximum steps **200**

**SOLVE** in **STEADY** mode: maximum steps **500**; minimum steps **20**

**MODE 6:**      **Frequency of Computation for Selected Variables**

**SYNTAX**      **SOLV {FREQ} [ $\Phi$ ] [N<sub>frq</sub>]**

**FREQ**      By default all variables are solved at every step of the solution process. Any exceptions to this are noted in the relevant sections. This modifier may be used to set a frequency for the computation of selected variables.

$\Phi$       **One or more** symbols to specify the variables for which the solution frequency is explicitly specified. Valid symbols are listed in Table 2.7.1. **There is no default value.**

**N<sub>frq</sub>**      The frequency, in terms of number of steps of the solution process, for the solution of the variables identified on the command.

#### **EXAMPLES**

---

**SOLVE FREQUENCY** for **C** is every **10** steps

**SOLVE FREQUENCY** for **FU**, **CO** and **C2** every **20** steps

**MODE 7:**      **Dummy Run Initial Condition Mode**

**SYNTAX**      **SOLV {OFF}**

**OFF**            The actual solution of the equations is disabled; however, a dummy run through the solution process is made to compute values of derived variables for the specified initial and boundary conditions. This mode is useful to check initial conditions and to create output or archive files of initial data and problem geometry for checking and verification of input.

#### **REMARKS**

---

The Mode 1 and 2 of the **SOLVE** command immediately initiate solution of the governing differential equations. Therefore, these commands should be specified only after input has been supplied that is complete to initiate solution. However, the sequence of calculations may be sub-divided into as many segments as desired and a **SOLVE** command specified for each segment. See Section 2.10 for more information.

#### **EXAMPLES**

---

**SOLVe OFF**

**SOLVe OFF** now to obtain initial output for problem verification.

**5.101 SOOT**

**PURPOSE** To activate soot computation and specify soot constants and parameters.

**MODE 1:** **Soot Particle Constants**

**SYNTAX** **SOOT** [**N<sub>1</sub>**, **N<sub>2</sub>**, **N<sub>3</sub>**, **N<sub>4</sub>**]

**N<sub>1</sub>** Soot particle density. The default value is 2000 (kg/m<sup>3</sup>).

**N<sub>2</sub>** Diameter of the smaller soot particles in microns. The default value is 1/40 micron.

**N<sub>3</sub>** Diameter of the larger soot particles in microns. If this value is  $\geq 0$  or is omitted, then the computations are performed with only one soot particle size specified by **N<sub>2</sub>**. The default value is 1 micron.

**N<sub>4</sub>** Percentage of the smaller soot particles in the total. This input is ignored if **N<sub>3</sub>** is zero or is omitted. The default is 90 percent.

**APPLICABILITY**

---

This command is available only for the **ANSWER®** Software Tool.

**EXAMPLES**

---

**SOOT** rho= 2200, d1 = 0.05

**SOOT** rho= 2200, d1 = 0.05, d2= 2., fraction 50 %

**MODE 2:**      **Soot Nuclei Constants**

**SYNTAX**      **SOOT {NUCL} [N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>6</sub>]**

**NUCL**          Soot nuclei constants are specified.

**N<sub>1</sub>**              The Arrhenius prefactor (**>0**) for the soot nuclei equation. The default value is  $1.16 \times 10^{31}$ .

**N<sub>2</sub>**              The Arrhenius activation energy constant divided by the universal gas constant (**>0**) for the soot nuclei equation. The default value is  $9.00 \times 10^4$ .

**N<sub>3</sub>**              The 1<sup>st</sup> exponent (**≥0**) of the soot nuclei equation. The default value is 1,000.

**N<sub>4</sub>**              The 2<sup>nd</sup> exponent of the soot nuclei equation. The default value is  $8.00 \times 10^{-16}$ .

**N<sub>5</sub>**              The 3<sup>rd</sup> exponent of the soot nuclei equation. The default value is  $1.00 \times 10^{15}$ .

**N<sub>6</sub>**              The constant (**≥0**) in the soot nuclei equation. The default value is 100.

#### **EXAMPLES**

---

**SOOT** NUCLei constants: 1.00E31, 1.E5, 1000, 1.E-15, 1.E-15, 100.

**SOOT** NUCLei constants: 5.00E31, 1.0E4, 900, 8.E-15, 1.E-14, 100.

**MODE 3:** Soot Formation Constants

**SYNTAX** SOOT {FORM} [N<sub>1</sub>, N<sub>2</sub>, , N<sub>7</sub>]

**FORM** Soot formation constants are specified.

**N<sub>1</sub>** The Arrhenius prefactor (>0) for the soot formation equation. The default value is  $1.5626 \times 10^{13}$ .

**N<sub>2</sub>** The Arrhenius activation energy constant divided by the universal gas constant (>0) for the soot nuclei equation. The default value is  $1.60 \times 10^4$ .

**N<sub>3</sub>** The 1<sup>st</sup> exponent (≥0) of the soot formation equation. The default value is 1.81.

**N<sub>4</sub>** The 2<sup>nd</sup> exponent of the soot formation equation. The default value is -0.50.

**N<sub>5</sub>** The 3<sup>rd</sup> exponent of the soot formation equation. The default value is -1.94.

**N<sub>6</sub>** The incipient carbon-to-oxygen ratio at which the soot formation starts. The default value is 0.1.

**N<sub>7</sub>** The incipient temperature (in absolute units) at which soot formation starts. The default value is 400.

### EXAMPLES

---

**SOOT FORM**ation constants are 1.90E13, 1.50E4, 2.0, -0.5, -2.0, 0.1, 500.

**SOOT FORM**ation constants: 1.90E13, 1.50E4, 2.0, -0.5, -2.0, 0.1, 500.

**MODE 4:** Soot Computation Frequency

**SYNTAX** SOOT {FREQ} [N<sub>1</sub>]

**FREQ** Frequency of soot computations is specified.

**N<sub>1</sub>** The frequency of soot computations in terms of number of steps of the solution process. If the value specified is <1 or the input is omitted, then the soot computations are triggered at the default or previously specified frequency. The default value is 10.

#### EXAMPLES

---

**SOOT** FREQuency of calculations every 20 steps

**SOOT** computation FREQuency by default.

5.102 SOURCE

**PURPOSE** To specify the sources (or sinks) of fluid, heat or chemical species.

**MODE 1:** Direct Source or Sink for a Single Selected Variable

**SYNTAX** SOUR {Φ} [TOTA] [func[ξ]] {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub>|fname} [option] [subrgn] [dir] [N<sub>n+1</sub>, N<sub>k</sub>], [ρ<sub>B</sub>]

**Φ** The symbol for the dependent variable for which the source is specified. Valid symbols are listed in Table 2.7.1. A symbol must be specified.

**TOTA** By default, the amount of source specified, or computed from func (ξ), is applied to each element of the subrgn. If this modifier is present then the amount is assumed to be the total amount over whole of the subrgn. In this case, the amount is distributed equally to all the elements of the subrgn unless the VOLUME or AREA modifiers are present.

**func** One of the modifiers listed in Table 4.2.1 that denotes the functional form of the source. If no function is specified, the value is assumed to be constant.

**ξ** One of the independent variables listed in Table 4.2.2. If no independent variable is specified, the variable is assumed to be time.

**N<sub>1</sub>,...,N<sub>n</sub>** The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.

**fname** The name of the file containing the numerical values N<sub>2</sub> through N<sub>n</sub>. This option is available only for selected functions. See Section 3.3 for additional information.

**option** Options selected for implementation of the source.

option	INTERPRETATION
<b>VOLU</b>	If the <b>TOTAL</b> modifier is not present, the source for each element is computed as: $Q = q \delta V$ . Here $q$ is the amount specified by the user and $\delta V$ is the volume of the element. The $q$ , in turn, is computed from $func(\xi)$ and $N_1$ through $N_n$ . If the <b>TOTAL</b> modifier is present, the amount for each element is computed as: $Q = q \delta V / V$ , where $V$ is the volume of the total <b>subrgn</b> .
<b>AREA</b>	If the <b>TOTAL</b> modifier is not present, the source for each element is computed as: $Q = q \delta A$ , where $\delta A$ is the area of the element boundary indicated by <b>dir</b> . If the <b>TOTAL</b> modifier is present, the source for each element is computed as: $Q = q \delta A / A$ , where $A$ is the total area of the <b>subrgn</b> in the <b>dir</b> direction.
<b>INTE</b>	By default, if <b>dir</b> points to a boundary wall, then any special treatment for the wall is deactivated and the diffusive flux at the wall is set to zero. If the <b>INTERNAL</b> modifier is present then the wall treatment and wall diffusive flux are retained.
<b>NORM</b>	In the absence of the <b>TOTAL</b> modifier, the source, $Q$ , is computed as: $Q = q \sum_i A_i \cdot V_i$ where $A_i$ is the $i^{th}$ direction component of the element boundary area specified by <b>dir</b> . $V_i$ are the values specified by <b>N<sub>n+1</sub></b> through <b>N<sub>k</sub></b> (2 for 2D, and 3 for 3D). In the presence of the <b>TOTAL</b> modifier, $Q$ is computed in a manner identical to that for the <b>AREA</b> modifier.
<b>DENS</b>	The computed source, $Q$ , is further multiplied by density. The density may be specified as the last value, <b>ρ<sub>B</sub></b> , on the command. If this value is omitted, then the boundary value at the node indicated by the <b>dir</b> direction is used if the <b>AREA</b> or <b>NORMAL</b> modifier is present, otherwise the local density for the element is used. If the <b>dir</b> modifier is present, then the boundary value of density is overwritten by the specified value.

**subrgn** The subregion for which the input is specified. If no subregion is specified, then entire computational domain is selected.

**dir** The orientation index for the element boundary associated with the source if **AREA** or **NORMAL** modifier is present. See Section 3.5 for available choices. There is no default value

for this input.

- $N_{n+1}, \dots, N_k$**  The normalizing vector components,  $V_i$ , if the **NORMAL** modifier is present. Two values must be specified for 2D and 3 for 3D flows. There are no default values for this input.
- $\rho_B$**  The density value that multiplies the computed source. It can only be specified if the **DENSITY** modifier is present and then it must be the last value on the command. If no value is specified but the modifier **DENSITY** is present, then default value is the boundary value at the node indicated by the **dir** direction if the **AREA** or **NORMAL** modifier is present, otherwise the default value is the local density for the element.

## EXAMPLES

---

Generic examples for this command are given in Section 4.4. The command keyword (**SOURCE**) must replace the keyword used in these examples. Some additional examples that illustrate the use of the attributes specific to this command are given below.

**SOURCE** for **T**: = **10** W/per unit by **VOLUME** for **SELECTED** region

**SOURCE T** is **TABLE** per unit **AREA** in **X**- direction: **3** sets (**TIME**, value) (**0.**, **0.01**), (**100**, **0.10**), (**200**, **0.20**)

**SOURCE** for **T 10** per unit **AREA** in **X**- direction for **SELECTED** region

**SOURCE** for **T 10** per unit area in **X**- direction with **NORMALIZED** of **0**, **1.5**, **2.5** **ID=VSOURCE**

**SOURCE** for **T 10** in **X**- dir with **NORMALIZED** vel **1**, **1.5**, **2.5** and **DENSITY** for **ID=VSOURCE**

**SOURCE** for **T 10 X**- dir **INTERNAL** with **NORMALIZED** vel **1**, **1.5**, **2.5** and **DENSITY** = **5** for **ID=VSOURCE**

**SOURCE** for **T 10** in per unit **VOLUME** multiply by **DENSITY** = **5** for **SELECTED** region

**MODE 2:** Flow Injection with Fixed Variable Values

**SYNTAX** SOUR {FLOW} [TOTA] [func[ξ]] {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub>|fname} [option] [subrgn] [dir] [Φ =N<sub>n+1</sub>, ..., Φ =N<sub>m</sub>] [N<sub>m+1</sub>, N<sub>k</sub>] [ρ<sub>B</sub>]

**FLOW** The source (or sink) for a variable is created due to direct fluid injection or withdrawal. The amount of the property of interest (momentum, heat, chemical species etc.) of the injected (or withdrawn) fluid acts as the source or sink for each of the relevant properties.

**TOTA** See Mode 1 specification

**func** See Mode 1 specification.

**ξ** See Mode 1 specification.

**option** See Mode 1 specification.

**subrgn** See Mode 1 specification.

**dir** See Mode 1 specification.

**fname** See Mode 1 specification.

**Φ** The symbols that denote the dependent variables injected with the fluid. If no dependent variable is specified, all properties in the injected fluid are assumed to be zero. This specification is ignored if the fluid is being withdrawn because the amount of property withdrawn is equal to the local value of that property.

**N<sub>1</sub>, ..., N<sub>n</sub>** See Mode 1 specification.

**N<sub>n+1</sub>, .. N<sub>m</sub>** Each of these represents the value of injected variable denoted by the symbol immediately preceding the value.

**N<sub>m+1</sub>, ..., N<sub>k</sub>** The normalizing vector components,  $V_i$ , if the **NORMAL** modifier is present. Two values must be specified for 2D and 3 for 3D flows. There are no default values for this input.

**ρ<sub>B</sub>** See Mode 1 specification.

**REMARKS**

The **SOURCE FLOW** command is designed to allow injection of fluid through a solid boundary (or blocked region) inside the domain of computation. Its use is most appropriate when the scale of injection is smaller than the grid size. If the scale of injection is comparable to the grid size, then the **FLOW** command may be more appropriate. The injection at an exterior domain boundary is often better accounted for by the **INLET** command.

In implementing this command it is assumed that at the point of injection, the boundary wall is essentially removed. The computational treatment is akin to that for inflow such as through an **INLET** boundary. If the injection occurs through a blocked element or at an exterior boundary of the domain, then the injected values are also assumed to be the boundary values of the variables. If the wall needs to be retained, such as for flow percolating through a solid matrix or through holes much smaller than the boundary area of the inflow element, then the **INTERNAL** modifier should be used with the command.

**EXAMPLES**

---

All examples cited for Mode 1 are applicable provided that the modifier **FLOW** is added. Some illustrative examples of the use of attributes specific to this mode are given below.

**SOUR**ce with **FLOW** injection: amount = 0.001 with T=50, C=1.

**SOUR**ce with **FLOW** per unit **AREA** of X- face: = 0.001 with T=50, C=1.

**SOUR**ce with **FLOW** withdrawal: amount =-0.001 per second

**SOUR**ce **FLOW**: **TABLE** 3 sets **SELE**cted (0, 0), (100, 1), (200, 0) U=1, V=0.1, W=0., T=10, K=0.001, L=1

**SOUR**ce for **FLOW**: **EXPO**ntial series with **TIME** 7 sets from 'SOURCE' T=100, C=0.

**SOUR**ce **FLOW** q=10 X- direction T=100, U=20. **NOR**Malized velocity 0., 1.5, 2.5 ID=VSOURce;

**SOUR**ce **FLOW** q= -10 in X- dir with **NOR**Malized vel 1., 1.5, 2.5 and **DENS**ity for ID=VSOURce

**SOUR**ce **FLOW** q=10 X- dir variable values: U=10, V=0, W=-20, T=100, K=0.03, L=0.5 **INT**ernal with **NOR**Malized vel 1., 1.5, 2.5 and **DENS**ity for ID=VSOURce:

**SOUR**ce **FLOW** q=10 per unit **VOL**ume injected variables: U=10, V=0, W=-20, T=100, K=0.03, L=0.5 multiply by **DENS**ity = 5 for **SELE**cted region

**MODE 3: Flow Injection with Fixed Variable Values and Computed Momentum Components**

**SYNTAX** **SOUR** {**MOME**} [**func**[\xi]] {**N**<sub>1</sub>, **N**<sub>2</sub>, ..., **N**<sub>n</sub>|**fname**} [**TOTA**] [**option**] [**Φ**=**N**<sub>n+1</sub>, ..., **Φ**=**N**<sub>m</sub>] [**N**<sub>m+1</sub>, **N**<sub>k</sub>] [**ρ<sub>B</sub>**] [**subrgn**] {**dir**}

**MOME** The source (or sink) for a variable is created due to direct fluid injection or withdrawal. The amount of the property of interest of the injected (or withdrawn) fluid acts as the source or sink for each of the relevant properties. In addition, the velocity components of the injected flow are computed from the source flow rate for the element, Q, as:

$$V_j = \frac{Q}{\rho_B A} n_j,$$

where  $V_j$  are the computed velocity components of the injected flow in the  $j^{\text{th}}$  direction,  $\rho_B$  is the density,  $A$  is the area of the element boundary specified by the **dir** modifier, and  $n_j$  is a normalizing vector. If the **NORMAL** modifier is present, then  $n_j$  is obtained from the user input, otherwise the area unit vector ( $A_j/A$ ; where  $A_j$  is the component in the  $j^{\text{th}}$  direction) is used as the normalizing vector. By default the density is the fluid density at the boundary, unless the user specifies a value. Any velocity input given by the user is ignored.

**func** See Mode 1 specification.

**ξ** See Mode 1 specification.

**N**<sub>1</sub>, ..., **N**<sub>n</sub> See Mode 1 specification.

**fname** See Mode 1 specification.

**TOTA** See Mode 1 specification.

**option** See Mode 1 specification. Unless the **NORMAL** modifier is present, it is assumed that the **AREA** modifier is in effect. Any specification of the **VOLUME** modifier is ignored. All other modifiers can be used as for Mode 1 Specification.

**Φ** See Mode 1 specification.

**N**<sub>n+1</sub>, .. **N**<sub>m</sub> These values represent the value of injected variable denoted by the symbol immediately preceding the value.

**N**<sub>m+1</sub>, ..., **N**<sub>k</sub> The normalizing vector components,  $V_i$ , if the **NORMAL** modifier is present. Two values must be specified for 2D and 3 for 3D flows. There are no default values for this input.

**ρ<sub>B</sub>** The density value to compute the velocity components. It can only be specified if the **DENSITY** modifier is present and then it must be the last value on the command. If no value is specified, then boundary value at the node indicated by the **dir** is used.

**subrgn** See Mode 1 specification.

**dir** See Mode 1 specification. This modifier must be present for this mode of command.

**REMARKS**

The **SOURCE MOMENTUM** command is similar to the **SOURCE FLOW** command except that the momentum (or velocity) of the incoming flow is computed on the assumption that the direction of injected mass is normal to the boundary and that the mass is uniformly distributed at the boundary. All other details of its implementation are identical to those for the **SOURCE FLOW** command.

**EXAMPLES**

---

All the examples cited for Mode 2, except those with **VOLUme** modifier, are applicable provided that the modifier **MOMEntum** is added. Some illustrative examples specific to this mode are given below.

**SOURce** with MOMEntum q=10 X- direction with ID=VSOURce; T=100, K=0.05, L=0.02

**SOURce** MOMEntum q=10 X- direction with ID=VSOURce; T=100, K=0.05, L=0.02, DENSIty 5

**SOURce** MOMEntum q=10 X- direction with ID=VSOURce; T=100, K=0.05, L=0.02, NORMAl 1. -1, 0

**SOURce** MOMEntum q=10 X- ID=VSOURce; T=100, K=0.05, L=0.02, NORMAl 1. -1, 0, DENSIty=5.

**SOURce** MOMEntum: TABLE 2 sets SELEct (0, 0), (100, 1) T=10, K=0.001, L=1

**SOURce** MOMEntum EXPOnential SERIes TIME 7 sets 'SOURCE' T=100, C=0. NORMAl 1. -1 SELEcte

**SOURce** MOMEntum q=10 X- dir INTERNAL for ID=VSOURce: injected variables: T=100, K=0.03, L=0.5  
NORMAlized vel 1., 1.5, -0.7 and DENSIty as exists

**MODE 4: Flow Injection at Fixed Spherical Angles with Computed Momentum Components**

**SYNTAX** **SOUR** {**ANGL**} [**func**{ $\xi$ }] {**N**<sub>1</sub>, **N**<sub>2</sub>, ..., **N**<sub>n</sub>|**fname**} [**TOTA**] [ $\Phi$  =**N**<sub>n+1</sub>,...,  $\Phi$  =**N**<sub>m</sub>] [**N**<sub>m+1</sub>, ..., **N**<sub>m+5</sub>] [**DENS** | **SPEE**] [ $\rho_B$  | **V**<sub>s</sub>] [**subrgn**] {**dir**}

**ANGL** Fluid is injected or withdrawn. The amount of the property of the injected (or withdrawn) fluid acts as the source or sink for each of the relevant properties. If the flow is injected, then the velocity component of the injected flow,  $U_i$  in the  $i^{th}$  direction, is either computed from the source flow rate,  $Q$ , or determined from a specified injection speed,  $V_s$ , as:

$$U_i = \frac{Q}{\rho_B A} n_i \quad \text{or} \quad U_i = v_s n_i$$

where  $\rho_B$  is the density at the neighboring element and  $A$  is the area of the element boundary specified by the **dir** modifier. The local direction vector,  $n_i$ , for each segment of the surface is obtained from the user input of two spherical angles and a reference axis.

- func** See Mode 1 specification.
- $\xi$  See Mode 1 specification.
- N**<sub>1</sub>, ..., **N**<sub>n</sub> See Mode 1 specification.
- fname** See Mode 1 specification.
- TOTA** See Mode 1 specification.
- $\Phi$  See Mode 1 specification.
- N**<sub>n+1</sub>, .. **N**<sub>m</sub> The values of injected variables denoted by the symbols immediately preceding the values.
- N**<sub>m+1</sub> The spherical angle,  $\Theta$  in degrees measured as the deflection of the injection vector from the face normal. The magnitude of the angle must be less than 180 degrees.
- N**<sub>m+2</sub> The spherical angle,  $\phi$  in degrees measured as the rotation of the injection vector with respect to the projection of the axis vector onto the face plane.
- N**<sub>m+3</sub>, **N**<sub>m+5</sub> The direction cosines of the axis vector used to measure the angle  $\phi$ ; 3 values must be specified since this option is only available for 3D flows.
- DENS** The density,  $\rho_B$ , is specified as the last value on the command.
- $\rho_B$  See Mode 3 specification.
- SPEE** The injection speed,  $V_s$ , is specified as the last value on the command.
- V**<sub>s</sub> The value  $V_s$  if the **SPEED** modifier is present. *There is no default value for this input.*
- subrgn** See Mode 1 specification.
- dir** See Mode 1 specification. *This modifier must be present for this mode of command.*

**EXAMPLES**

All examples cited for Mode 1 are applicable provided that the modifier **ANGLE** is added and appropriate input for spherical angles is appended. Some illustrative examples of the use of attributes specific to this mode are given below.

**SOUR**ce q=10, X- direction with ID=VSOURce; T=100, K=0.05, L=0.02, Theta=45, phi=20, axis=1., 0, 0  
**SOUR**ce q=10, X- ID=VSOURce; T=100, K=0.05, L=0.02, Theta=45, phi=20, axis=1., 0. 0, SPEEd=120  
**SOUR**ce q=10, X- ID=VSOURce; T=100, K=0.05, L=0.02, Theta=45, phi=20, axis=1., 0. 0, DENsity=2.5

**MODE 5: Flow Injection with Multiple Tabular Functions****SYNTAX** **SOUR** {**FLOW**} { $\Phi$ } {**TABL**} {**MULT**} [**option**] [**subrgn**] {**N<sub>sets</sub>**} {**N<sub>1</sub>**, ..., **N<sub>n</sub>**|**fname**}**FLOW** The source (or sink) for a variable is created due to direct fluid injection or withdrawal. The amount of the property of interest (momentum, heat, chemical species etc.) of the injected (or withdrawn) fluid acts as the source or sink for each of the relevant properties. $\Phi$  **One or more** of the symbols that denote the dependent variables, which are specified as functions of time. **There is no default value; at least one symbol must be specified.****TABL** The variables are specified as tabular functions of time. **This is the only available option in this mode.****MULT** Multiple variables are tabular functions of time.**option** See Mode 1 specification.**subrgn** See Mode 1 specification.**N<sub>sets</sub>** The number of sets of data for the tabular functions.**N<sub>1</sub>, ..N<sub>n</sub>** The **N<sub>sets</sub>** sets of data for the flow rate and variables as tabular functions of time. Each set must consist of time, flow rate, and one value for each variable specified by the symbol  $\Phi$  on the command, in that order. Thus if 4 variables are selected, then each data set must consist of 6 values. **There is no default value; the correct number of values must be specified.** If the flow rate is negative (withdrawal of fluid), then specified value of the property is ignored since it is assumed that the fluid is being withdrawn with an amount of property equal to the local value of that property.**fname** See Mode 1 specification.**EXAMPLES****SOUR**ce with **MULT**iple **TABL**e functions: 4 sets

Time	Flow	U	V	T
0.	0.001	1.00	0.02	100
1.	0.002	0.50	0.01	200
2.	0.004	2.00	0.01	500
5.	0.010	5.00	0.02	600

**SOUR**ce with **FLOW** **MULT**iple **TABL**e: 4 sets per unit **VOLUME**

Time	Flow	U	V	T
0.	0.001	1.00	0.02	100
1.	0.002	0.50	0.01	200
2.	0.004	2.00	0.01	500
5.	0.010	5.00	0.02	600

**MODE 6: Solubility-Limited Source for a Chemical Species****SYNTAX** **SOUR** { $\Phi$ } {**SOLU**} [**func**{ $\xi$ }] {**N**<sub>1</sub>, **N**<sub>2</sub>, ..., **N**<sub>n</sub>|**fname**} {**Q**<sub>total</sub>} [**T**<sub>start</sub>][**subrgn**] **$\Phi$**  A symbol that denotes the dependent variable for one of the chemical species. A symbol must be specified.**SOLU** The source is solubility-limited. That is, the source is specified as the total initial mass of a chemical species and the solubility of the species in the fluid phase is limited to a maximum saturation value that is defined by the rate of dissolution equation as shown below

$$S_C = f_s (C_s - C).$$

where

S is the rate of species dissolution,

f<sub>s</sub> is a dissolution frequency,C<sub>s</sub> is the solubility limit (maximum concentration) of the species in the fluid.

C the concentration of the species in the fluid

**func** The function that specifies the solubility limit, C<sub>s</sub>, for the species. **$\xi$**  See Mode 1 specification.**N<sub>1</sub>, ..., N<sub>n</sub>** See Mode 1 specification.**fname** See Mode 1 specification.**Q<sub>total</sub>** Total amount of material to be dissolved. The default value is 0.**T<sub>start</sub>** Time at which release begins. The default value is 0.**subrgn** See Mode 1 specification.**EXAMPLES**

All examples cited for Mode 1 are applicable provided that the modifier "**SOLU**" is added and the values for the variables (N<sub>n+1</sub> and N<sub>n+2</sub>) are appended at the end of the command, as necessary. Some illustrative examples of the use of attributes specific to this mode are given below.

**SOURCE** for FF: SOLUbility limited: Cs=0.05, S=100 kg, t=0 for selected zone.**SOURCE** for FU: SOLUbility: 75 sets file 'SOURCE.DAT', S=75., t\_start = 20.

**MODE 7: Radiation Type of Source For a Variable****SYNTAX** SOUR { $\Phi$ } {RADI} [COEF] [VARI] {H,  $\Phi_{eq}$ } [POSI | NEGA] [option] [subrgn] **$\Phi$**  A symbol to denote the dependent variable for which the source is specified. A symbol must be specified; otherwise, a fatal error will occur.**RADI** The radiation type of source is specified. This source is defined as:

$$Q = H (\Phi_{eq} - \Phi) .$$

In the above equation, Q is the source for  $\Phi$  in appropriate units, H is a transfer coefficient and  $\Phi_{eq}$  is an equilibrium value of  $\Phi$ . The H and  $\Phi_{eq}$  may be constants or symbols for previously defined variables.

**COEF** By default it is assumed that h is a constant (with its value specified as a numerical constant on the command). However, if this modifier, along with the **VARI** modifier, is present, then h must be a symbol for a previously defined variable.**VARI** This modifier must be present with the **COEF** modifier if H is specified as a symbol for a variable. Otherwise it is ignored.**H** The transfer coefficient, H, of the radiation source. A numerical value must be specified unless the modifiers **COEF** and **VARI** are present. In this case, a valid symbol for a previously defined variable must be specified. **$\Phi_{eq}$**  The equilibrium value for the radiation source. A numerical value or a valid symbol for a previously defined variable must be specified.**POSI** The source is implemented only if  $Q > 0$ .**NEGA** The source is implemented only if  $Q < 0$ .**option** See Mode 1 specification. **NORMAL** and **DENSITY** options are not available with this type of source**subrgn** See Mode 1 specification.**EXAMPLES****SOURCE** for T: RADlation type coefficient 0.001, equilibrium value = 15 degrees.**SOURCE** for T: RADlation hcoef = 0.001 value from variable EQVALu! EQVA is a symbol**SOURCE** for T: RADlation VARI COEF as function HVALu, value=15! HVAL is a symbol**SOURCE** for T: RADl VARI COEF HCOEf & HVALu! HCOE & EQVA are symbols

**MODE 8:** Source Term as a Linear Decay or Half-Life.

**SYNTAX** SOUR { $\Phi$ } {DECA} [HALF] [subrgn] { $\lambda$ | $\sigma$ }

$\Phi$  A symbol to denote the variable for which the source is specified.

**DECA** The decay rate of the variable is specified; it is equivalent to introducing a source term in the corresponding governing equation for  $\Phi$  as:

$$Q = -\lambda \rho \Phi,$$

In this equation, Q is the source rate for  $\Phi$ ,  $\lambda$  is the decay rate and  $\rho$  is the fluid density if the variable  $\Phi$  is defined in terms of mass units and unity if it is defined in terms of volumetric units.

**LIFE** Half-life of the property  $\Phi$  is specified. The decay rate is computed from the half life,  $\sigma$ :

$$\lambda = -\frac{\log(0.5)}{\sigma}$$

**subrgn** See Mode 1 specification.

$\lambda$  The decay rate,  $\lambda$ , for the variable  $\Phi$ .

$\sigma$  The half-life,  $\sigma$ , for the variable  $\Phi$ .

## EXAMPLES

---

**SOURce** DECAy rate for C = 0.001

**SOURce** DECAy rate for C = 0.001 in the SELEcted subregion

**SOURce** DECAy rate for C = 0.001 in the subregion ID=DRGN

**SOURce** DECAy rate for half LIFE of 29 years in subregion ID=DRGN

**MODE 9:** Source as a Linear Function of User-Defined Reaction Rates

**SYNTAX** SOUR { $\Phi$ } {REAC} {id\_1, . . . , id\_n} {N<sub>1</sub>, .., N<sub>n</sub>}[subrgn]

$\Phi$  A symbol to denote the chemical species for which the source is specified.

**REAC** Modifier specifying the source term as a linear combination of reaction rates for a given

$$Q = \sum_n C_n R_n ,$$

chemical species according to the equation:

In this equation, Q is the source for  $\Phi$ , C<sub>n</sub> are the scaling constants and R<sub>n</sub> are the previously specified reactions.

**id\_1,.., id\_n** Identifiers for the reactions, R<sub>n</sub>, described by Mode 2 of **REACTION** command, which comprise the source. Up to 9 reactions may be specified.

**N<sub>1</sub>, .., N<sub>n</sub>** The scaling constants, C<sub>n</sub>, for linear combination of the selected reaction rates. These must include the stoichiometric coefficients and the conversion factors to ensure that the source term is expressed in the proper units (kg of specie  $\Phi$  per second). The number of constants must match the number of reactions specified.

**subrgn** See Mode 1 specification.

## EXAMPLES

---

**SOUR**ce REACtion type FU LINEar -1. \* R1

**SOUR**ce REACtion type CH LINEar -1. \* R2 + 0.9586 \* R1 in SELEcted subdomain.

**SOUR**ce REACtion type CH LINEar -1. \* R2 + 0.9586 \* R1 for ID=RRGN

**MODE 10:** Flux Transfer between Adjacent Elements

**SYNTAX** SOUR {FLUX} { $\Phi$ } {VALU|DIFF|GRAD} [AREA|NORM|VOLU] {dir} [subgrn] [func[ $\xi$ ]] {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub>|fname} [N<sub>n+1</sub>, N<sub>k</sub>] [DENS] [ $\rho_B$ ]

**$\Phi$**  A symbol to denote the dependent variable for which the flux transfer is specified. A symbol must be specified.

**FLUX** The flux, Q, is transferred from a donor to a receptor element across an interface between the two elements according to the relation:

$$Q = q F_{\Phi} F_V F_{\rho}$$

Here, q is the amount specified by the user (**func ( $\xi$ )** and **N1** through **Nn**). The factors **F $\Phi$** , **F<sub>V</sub>** and **F <sub>$\rho$</sub>**  are functions of other modifiers as described below. The quantity Q is algebraically added to the source for the receptor cell and subtracted from the source term of the donor cell. **There is no net source for the system.**

**VALU** **F $\Phi$**  =  **$\Phi_0$** , where  **$\Phi_0$**  is the value of  **$\Phi$**  in the donor cell (that lies upstream with respect to direction of the computed Q). The source acts essentially as a convective flux across the interface between the donor and the receptor cells. This type of source can be used to implement processes such as settling of particulate material or droplets due to body forces. This is the default option.

**DIFF** **F $\Phi$**  =  **$\Phi_B - \Phi_F$** , where  **$\Phi_B$**  and  **$\Phi_F$**  are the values of  **$\Phi$**  in the “boundary” and “field” cells, respectively. The field cell is that defined by the **subgrn** modifier; the boundary cell is selected in reference to the field cell by the **dir** modifier. The source acts as a diffusive flux and enhances equilibrium between neighboring elements. For this type of source, the boundary cell acts as the donor cell and the field cell as the receptor cell. This option can be used to incorporate supplementary equilibrium, diffusive or radiative processes.

**GRAD** **F $\Phi$**  = ( **$\Phi_B - \Phi_F$** )/L, where L is the eulerian distance between the donor and receptor cell nodes. This type of source allows simulation of a diffusive process that depends on the gradient of the variable across an interface.

**AREA** The factor **F<sub>V</sub>** is set equal to the area of the receptor cell boundary indicated by the **dir** modifier. If the **AREA**, **NORM** and **VOLU** modifiers are all absent, then **F<sub>V</sub>** is set to unity.

**NORM** **F<sub>V</sub>** =  $\sum A_i V_i$ , where **A<sub>i</sub>** is the projected area of the interface in the direction of the **i<sup>th</sup>** coordinate denoted by **dir** modifier, and **V<sub>i</sub>** are the user inputs, **N<sub>n+1</sub>** through **N<sub>k</sub>**. The summation is over 2 terms for 2D, and 3 for 3D simulations.

**VOLU** The factor **F<sub>V</sub>** is set equal to the volume of the receptor cell.

**subgrn** See Mode 1 specification.

**func** See Mode 1 specification.

**$\xi$**  See Mode 1 specification.

**N<sub>1</sub>, ..., N<sub>n</sub>** See Mode 1 specification.

**fname** See Mode 1 specification.

**N<sub>n+1</sub>, ..., N<sub>k</sub>** See Mode 1 specification.

**DENS** **F <sub>$\rho$</sub>**  is equal to  **$\rho_B$** . In the absence of this modifier, **F <sub>$\rho$</sub>**  is equal to unity.

**$\rho_B$**  The density value that multiplies the computed source. It can only be specified if the **DENSITY** modifier is present and then it must be the last value on the command. If no value is specified but the modifier **DENSITY** is present, then default value is the boundary value at the node indicated by the **dir** direction if the **AREA** or **NORMAL** modifier is present, otherwise the default value is the local density for the element.

**EXAMPLES**

All examples cited for Mode 1 are applicable provided that the modifier **FLUX** is added. Some illustrative examples of the use of attributes specific to this mode are given below.

**SOURCE** FLUX type for T: q=0.1 for X- direction of currently active subregion.

**SOURCE** FLUX type for C1: q=1.5 multiply by AREA of X- direction interface for ID=RGN1.

**SOURCE** FLUX for C2: q=1.5 NORMAlized velocities 0, 0, 0.25 for X- direction of ID=RGN1.

**SOURCE** FLUX C2: q=1.5 NORM 0.12., 0, 0.25 multiply by DENSity; X- direction ID=RGN1.

**SOURCE** FLUX C2: DIFFerence. q=1.5 NORM 0.12., 0, 0.25 DENSity = 5; X- ID=RGN1.

**SOURCE** FLUX C2: GRADient. q=1.5 VOLUmetric DENS; X- ID=RGN1.

**MODE 11:**     Disable Previously Specified SOURce commands

**SYNTAX**     **SOUR** {OFF} {Φ} {subrgn}

**Φ**             See Mode 1 specification.

**OFF**          Previous **SOURCE** commands for the identified subregion are deactivated for the specified variable. A new specification may follow.

**subrgn**       See Mode 1 specification.

#### **EXAMPLES**

---

**SOUR**ce OFF for T for most recently SELEcted region

**SOUR**ce OFF for T for ID=MIDDLE

## 5.103 SPECIFIC HEAT

<b>PURPOSE</b>	To specify the specific heat of the fluid.
<b>MODE 1:</b>	<b>Generic Functional Form for Specific Heat</b>
<b>SYNTAX</b>	<b>SPEC {func[ξ]} {N<sub>1</sub>, N<sub>2</sub>, ..., N<sub>n</sub> fname} [phase] [subgrn]</b>
<b>func</b>	One of the modifiers listed in Table 4.2.1 that denotes the functional form of the specific heat. If no function is specified, the value is assumed to be constant.
<b>ξ</b>	One of the independent variables listed in Table 4.2.2. If no independent variable is specified, the variable is assumed to be time.
<b>N<sub>1</sub>,...,N<sub>n</sub></b>	The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.
<b>fname</b>	The name of the file containing the numerical values N <sub>2</sub> through N <sub>n</sub> . This option is available only for selected functions. See Section 3.3 for additional information.
<b>phase</b>	The phase for which the input is specified. See Section 3.6 for available options. By default the input pertains to the 1 <sup>st</sup> phase of the fluid. This modifier is available only for the multi-phase versions of the PORFLOW® and ANSWER® Software Tools.
<b>subgrn</b>	The subregion for which the input is specified. See Section 3.4. If no subregion is specified, the entire computational domain is selected.

**EXAMPLES**

**SPECIFIC** heat = 1234

Generic examples for this command are given in Section 4.4. The command keyword (**SPECIFIC**) must replace the keyword used in these examples and the dependent variable ( $\Phi$ ) must be omitted. Only the functional form, the independent variable and the numerical values need to be specified.

5.104 SPILL

**PURPOSE** To specify the particulars of an Oil Spill on Water Surface

**SYNTAX** **SPIL** [**OIL**] [**TOTA**] [**func**( $\xi$ )] {**N**<sub>1</sub>, **N**<sub>2</sub>, ..., **N**<sub>n</sub>|**fname**} [**option**] [**subrgn**] [**dir**] [**N**<sub>n+1</sub>, **N**<sub>k</sub>]

**OIL** The symbol for the dependent variable, called **OIL**, which represents the thickness of oil layer on the water surface. If this symbol is not specified, it is assumed to be present by default.

**TOTA** By default, the amount of spill volume specified, or computed from **func** ( $\xi$ ), is applied to each element of the **subrgn**. If this modifier is present then the amount is assumed to be the total amount over whole of the **subrgn**. In this case, the amount is distributed equally to all the elements of the **subrgn** unless the **VOLUME** or **AREA** modifiers are present.

**func** One of the modifiers listed in Table 4.2.1 that denotes the functional form of the source. If no function is specified, the value is assumed to be constant.

$\xi$  One of the independent variables listed in Table 4.2.2. If no independent variable is specified, the variable is assumed to be time.

**N**<sub>1</sub>,...,**N**<sub>n</sub> The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.

**fname** The name of the file containing the numerical values **N**<sub>2</sub> through **N**<sub>n</sub>. This option is available only for selected functions. See Section 3.3 for additional information.

**option** Options selected for implementation of the source.

option	INTERPRETATION
<b>VOLU</b>	If the <b>TOTAL</b> modifier is not present, the spill volume for each element is computed as: $Q = q \delta V$ with <b>q</b> is the value specified by the user and $\delta V$ is the volume of the element. The <b>q</b> , in turn, is computed from <b>func</b> ( $\xi$ ) and <b>N</b> <sub>1</sub> through <b>N</b> <sub>n</sub> . If the <b>TOTAL</b> modifier is present, the amount for each element is computed as: $Q = q \delta V / V$ , where <b>V</b> is the volume of the total <b>subrgn</b> .
<b>AREA</b>	If the <b>TOTAL</b> modifier is not present, the spill volume for each element is computed as: $Q = q \delta A$ where $\delta A$ is the area of the element boundary indicated by <b>dir</b> . If the <b>TOTAL</b> modifier is present, the amount for each element is computed as: $Q = q \delta A / A$ , where <b>A</b> is the total area of the <b>subrgn</b> in the <b>dir</b> direction.
<b>INTE</b>	By default, if <b>dir</b> points to a boundary wall, then any special treatment for the wall is deactivated and the diffusive flux at the wall is set to zero. If the <b>INTERNAL</b> modifier is present then the wall treatment and wall diffusive flux are retained.
<b>NORM</b>	In the absence of the <b>TOTAL</b> modifier, the spill volume, <b>Q</b> , is computed as: $Q = q \sum_i A_i \cdot V_i$ where $A_i$ is the $i^{th}$ direction component of the element boundary area specified by <b>dir</b> . $V_i$ are the values specified by <b>N</b> <sub>n+1</sub> through <b>N</b> <sub>k</sub> (2 for 2D, and 3 for 3D). In the presence of the <b>TOTAL</b> modifier, <b>Q</b> is computed in a manner identical to that for the <b>AREA</b> modifier.

**subrgn** The subregion for which the input is specified. If no subregion is specified, then entire computational domain is selected.

**dir** The orientation index for the element boundary associated with the source if **AREA** or **NORMAL** modifier is present. See Section 3.5 for available choices. There is no default value for this input.

**N**<sub>n+1</sub>, ..., **N**<sub>k</sub> The normalizing vector components,  $V_i$ , if the **NORMAL** modifier is present. Two values must be specified for 2D and 3 for 3D flows. There are no default values for this input.

**REMARKS**

This command activates the solution of an oil spill equation where the oil is assumed to be immiscible with the underlying water and travels as a separate layer on top of water. The current implementation assumes that the oil travels with the speed of the underlying water but can have its own diffusion, decay and other processes. These other processes must be specified through **SOURCE**, **DECAY**, **REACTION** or other commands.

The syntax of the input for the **SPILL** command is identical to that for the **SOURCE** command (Mode 1). For 2D depth-averaged mode of **TIDAL®** Software Tool, if the specified quantity is the **volume of oil (recommended option)** then the computed quality, is the thickness of oil layer. If the specified quantity is the mass of oil, then the computed quality is the thickness of oil layer multiplied by oil density., the computed variable at each element is the local thickness of the oil layer. For other software tools, the computed quantity is the mass concentration (e.g. kg/m<sup>3</sup>) if mass is specified and volume concentration (e.g. m<sup>3</sup>/m<sup>3</sup>) if volume is specified.

**EXAMPLES**

**SPILL ID=SPILL\_Region TABLE** of values **3** Sets in **TOTAL** mode for whole of the region (follow below)

<b>TIME</b>	<b>Volume(cubic meters)</b>
<b>0</b>	<b>100</b>
<b>100</b>	<b>100</b>
<b>500</b>	<b>0</b>

Generic examples for this command are given in Section 4.4. See also the **SOURCE** command (Mode 1). The command keyword (**SPILL**) must replace the keyword used in these examples.

**MODE 2:** Pre-Defined Specific Heat Function

**SYNTAX** SPEC {GORD} [fname]

**GORD** The specific heat is computed from the 4<sup>th</sup> order polynomial relation of Gordon & McBride for multi-species gases. (S. Gordon and B.J. McBride, 1971. *Computer Program for Calculation of Complex Chemical Equilibrium Compositions, Rocket Performance, Incident and Reflected Shocks and Chapman-Jouguet Detonations*, NASA SP-273)..

$$C_p = a + b T + c T^2 + d T^3 + e T^4$$

Here  $C_p$  is the specific heat of a species,  $T$  is the temperature in absolute units and  $a$ ,  $b$ ,  $c$ ,  $d$  and  $e$  are empirical constants in molar units. The enthalpy,  $h$ , is then given by:

$$h = h_0 + a T + \frac{b}{2} T^2 + \frac{c}{3} T^3 + \frac{d}{4} T^4 + \frac{e}{5} T^5$$

where  $h_0$  is the heat of formation.

**fname** The name of the file containing the numerical values to define the Gordon and McBride relations. The file format must be consistent with the format required by the Gordon-McBride specification. If no file name is then the default data is obtained from the initialization file. Any numerical input on the command is ignored.

## EXAMPLES

---

**SPEC**ific heat from GORDon-McBride formulae with default library values

**SPEC**ific heat by GORDon-McBride on file called 'CPHTDATA.JP4'

**MODE 3:** Pre-Defined Specific Heat Function

**SYNTAX** SPEC {ACRI}

**ACRI** The specific heat is computed from the built-in 4<sup>th</sup> order polynomial relations similar to Gordon & McBride for multi-species gases.

$$C_p = a + b T + c T^2 + d T^3 + e T^4$$

Here  $C_p$  is the specific heat of a species,  $T$  is the temperature in absolute units and  $a$ ,  $b$ ,  $c$ ,  $d$  and  $e$  are empirical constants in mass units. The enthalpy,  $h$ , is then given by:

$$h = h_0 + a T + \frac{b}{2} T^2 + \frac{c}{3} T^3 + \frac{d}{4} T^4 + \frac{e}{5} T^5$$

## EXAMPLES

---

**SPEC**ific heat from ACRI default library values

## 5.105 STACK

**PURPOSE** To store some information or variable on the stack for later use.

**MODE 1:** Store Location of an Element for Later Use

**SYNTAX** STAC {LOCA} [ELEM|IJ|IJK] {N<sub>1</sub>, .., N<sub>n</sub>}

**LOCA** A grid location is stored in the stack. Only a single location can be stored; this may be replaced as often as needed. Only the internal elements can be specified. The boundary nodes, If specified, are ignored.

**ELEM** The numerical input specifies the grid element number that is stored in the stack.

**IJK** The numerical input specifies the grid indices of elements (I, J) for 2D or (I, J, K) for 3D geometry. This option can be used only for structured grids.

**N<sub>1</sub>, .., N<sub>n</sub>** The element number or grid indices for a structured grid. There is no default value; appropriate input must be supplied.

**EXAMPLES**

---

**STACK LOCATION** of **ELEMENT** number **2978** for later use.

**STACK LOCATION** of IJK indices: **(2, 5)** for a two-dimensional structured grid

**STACK LOCATION** of IJK indices: **(2, 5, 2)** for three-dimensional structured grid

<b>MODE 2:</b>	<b>Store A Local Grid Transformation for Later Use</b>
<b>SYNTAX</b>	<b>STAC {TRAN} [MATR PLAN POIN] {N<sub>1</sub>, .., N<sub>n</sub>} {N<sub>n+1</sub>, .., N<sub>m</sub>} [N<sub>7</sub>, .., N<sub>9</sub>]</b>
<b>TRAN</b>	A local grid transformation is stored for later use. Only a single transformation can be stored; this may be replaced as often as needed. The transformation may be later used by specific <b>FREEFORM™</b> commands that allow the use of the <b>STACK</b> modifier.
<b>MATR</b>	The 9-component transformation matrix ( <b>T</b> ) between the local ( <b>x</b> ) and the global ( <b>X</b> ) coordinate systems is specified directly. The transformation maps the global system to the local system as $\mathbf{x}_i = \mathbf{T}_{ji} \mathbf{X}_j$ . The index <b>j</b> varies the fastest. Each row of the transformation matrix is interpreted as a unit vector directed along the <b>x<sub>i</sub></b> direction. This is the default option.
<b>PLAN</b>	The plane of transformation is specified by the coordinates of a point on the plane and the direction cosines normal to the plane. The transformation is then computed by taking <b>x<sub>1</sub></b> to be the unit normal to the plane, <b>x<sub>2</sub></b> is computed by intersecting the plane with the bounding box of the domain, and <b>x<sub>3</sub></b> is computed as the cross product of <b>x<sub>1</sub></b> and <b>x<sub>2</sub></b> . The user has no control over the directions of <b>x<sub>1</sub></b> and <b>x<sub>2</sub></b> . However, they are guaranteed to be in the specified plane and mutually perpendicular.
<b>POIN</b>	The plane of transformation is specified by the coordinates of 2 points. The plane is assumed to pass through the 2 <sup>nd</sup> point and the normal points from the 1 <sup>st</sup> to the 2 <sup>nd</sup> point. The transformation is computed in the same way as the previous option.
<b>N<sub>1</sub>, .., N<sub>n</sub></b>	The 1 <sup>st</sup> through 3 <sup>rd</sup> components of the transformation matrix if the <b>MATRIX</b> modifier is present, the (x, y; for 2D) or (x, y, z; for 3D) coordinates of the point on the plane of transformation if the <b>PLANE</b> modifier is present and the coordinates of the 1 <sup>st</sup> point if the <b>POINT</b> modifier is present. There is no default value.
<b>N<sub>n+1</sub>, .., N<sub>m</sub></b>	The 4 <sup>th</sup> through 6 <sup>th</sup> components of the transformation matrix if the <b>MATRIX</b> modifier is present, a vector (or direction cosines) to specify the normal to the plane of transformation if the <b>PLANE</b> modifier is present and the coordinates of the 2 <sup>nd</sup> point (on the plane of transformation) if the <b>POINT</b> modifier is present. There is no default value.
<b>N<sub>7</sub>, .., N<sub>9</sub></b>	The 7 <sup>th</sup> through 9 <sup>th</sup> components of the transformation matrix if the <b>MATRIX</b> modifier is present; otherwise this input is ignored. There is no default value.

## EXAMPLES

**STACK TRANSFORMATION** between local & global grid is:

```
1    0    0
0    0   -1
0    1    0
```

**STACK TRANSFORMATION** between local & global grid is:

```
1.0      0.0      0.0
0.0      0.9950040  -0.0998334
0.0      0.0998334   0.9950040
```

**STACK TRANSFORMATION** for **PLANE** at (0.12, 0.55) normal vector as: (1, -1)

**STACK TRANSFORMATION** for **PLANE** at (0.12, 0.55, 0.012) normal vector as: (1, -1, 0)

**STACK TRANSFORMATION** between **POINTS** (0.12, 0.55, 0.012) (0.15, 0.47, 0.012)

**MODE 3:** Store Scaling Factors to Normalize Output of Variables

**SYNTAX** STAC {WRIT} {Φ|OFF} {a} [b]

**WRIT** The output produced by the **WRITE** command for the specified variable is scaled by **a** and **b** according to:

$$\Phi_{\text{out}} = a \Phi + b$$

**Φ** A symbol that denotes the variable for which the output is required. The valid symbols include those listed in Table 2.8.1-3 plus the user defined variables (**ALLOCATE** command) and the real type variables defined in the initialization file (xxxINIT.ACR). There is no default value; a symbol must be specified.

**OFF** Any previous **STACK WRITE** commands are deactivated; new commands may follow.

**a** The multiplier factor for the variable. There is no default value.

**b** The additive constant for scaling the variable. The default value is 0.

#### REMARKS

---

Up to 10 **STACK WRITE** commands each with its own **Φ** may be active at any given time.

#### EXAMPLES

---

**STACK WRITE T** by scaling with 1.8 and add 32 to convert deg Celsius to Fahrenheit.

**STACK WRITE P** by scaling with 1.00E5 and add 0.

## 5.106 STATISTICS

<b>PURPOSE</b>	To compute and obtain output of the statistics for a dependent variable for a selected subregion.
<b>SYNTAX</b>	<b>STAT</b> { $\Phi$ } [ <b>MAXI</b> ] [ $\Psi_1, \Psi_2, \dots, \Psi_N$ ] [ <b>subrgn</b> ] [ <b>OFF</b> ] [ <b>fname</b> ] [ <b>TABL</b> ] [ <b>V<sub>frq</sub></b> ] [ <b>TIME</b> ]
<b><math>\Phi</math></b>	<b>One and only one</b> symbol for the dependent variable for which the statistics are required. By default, the computed statistics consist of the minimum, maximum, mean and standard deviation, and the location of the minimum and the maximum. If the modifier <b>MAXI</b> is present then only the maximum and its location is computed. The valid symbols are listed in Table 2.8.1-3. <b>One and only one character string must be specified for each command.</b>
<b>MAXI</b>	If this modifier is present, then only the maximum value and its location is computed.
<b><math>\Psi_1, \Psi_2, \dots, \Psi_N</math></b>	Up to 10 accompanying variables whose values at the location of the maximum value of $\Phi$ are also printed to the file. There is no default value.
<b>subrgn</b>	The subregion for computation of statistics. <b>If no subregion is specified, the entire domain is selected.</b>
<b>OFF</b>	Computation of statistics for the specified variable and subregion is deactivated.
<b>fname</b>	The file name for output. <b>A different output file may be specified for each command. If no file name is specified then the output is directed to the most recent previously specified file. If no previous file was specified, then the output is directed to a file with the same name as the Standard Output file but with the extension removed and replaced by “_STATS.TMP”. For example, if the Standard Output file is “PROBLEM.OUT”, then the file is named “PROBLEM_STATS.TMP”. A summary of output is also printed to the standard output device at the end of simulations.</b>
<b>TABL</b>	If this modifier is specified, then a tabulated summary of the computed statistics appears in the standard output file at the end of simulations.
<b>V<sub>frq</sub></b>	The frequency (step or time interval) of output. See Section 3.7 for further details. <b>The default value is set so that output is obtained at every step of simulations.</b>
<b>TIME</b>	<b>By default, V<sub>frq</sub> is interpreted to be the frequency of computations in terms of number of steps. If this modifier is present, then V<sub>frq</sub> is interpreted to be the time interval between successive computations.</b>

**REMARKS**

The user should perform these computations only at the required frequency since frequent computations will result in increased computation time.

**EXAMPLES**

**STAT**istics for U for the entire domain

**STAT**istics for T **MAXI**imum for SELEcted region every 20 steps **!Compute only maximum**

**STAT**istics for T for SELEcted region every 20 steps with TABL**E**s of output

**STAT**istics for T for subregion defined by ID=VAUL**t** every TIME=0.50 units

**STAT**istics for T for ID=VAUL**t** OFF **!!!** Switch off previously specified STAT command

**STAT**istics for U to 'FLUX.OUT' for SELEcted region at TIME interval of 0.4

**STAT**istics for U for the entire domain along with V W P T **! V W P T are accompanying variables**

## 5.107 STORAGE COEFFICIENT

**PURPOSE** To specify the Storage coefficient for the governing differential equations. This command is effective only for the **PORFLOW®** Software Tool.

**MODE 1:** **STORAGE Coefficient as a General Function**

**SYNTAX** **STOR** { $\Phi$ } {func[ $\xi$ ]} { $N_1, N_2, \dots, N_n$ |fname} [**TOTA**] [**subrgn**]

$\Phi$  A symbol for the dependent variable for which the storage coefficient is specified. Valid symbols are listed in Table 2.7.1. There is no default value; a symbol must be specified. The typical storage or accumulation term for a transport equation is represented as:

$$\frac{\partial}{\partial t} (\alpha \rho \Phi)$$

where  $\alpha$  is the storage coefficient,  $\rho$  is the fluid density and  $\Phi$  is dependent variable. For species equations, the storage coefficient is more commonly known as the “retardation” coefficient. See **REMARKS** below.

**func** One of the modifiers listed in Table 4.2.1 that denotes the functional form of the input. For this input, the function specifies the value of the appropriate storage coefficient for the corresponding  $\Phi$  variable. If no function is specified then the value is assumed to be constant.

$\xi$  One of the independent variables listed in Table 4.2.2. If no independent variable is specified, the variable is assumed to be time.

$N_1, \dots, N_n$  The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.

**fname** The name of the file containing the numerical values  $N_2$  through  $N_n$ . This option is available only for selected functions. See Section 3.3 for additional information.

**TOTA** This modifier is effective only for the **PORFLOW®** Software Tool. By default it is assumed that any chemical species is adsorbed only in the wetted part to the solid; In this case the effective retardation coefficient ( $R_D$ ) is defined by Equation (6) below.

However if the **TOTA** modifier is present, then the species is assumed to be stored in the total solid and the effective retardation coefficient is defined by Equation (7) below:

**subrgn** The subregion for which the input is specified. See Section 3.4. If no subregion is specified, the entire computational domain is selected.

**REMARKS**

The term “storage coefficient” is normally applied only to the transport of chemical species. However, this command may be used to specify the “storage coefficient” for any of the governing transport equations. **This mode of the command is currently implemented only for the PORFLOW® Software Tool and then only for the pressure and species equations.**

In a fluid-solid matrix the total mass of a species per unit volume is given by:

$$C_e^k = \theta_E S^n C^k + (1 - \theta_E) S_s \rho_s C_s^k. \quad (1)$$

In this equation the superscript k denotes the k<sup>th</sup> component of a group of species and:

- $C_e$  is the equivalent concentration of the species in units of [M/L<sup>3</sup>] (e.g.kg/m<sup>3</sup>),
- $C$  is the concentration of the species in the fluid phase in units of [M/L<sup>3</sup>] (e.g.kg/ m<sup>3</sup>),
- $C_s$  is the concentration of the species in the solid phase in units of [M/M] (e.g.kg/kg of solid),
- $\theta_E$  is the effective porosity of the host matrix,
- $S$  is the phase saturation,
- $S_s$  is fraction of solid that contains the species; it is equal to 1 if species is present throughout the solid and is equal to  $S^n$  if only the ‘wet’ component of the solid contains the species,
- $\rho_s$  is the particle (not bulk) density of the solid, and

The **PORFLOW®** provide for non-linear equilibrium between phases as well as explicit solution of the concentration of each chemical species in different fluid and solid phases with full accounting of phase exchange and geochemistry of species (see **PRECIPITATION** and **TRANSPORT** commands). However, a common assumption in environmental applications is that a chemical equilibrium exists between the fluid and the solid constituents of the porous matrix. The concentrations in the fluid and the solid are then related through a linear or non-linear isotherm (Freeze and Cherry 1979). With  $k_d$  as the sorption or partitioning coefficient for the species, the concentrations in the solid and the fluid are related by:

$$C_s^k = k_d^k C^k. \quad (2)$$

In this equation,  $C$  is in units of  $[M/L^3]$ ,  $C_s$  in units of  $[M/M]$  and  $k_d$  in units of  $[L^3/M]$ . Consequently Equations 1 and 2 combine to yield:

$$\begin{aligned} C_e^k &= \varphi_E \left( 1 + \frac{S_s}{S^n} \frac{1 - \theta_E}{\theta_E} \rho_s k_d^k \right) C^k. \\ &= \varphi_E R_D^K C^k \end{aligned} \quad (3)$$

The saturated porosity,  $\varphi_E$ , and the retardation coefficient,  $R_D$ , are defined by:

$$\varphi_E = S^n \theta_E. \quad (4)$$

$$R_D^k = 1 + \frac{S_s}{S^n} \frac{1 - \theta_E}{\theta_E} \rho_s k_d^k. \quad (5)$$

**PORFLOW®** provides for  $S_s$  to be set equal to fluid saturation,  $S^n$  or to unity. The default assumption is the former and then saturation drops out of the retardation relation. In this case:

$$R_D^k = 1 + \frac{1 - \theta_E}{\theta_E} \rho_s k_d^k. \quad (6)$$

If it is assumed that  $S_s=1$ , then it implies that some mechanism is available for the chemical species to move into the dry solid which is not connected by effective porosity. It further implies that even if saturation falls to zero, some concentration may remain in the dry solid matrix. In this limiting case, there is no reason for the solid concentration to be related to a hypothetical fluid concentration and the basic isotherm assumption that the species may move freely between the fluid and the solid phases is no longer valid. This assumption then yields:

$$R_D^k = 1 + \frac{1 - \theta_E}{\varphi_E} \rho_s k_d^k. \quad (7)$$

## EXAMPLES

---

Generic examples for this command are given in Section 4.4. The command keyword (**STORAGE**) must replace the keyword used in these examples.

**MODE 2:** Dynamic Computation of Storage or Accumulation Term

**SYNTAX** STOR {Φ} {DYNA}

**Φ** One or more symbols for the dependent variables for which the input is specified. Valid symbols are listed in Table 2.7.1. There is no default value; at least one symbol must be specified.

**DYNA** A typical storage term,  $S_{\Phi}$ , for a transport equation and its numerical representation is given by the relation:

$$S_{\Phi} = \frac{\partial}{\partial t} (\alpha \rho \Phi) = \frac{(\alpha \rho \Phi)^{n+1} - (\alpha \rho \Phi)^n}{\delta t},$$

where superscripts  $n$  and  $n+1$  denote time steps. At the start of a step the total storage for the variable may be computed from the current available values of  $\alpha$ ,  $\rho$  and  $\Phi$  or the product  $(\alpha \rho \Phi)$  can be stored at the end of  $n^{\text{th}}$  time step and re-used at the  $n+1^{\text{th}}$  step. When  $\alpha$  and  $\rho$  are constants, the two procedures lead to identical results. However, if  $\alpha$  and  $\rho$  are functions of other variables, then depending on the method and sequence of solution of equations,  $\alpha$  and  $\rho$  may be revised after  $\Phi^n$  was computed. That is the prevailing values of  $\alpha$  and  $\rho$  may have been used to compute  $\Phi^n$  at the  $n^{\text{th}}$  time step and then modified by subsequent operations. For strict compliance with balance equations the product of  $(\alpha \rho \Phi)$  used at the  $n^{\text{th}}$  time step is stored and used in computing the values at the  $n+1$  step. This is the default practice in ACRi software tools.

If **DYNA** modifier is present, then the required product  $(\alpha \rho \Phi)^n$  is recomputed from the most recent values of the variables at the start of the  $n+1^{\text{th}}$  step. This removes the need for storage arrays to store the previous product and also accounts for dynamic changes in  $\alpha$  and  $\rho$ . However, the strict mass balance of the  $\Phi$  variables may not be achieved if  $\alpha$  and  $\rho$  change significantly from one step to another.

## EXAMPLES

---

**STOR** term for C computed DYNAmically

**STOR** term for C, K and E computed DYNAmically

**MODE 3:** Deactivate or Reactivate Storage or Accumulation Term

**SYNTAX** STOR { $\Phi$ } {OFF|ON}

**$\Phi$**  One or more symbols for the dependent variables for which the input is specified. Valid symbols are listed in Table 2.7.1. There is no default value; at least one symbol must be specified.

**OFF** The storage or accumulation term in the governing equation for the variable is deactivated. This is equivalent to solving the steady state form of the equation.

**ON** Storage term, if previously deactivated, if re-activated.

#### REMARKS

---

This command may be used to explore the effect of the storage on the solution of the transport equation for a selected variable. At any given time at least one of the three (storage, convection and diffusion) terms in the transport equation must stay active, otherwise the transport equation has no solution.

#### EXAMPLES

---

**STOR** term for C is OFF

**STOR** term for C is now ON again

**MODE 4:** Minimize Memory Allocation for Storage or Accumulation Term

**SYNTAX** STOR {MINI|MIN}

**MINI** A typical storage term for a transport equation, S, is given by the relation:

$$S_{\Phi} = \frac{\partial}{\partial t}(\alpha \rho \Phi) = \frac{(\alpha \rho \Phi)^{n+1} - (\alpha \rho \Phi)^n}{\delta t},$$

By default if  $\alpha$  and  $\rho$  vary with time, then product  $(\alpha \rho \Phi)$  must be stored at the end of  $n^{\text{th}}$  time step for strict mass balance of the variable. This requires the allocation of memory to store the product for each variable. This is the default practice in ACRi software tools.

However if **MINI** modifier is present, then the required product  $(\alpha \rho \Phi)^n$  is recomputed from the most recent current values of the variables at the start of the  $n+1^{\text{th}}$  step.

**MIN** Same as **MINI**.

## EXAMPLES

---

**STOR** term allocate MINimum storage

## 5.108 SWIRL

**PURPOSE** To specify the transformation of the velocity components from an induced swirl. This command is effective only for the ANSWER® Software Tool.

**MODE 1:** Swirl induced by Vanes at Fixed Angles or by Solid Body Rotation

**SYNTAX** **SWIR** {**ANGL|SOLI**} [**LOCA|FLUX|SOUR**] {**N<sub>1</sub>**} {**N<sub>2</sub>, ..,N<sub>n</sub>**} [**R<sub>swirler</sub>**] [**subrgn**] [**dir**]

**ANGL** Swirl is induced in the manner of the flow passing vanes at a fixed angle. With  $\alpha$  as the vane angle, the tangential (swirling) velocity,  $V_T$ , at the swirl plane is computed as:

$$V_T = V_N \tan(\alpha),$$

where  $V_N$  is the normal component of velocity at the surface.

**SOLI** Swirl is induced in the manner of solid body rotation. With  $\omega$  as the angular velocity, the tangential velocity (swirl component),  $V_T$ , at the swirl plane is computed as:

$$V_T = 2 \pi \omega r,$$

where  $r$  is the radius from the specified center of rotation.

**LOCA** The normal component of velocity at the swirl plane,  $V_N$ , is computed as:

$$V_N = \sum_i n_i U_i$$

where  $n_i$  is the normal vector and  $U_i$ , the velocity vector just upstream of the swirl plane. This is the default option if none is specified.

**FLUX** The normal component of velocity at the swirl plane,  $V_N$ , is computed as:

$$V_N = \frac{q}{\rho A}$$

where  $\rho$  is the fluid density,  $q$  is the flow rate and  $A$  is the area of the surface.

**SOUR** The normal component of velocity at the swirl plane,  $V_N$ , is computed as:

$$V_N = \frac{q}{\rho A}$$

where  $\rho$  is the fluid density,  $q$  is the source and  $A$  is the area of the surface. If this modifier is present, then a TRANSFER command with SOURCE modifier must have been previously specified for the same subrgn.

**subrgn** The subregion that defines the surface at which the velocity transformation occurs. If the subregion was previously defined by a LOCATE MATCH command, then the transformation is applied only to the 2<sup>nd</sup> surface. A subregion must be defined; there is no default value.

**dir** The orientation index for the element boundary associated with the input. See Section 3.5 for available choices. There is no default value for this input. A value must be supplied unless the subrgn was defined by a LOCATE PAIR or LOCATE MATCH command.

**N<sub>1</sub>** The vane angle,  $\alpha$ , in degrees if the ANGLE modifier is specified or the angular velocity of rotation,  $\omega$ , if the SOLID modifier is specified. The numerical value of  $\alpha$  must be less than 89.9. There is no default value for this input.

**N<sub>2</sub>, N<sub>n</sub>** The (x, y) or (x, y, z) coordinates of the center of the swirl around which the velocity vector is rotated. The center must lie in the plane of the surface defined subrgn and dir. The transformed tangential velocity is assumed to lie in this plane. Two values are needed for 2D and 3 for 3D geometry. There is no default value for this input.

**R<sub>swirler</sub>** The radius of the swirler. This is radius of the circle (in the plane of the surface) within which the velocity components are transformed. Any velocity components in the surface outside this radius are not transformed. The default value 10<sup>30</sup>.

## REMARKS

---

The local Cartesian components of velocity at the swirl plane are computed from the normal and tangential velocity components by transforming from cylindrical to Cartesian components. The global Cartesian components of velocity are obtained by transforming the local components as:

$$U_j = T_{ij} u_i$$

where  $T_{ij}$  is the transformation vector and  $u_i$  is the velocity vector.

**The transformation is applied only if the computed  $V_N > 0$ ; otherwise the command is ignored. The computed Cartesian velocity components are set at a location just upstream of the swirl surface. A BLOCK or FIX command must be used to ensure that these values are not overwritten by values computed from the governing momentum equations.**

## EXAMPLES

---

**SWIRL** velocity with a vane ANGLE of 50 degrees center at (0.14, 0.22, 0.0) ID=Q\_TRANSFER  
**SWIRL** vane ANGLE = 50 degrees VN from LOCAL velocity; center at (0.14, 0.22, 0.0) ID=Q\_TRANSFER  
**SWIRL** vane ANGLE = 50 degrees VN from FLUX; center at (0.14, 0.22, 0.0) ID=Q\_TRANSFER  
**SWIRL** vane ANGLE = 50 degrees VN from SOURce; center at (0.14, 0.22, 0.0) ID=Q\_TRANSFER  
**SWIRL** velocity ANGLE of 50 deg. Center: (0.14, 0.22, 0.0) within radius of 0.25 meters ID=Q\_TRANSFER  
**SWIRL** with angular speed of 200 radian/second center at (0.14, 0.22, 0.0) ID=Q\_TRANSFER  
**SWIRL** with 200 radians/second; center: (0.14, 0.22, 0.0) within radius of 0.25 meters ID=Q\_TRANSFER

**MODE 2:** Swirl from User Specified Value of Function

**SYNTAX** **SWIR** [LOCA|FLUX|SOUR] [subrgn] [dir] {N<sub>1</sub>} [N<sub>2</sub>, ..,N<sub>n</sub>|fname] [N<sub>n+1</sub>,...,N<sub>m</sub>] [R<sub>swirler</sub>]

**LOCA** See Mode 1 Specification.

**FLUX** See Mode 1 Specification.

**SOUR** See Mode 1 Specification.

**subrgn** See Mode 1 Specification.

**dir** See Mode 1 Specification.

**N<sub>1</sub>, ..., N<sub>n</sub>** The numerical constants and coefficients for the selected function. See Section 4.4 for more details. **There are no default values for this input.**

**fname** The name of the file from which numerical values **N<sub>2</sub>** through **N<sub>n</sub>** are read. This option can be used only if the selected function is a table or one of the series functions. See Section 3.3 for further information.

**N<sub>n+1</sub>, ...,N<sub>m</sub>** The (x, y) or (x, y, z) coordinates of the center of the swirl around which the velocity vector is rotated. The center must lie in the plane of the surface defined **subrgn** and **dir**. The transformed tangential velocity is assumed to lie in this plane. Two values are needed for 2D and 3 for 3D geometry. **There is no default value for this input.**

**R<sub>swirler</sub>** The radius of the swirler. This is radius of the circle (in the plane of the surface) within which the velocity components are transformed. Any velocity components in the surface outside this radius are not transformed. **The default value 10<sup>30</sup>.**

## EXAMPLES

Generic examples for this command are given in Section 4.4. The command keyword (**SWIRL**) must replace the keyword and the dependent variable ( $\varphi$ ) used in these examples. Some additional examples that illustrate the use of the attributes specific to this command are given below.

**SWIRL** tangential velocity 0.25; center at (0.14, 0.22, 0.0) ID=SWIRL

**SWIRL** tangential velocity 0.25; normal from LOCAL; at (0.14, 0.22, 0.0) ID=SWIRL

**SWIRL** tangential velocity 0.25; normal from FLUX; at (0.14, 0.22, 0.0) ID=SWIRL

**SWIRL** tangential velocity 0.25; normal from SOUR; at (0.14, 0.22, 0.0) ID=SWIRL

**SWIRL** factor of 0.25; center: (0.14, 0.22, 0.0) within radius of 0.25 meters ID=SWIRL

<b>MODE 3:</b>	<b>Swirl Profile or Mapping from a data file</b>
<b>SYNTAX</b>	<b>SWIR</b> [PROFile] [RADial] {subrgn} {dir} {N <sub>1</sub> ,N <sub>2</sub> ,N <sub>3</sub> ,N <sub>4</sub> } {fname} [SCALE [NONE] N <sub>5</sub> ,N <sub>6</sub> ] [AXIS N <sub>7</sub> ,N <sub>8</sub> ,N <sub>9</sub> ] [ALWA] [FUEL=N <sub>10</sub> ] [DIAG] (Φ <sub>1</sub> .. Φ <sub>n</sub> )
<b>PROF</b>	Implies that a profile of the velocity components and other specified variables be read in from a data file.
<b>RADI</b>	Implies a radial swirler. The default is an axial swirler.  <i>For an <u>axial</u> swirler, the axial velocity component is scaled to match the mass flow.</i> The other velocity components are then scaled by the same factor. If the turbulent kinetic energy is listed as one of the dependent variables on this command, then it is scaled by the square of the scale factor. <i>The velocity profile data is assumed to be a function of the radius.</i>  <i>For a <u>radial</u> swirler, the radial velocity component is scaled to match the mass flow.</i> The other velocity components are scaled by the same factor. If the turbulent kinetic energy is listed as one of the dependent variables on this command, then it is scaled by the square of the scale factor. <i>The velocity profile data is assumed to be a function of the axial distance.</i>
<b>subrgn</b>	The subregion that defines the surface at which the velocity transformation occurs and also defined to be an inlet or open or dirichlet boundary. A subregion must be defined; there is no default value.
<b>dir</b>	The orientation index for the element boundary associated with the input. See Section 3.5 for available choices. There is no default value for this input. A value must be supplied unless the <b>subrgn</b> was defined by a <b>LOCATE PAIR</b>
<b>N<sub>1</sub></b>	<b>The mass flow through the subregion.</b> This subregion should be defined as an INLET or an OPEN or DIRICHLET boundary. There are no default values for this input.
<b>N<sub>2</sub>, N<sub>3</sub>, N<sub>4</sub></b>	The (X, Y, Z) coordinates of the origin of the local cylindrical system. This input is mandatory for both axial and radial swirlers. There are no default values for this input.
<b>fname</b>	<b>The name of the file from which profile data are read in.</b> This file is mandatory. The file consists of an <b>optional</b> header block ending with a “ <b>End Header</b> ” string starting from the first column, followed by columns of data, one per variable. The first four data columns are the <i>independent variable</i> (axial distance for radial swirler / radial distance for axial swirler), <i>axial velocity</i> , <i>radial velocity</i> and the <i>swirl velocity</i> . It may also contain additional columns of data for the other variables specified on the command (in the same order). Extra columns of data are ignored.
<b>SCALE</b>	Scale factors are specified to normalize the transformed coordinate prior to interpolating the dependent variables from the profile data file. The min and max values of the independent variable should be specified. <i>In the absence of this modifier, the min value is set to <u>zero</u> and the max value to the actual <u>maximum radial extent</u> of the sub-region for an <u>axial swirler</u>, and to the actual min and max values of the <u>axial extent</u> of the sub-region for a <u>radial swirler</u>.</i> The independent variable is thus scaled to lie between 0 and 1.
<b>SCALE NONE</b>	Turns off all scaling. The independent variable is not scaled prior to doing the interpolation.
<b>N<sub>5</sub>, N<sub>6</sub></b>	The min and max scale factors for the transformed coordinate for the <b>SCALE</b> modifier. The coordinate is transformed as $r = (R - N_5) / (N_6 - N_5)$ . <b>Must be specified if the SCALE modifier is present without the NONE modifier.</b> In the absence of the <b>SCALE</b> modifier, these values are computed internally as described above.
<b>AXIS</b>	The axis of the transformation between the global 3D Cartesian system and the local cylindrical system applicable to the defined subregion. <b>For an axial swirler, this modifier and the associated data may be omitted</b> , in which case it is computed as the normal pointing into the mesh from a suitable boundary node. <b>For a radial swirler this input and the associated data is mandatory.</b>
<b>N<sub>7</sub>, N<sub>8</sub>, N<sub>9</sub></b>	The three components of direction vector of the axis of the local cylindrical coordinate system. <i>There are no default values for a radial swirler.</i> For an axial swirler, in the absence of the <b>AXIS</b> modifier, these are computed as the normal pointing into the mesh from a suitable boundary node.
<b>FUEL</b>	This optional input specifies a fuel mass flow rate. The presence of this modifier causes the

	Fuel Mass Fraction, FU, to be scaled to match the specified fuel mass flow rate.
<b>N<sub>10</sub></b>	The fuel mass flow rate if the <b>FUEL</b> modifier is specified. There is no default.
<b>Φ<sub>1..n</sub></b>	Optional list of scalar variables (except pressure) also to be interpolated from the profile data.
<b>ALWA</b>	This modifier causes the command to be executed at every iteration/time step. The default is to execute only once in the startup phase of the next <b>SOLVE</b> command. This behavior is analogous to the <b>SET</b> vs. <b>SET ALWAYS</b> command. This modifier may be used if the density is a function of the dependent variables and an iterative procedure is needed to determine the final boundary conditions.
<b>DIAG</b>	This modifier causes detailed diagnostics to be output. In the absence of the <b>DIAGnostics</b> modifier, only a brief summary is output for each swirler.

## REMARKS

---

This command only applies to a 3D Cartesian framework. A global 3D Cartesian to local cylindrical transformation is computed from the center and the axis. Any requested scaling is carried out. The velocity components and other variables are then interpolated from the profile data. The velocity components are then scaled to match the specified mass flux. These velocity components and other variables are then applied to the boundary specified by the sub-region.

If the turbulent kinetic energy, **K**, is also present on the command line, then it is also scaled as the square of the scaling for the velocity variables. However, the turbulent length scale, **L**, is not scaled.

If the fuel mass flow rate is specified, the fuel mass fraction, **FU**, is adjusted to match the specification.

A positive mass flow specification causes the flow to enter the domain whereas a negative mass flow specification causes the flow to leave the domain. If the data in the velocity profile file is not consistent with the mass flow specification, then a negative scale factor is computed. This has the effect of inverting the profile and reversing the sense of the swirl component.

EXAMPLES

**SWIRler** PROFILE mdot=50.0 kg/s center= (0.4, 0.5, 1.0)  
 FUEL mdotf=5 kg/s file='swirl3.dat'  
 additional variables in file: T K L FU at ID=RGN3 Z+ **ALWA**ys detailed DIAGnostics on.

**SWIRler** PROFILE mdot=10.0 kg/s center=(1.0,0.5,0.5)  
 SCALing (3.53553E-02, 0.1060660) file='swirl3.dat'  
 additional variables in file: T K L at ID=RGN2

The contents of file 'swirl3.dat':

Swirler mapping file for axial swirler:

r	u	Ur	Ut	T	K	L	FU
0	-0.1	0	0.1	300	0.001	0.1	0.01
0.1	-0.2	0	0.2	350	0.002	0.15	0.02
0.2	-0.3	0	0.3	400	0.003	0.2	0.03
0.3	-0.4	0	0.4	450	0.004	0.25	0.04
0.4	0.0	0	0.5	500	0.0001	0.3	0.05
0.5	0.6	0	0.6	550	0.006	0.35	0.06
0.6	0.7	0	0.7	600	0.007	0.4	0.05
0.7	0.8	0	0.8	400	0.008	0.45	0.04
0.8	0.9	0	0.9	300	0.009	0.5	0.03
0.9	1.0	0	1.0	300	0.01	0.55	0.02
1.0	1.0	0	0.0	300	0.01	0.6	0.01

**SWIRler** PROFILE RADial mdot=50.0 kg/s center=(0.5,0.5,0.5)  
 AXIS=(1,0,0) file='swirl2-r.dat'  
 additional variables in file: T K L at ID=RGN1

**SWIRler** PROFILE RADial mdot=50.0 kg/s center=(0.5,0.5,0.5) AXIS=(1,0,0)  
 FUEL mdotf=5 kg/s file='swirl3-r.dat'  
 additional variables in file: T K L FU at ID=RGN1 **ALWA**ys detailed DIAGnostics on.

The contents of file 'swirl3-r.dat':

Swirler mapping file for radial swirler:

X	U	Ur	Ut	T	K	L	FU
0	-0.5	-0.0	0.0	300	0.005	0.1	0.01
0.1	-0.4	-0.1	0.1	350	0.004	0.15	0.02
0.2	-0.3	-0.2	0.2	400	0.003	0.2	0.03
0.3	-0.2	-0.3	0.3	450	0.002	0.25	0.04
0.4	-0.1	-0.4	0.4	500	0.001	0.3	0.05
0.5	0.0	-0.5	0.5	550	0.0001	0.35	0.06
0.6	0.1	-0.4	0.4	600	0.001	0.4	0.05
0.7	0.2	-0.3	0.3	650	0.002	0.45	0.04
0.8	0.3	-0.2	0.2	700	0.003	0.5	0.03
0.9	0.4	-0.1	0.1	750	0.004	0.55	0.02
1.0	0.5	-0.0	0.0	800	0.005	0.6	0.01

**5.109 SYMMETRY**

**PURPOSE** To identify an external boundary of the computational domain as a symmetry plane or axis.

**SYNTAX** **SYMM** {**dir**} [**subrgn**] [**OFF**]

**dir** The orientation index for the symmetry boundary. See Section 3.5 for available choices. There is no default value; a value must be specified.

**subrgn** The subregion to be identified as a symmetry boundary. If no subregion is specified, the outermost "**dir**" oriented boundary of the entire computational domain is selected.

**OFF** Previously specified symmetry boundary for the identified subregion is deactivated. A new specification for this subregion may follow.

**REMARKS**

---

A symmetry boundary, by definition, is assumed to be a boundary where the normal fluxes for [all](#) dependent variables are zero; that is no amount of the property in question leaves or enters the symmetry boundary.

**EXAMPLES**

---

**SYMM**etry at Y- boundary

**SYMM**etry at Y- for ID=B\_SYMMETRY

**SYMM**etry OFF for Y- and ID=B\_SYMMETRY

## 5.110 THERMAL

**PURPOSE** To specify thermal properties of the porous matrix, or of the planar or linear features. This command is effective only for the **PORFLOW®** Software Tool.

**SYNTAX** **THER** [**C<sub>ps</sub>**, **K<sub>Ts</sub>**, **α<sub>L</sub>**, **α<sub>T</sub>**] [**subrgn**]

**C<sub>ps</sub>** Specific heat of the dry solid,  $C_{ps}(\geq 0)$ , with typical units as [J/(kg °K)]. The default value is 1000 if **GAS** or **DENSITY GAS** command is given otherwise it is 1.

**K<sub>Ts</sub>** Thermal conductivity of the dry solid,  $K_{Ts}(\geq 0)$ , with typical units as [J/(m-s °K)]. The default value is 0.

**α<sub>L</sub>** Longitudinal dispersivity,  $\alpha_L(\geq 0)$  with typical units as. [m]. The default value is 0.

**α<sub>T</sub>** Transverse dispersivity,  $\alpha_T(\geq 0)$  with typical units as [m]. The default value is 0.

**subrgn** The subregion for which the input is specified. See Section 3.4 for more details. If no subregion is specified, then entire computational domain is selected.

**APPLICABILITY**

This command is available only for the **PORFLOW®** Software Tool.

**REMARKS**

More complex functional relations for specific heat can be specified by the **SPECIFIC HEAT** command, those for the thermal conductivity by the **CONDUCTIVITY** command and for the dispersivity by the **DISPERSIVITY** command. Those commands provide greater flexibility and their use is recommended over this command.

**EXAMPLES**

**THER**mal properties cs = 26, kt = 45

**THER**mal props cs = 26, ks = 45, alfa1 = 0, alfa2 = 0.

**THER**mal properties ce = 26, ke = 45, alpha1 = 10, alfa2 = 2

## 5.111 TIDE

**PURPOSE** To specify the height of the tide at an open boundary as a function of time and space. This command is effective only for the TIDAL® Software Tool.

**MODE 1:** Tide as a Cosine or Sine Function

**SYNTAX** TIDE [dir] {COSI|SINE} {N} {T<sub>1</sub>, a<sub>1</sub>, φ<sub>1</sub>} [T<sub>n</sub>, a<sub>j</sub>, φ<sub>n</sub>]

**dir** One of the character strings: EAST, WEST, NORTH, or SOUTH. It denotes the boundary of the domain of interest where the water elevation is specified as a function of time and space. If no direction is specified, then the tidal boundary is assumed to be at the west.

**COSI** The tide is a cosine function given by:

$$\eta = \sum_{n=1}^N \eta_0^n \cos( f^n t + \phi^n ) ,$$

**SINE** The tide is a sine function given by:

$$\eta = \sum_{n=1}^N \eta_0^n \sin( f^n t + \phi^n ) ,$$

**N** Number of sets of harmonic tidal components that follow. The default value is 0.

**T<sub>1</sub>** The time period of the 1<sup>st</sup> component of the tide. The default value is 0.

**a<sub>1</sub>** The amplitude of the 1<sup>st</sup> component of the tide. The default value is 0.

**φ<sub>1</sub>** The phase of the 1<sup>st</sup> component of the tide. The default value is 0.

**T<sub>n</sub>, a<sub>j</sub>, φ<sub>n</sub>** Triplets of period, amplitude, phase lag for the 2<sup>nd</sup> through N<sup>th</sup> component of the tide.

**APPLICABILITY**

This command is available only for the TIDAL® Software Tool.

**EXAMPLES**

TIDE is COSine function, 1 set: period=24 hrs, amplitude=30 cm, phase=12 hrs

TIDE SOUTH is the sum of SINE functions, 2 sets: (24, 30, 0) (676, 10, 0)

**MODE 2:** Tide specified as a table of Time and Height

**SYNTAX** TIDE [dir] {Nsets} {t<sub>1</sub>, η<sub>1</sub>} [t<sub>n</sub>, ..., η<sub>n</sub>]

**dir** One of the character strings: **EAST**, **WEST**, **NORTH**, or **SOUTH**. It denotes the boundary of the domain of interest where the water elevation is specified as a function of time and space. If no direction is specified, then the tidal boundary is assumed to be at the west.

**Nsets** Number of sets of tide values which follow. The default value is 0.

**t<sub>1</sub>** The time at which the first tide height is specified. The default value is 0.

**η<sub>1</sub>** The tide height at time **t<sub>1</sub>**. The default value is 0.

**t<sub>n</sub>, ..., η<sub>n</sub>** Pairs of time and tide height for one complete tidal cycle

#### APPLICABILITY

---

This command is available only for the **TIDAL®** Software Tool.

#### EXAMPLES

---

**TIDE WEST:** (time, height) table, 5 sets: (0, 10) (3, 15) (9, 20) (12, 25) (24, 10)

<b>MODE 3:</b>	<b>Tide as a Linear Function of Distance and Time</b>
<b>SYNTAX</b>	<b>TIDE</b> { <b>LINE</b> } [ <b>dir</b> ] { <b>Nsets</b> } { <b>t<sub>1</sub></b> , <b>η<sub>11</sub></b> , <b>η<sub>12</sub></b> } [] { <b>ξ<sub>start</sub></b> , <b>ξ<sub>end</sub></b> }
<b>LINE</b>	The tidal boundary condition at the specified boundary is taken to be a linear function of both time and distance along the boundary.
<b>dir</b>	One of the character strings: <b>EAST</b> , <b>WEST</b> , <b>NORTh</b> , or <b>SOUTH</b> . It denotes the boundary of the domain of interest where the water elevation is specified as a function of time and space. If no direction is specified, then the tidal boundary is assumed to be at the west.
<b>Nsets</b>	Number of sets of tide values which follow. There is no default value; a value must be specified.
<b>t<sub>1</sub></b>	The time at which the first tide height is specified.
<b>η<sub>11</sub></b>	The value of the tide at the first location ( <b>Nn+1</b> below) on the specified boundary.
<b>η<sub>12</sub></b>	The value of the tide at the second location ( <b>Nn+2</b> below) on the specified boundary.
<b>t<sub>n</sub>, η<sub>n1</sub>, η<sub>n2</sub></b>	Triplets of time, and tide at the first and second locations in the manner of <b>t<sub>1</sub></b> , <b>η<sub>11</sub></b> , <b>η<sub>12</sub></b> .
<b>ξ<sub>start</sub>, ξ<sub>end</sub></b>	Grid coordinates of the starting and ending locations on the boundary for linear interpolation of the tide.

**APPLICABILITY**

This command is available only for the **TIDAL®** Software Tool.

**EXAMPLES**

**TIDE** NORTh LINEar 3 sets: (0,1,0) (5.E4,5,10) (1.E5,0,1) at X: 0, 1000

**MODE 4:** Tide Height Determined by a Connected Reservoir

**SYNTAX** TIDE {RESE} [dir] {QR}

**RESE** The tidal boundary condition for tide height,  $\eta$ , as a function of time,  $t$ , at the specified boundary is given by:

$$\frac{\partial \eta}{\partial t} = + - \frac{Q}{Q_R} ,$$

Here  $Q$  is the actual computed flow rate at the boundary and  $Q_R$  is the specified capacity factor.

**dir** One of the character strings: **EAST**, **WEST**, **NORTH**, or **SOUTH**. It denotes the boundary of the domain of interest where the water elevation is specified as a function of time and space. If no direction is specified, then the tidal boundary is assumed to be at the west.

**QR** The reservoir capacity factor. The value must be greater than or equal to zero. The default value is 0.

#### EXAMPLES

---

**TIDE** EAST boundary: RESErvoir, capacity factor = 1.0E10

**5.112 TIME**

**PURPOSE** To set the initial simulation time for a problem.

**SYNTAX** **TIME** {**N1**}

**N1** The starting time ( $\geq 0$ ) for simulations. The default value is 0.

**EXAMPLES**

---

**TIME** = 50 years at start of simulations

**5.113 TITLE****PURPOSE** To specify the problem title.**SYNTAX** **TITL** followed by character information**REMARKS**

---

The specification must be restricted to one 80 character record. Title specification is included in all output files generated by the ACRi Software.

**EXAMPLES**

---

**TITL**e ILLUSTRATIVE PROBLEM - DEFAULT SET UP - 07/01/93: ACRi/akr

## 5.114 TORTUOSITY

**PURPOSE** To define tortuosity factor for the flow path in the porous material matrix. This command is effective only for the **PORFLOW®** Software Tool.

**SYNTAX** **TORT** [**V<sub>1</sub>**, **V<sub>2</sub>**, **V<sub>3</sub>**] [**subrgn**]

**V<sub>1</sub>**, **V<sub>2</sub>**, **V<sub>3</sub>** The x, y and z direction components, respectively, of the tortuosity factor  $\tau_{ij}$  as shown in the equation below that are used to compute the effective diffusivity. Only components in the principal directions are used. The default value is 1.

$$\Gamma_{ij}^T = (1 - \theta_T) k_S + \theta_T \tau_{ij} S k_f + \theta_E S \rho D_{ij}$$

$$\Gamma_{ij}^\Phi = \theta_D \tau_{ij} S \rho D_M^\Phi + \theta_E S \rho D_{ij}$$

$$D_{ij} = \alpha_T \delta_{ij} \underline{v} + (\alpha_L - \alpha_T) \frac{|v_i v_j|}{\underline{v}}$$

Where the superscript “n” denotes the nth fluid phase and::

$\Gamma_{ij}^T$ , $\Gamma_{ij}^\Phi$	are the diffusion coefficients for thermal and species equations,
$\theta_E, \theta_T, \theta_D$	are the effective, total and diffusional porosities of the material
$k_S, k_f$	are the solid and fluid thermal conductivities,
$D_M$	is the molecular diffusivity of the species
$D_{ij}$	is the dispersion tensor
$v_i, \underline{v}$	are the components of particle velocity and the modulus of velocity,
$S$	is the saturation of the fluid phase
$\alpha_L, \alpha_T$	are the longitudinal and transverse dispersion coefficients
$\rho$	is the density of fluid
$\tau_{ij}$	is the tortuosity or constrictivity tensor

**APPLICABILITY**

This command is currently available only for the **PORFLOW®** Software Tool.

**EXAMPLES**

**TORTUOSITY** factors: **0.7**, **0.6**, **0.9**

5.115 TRACK

**PURPOSE** To compute particle tracks and their corresponding elapsed time in the flow field.

**MODE 1:** Particle Locations and Tracking Options

**SYNTAX** TRAC [fname] [TABL] [STOP] [option] {N<sub>1</sub>, ..N<sub>n</sub>} [V<sub>stop</sub>, T<sub>0</sub>, N<sub>frq</sub>]

**fname** The name of the file to which the output is directed. See Section 3.3 for additional information. If no name is specified then the output is directed to a file with the same name as the Standard Output file but with the extension removed and replaced by “\_TRACK.TMP”. For example, if the Standard Output file is “PROBLEM.OUT”, then the file is named “PROBLEM\_TRACK.TMP”. At any time only one track file can be open. If a new file name is given, then the previous file is closed and output is directed to the new file. A summary of output is also printed to the standard output device at the end of simulations

**TABL** The particle track data are printed in a tabular form at the end of simulations.

**STOP** The particle stopping criterion is explicitly specified as one of the options listed in the table below. If no stopping criterion is specified then the particle is tracked to the end of simulations or till it reaches or crosses a boundary element.

**option** The stopping option for the particle; the modifier **STOP** must be present for one of these options to be effective.

option	INTERPRETATION
<b>X</b>	Particle tracking stops if its x-coordinate exceeds the N <sub>n+1</sub> <sup>th</sup> numerical value.
<b>Y</b> or <b>R</b>	Particle tracking stops if its y-coordinate exceeds the N <sub>n+1</sub> <sup>th</sup> numerical value.
<b>Z</b> or <b>THETA</b>	Particle tracking stops if its z-coordinate exceeds the N <sub>n+1</sub> <sup>th</sup> numerical value.
<b>DIST</b>	Particle tracking stops when its distance from the point of release exceeds the N <sub>n+1</sub> <sup>th</sup> numerical value. This is default option if <b>STOP</b> modifier is specified.
<b>TIME</b>	Particle tracking stops when the time exceeds the N <sub>n+1</sub> <sup>th</sup> numerical value.
<b>ELAP</b>	Particle tracking stops when the elapsed time from its moment of release exceeds the N <sub>n+1</sub> <sup>th</sup> numerical value.
<b>FREE</b>	Particle tracking stops if it reaches a free surface or zone of saturation.

**N<sub>1</sub>, .., N<sub>n</sub>** The coordinates of the starting location of the particle. Two values are required for the 2D and 3 for 3D input modes.

**V<sub>stop</sub>** The numerical value pertaining to the stopping criterion as described under options listed in the table above. There is no default value. A dummy numerical value must be specified if any numerical input follows for the time or frequency input below.

**T<sub>0</sub>** In the transient solution mode, the starting time for the particle; the default value is 0. In the steady state solution mode, the time interval for the particle computations; the default value is 1.

**N<sub>frq</sub>** The frequency index for providing the output in the particle track file. The output is obtained every **N<sub>n+3</sub>** (≥1) steps. For example, a specification of 10 result in output at the 10th, 20th, 30th, etc. step. If no input is specified, the frequency is assumed to be 1.

**EXAMPLES**

---

**TRACk**s for particle start at: (35.0, 5.27)

**TRACk**s for particle start at: (1.22, 10.0, 19.3) print **TABLe**s also

**TRACk** particle at: (1.22, 10.0) print **TABLe**s and save on file 'TRACK.SAV'

**TRACk** particle at: (1.22, 10.0, 19.3) **STOP** at X=100. Starting time = 20 years

**TRACk** at: (1.2, 10.0) stop\_distance 150 m; delta\_t 0.1 yrs (steady state mode)

**TRACk**s at: (35.0, 5.27) **STOP TIME** 200. Start at 0 yrs; output every 20 steps

**TRACk**s for particle start at: (1.22, 10.0, 19.3) **STOP** at **FREE** surface.

**TRACk**s at: (35.0, 5.27) **STOP ELAP**sed **TIME** 100. on file "PARTICLE.TRK"

**MODE 2:** Particle Tracking Factors

**SYNTAX** TRAC {FACT} [subrgn] [MULT | DIVI] {V<sub>factor</sub>}

**FACT** The particle velocity in the specified region is multiplied or divided by a specified factor.

**subrgn** The subregion for which the input is specified. See Section 3.4. If no subregion is specified, the entire computational domain is selected

**MULT** Particle velocity in the selected region is multiplied by V<sub>factor</sub>. This is the default option.

**DIVI** Particle velocity in the selected region is divided by V<sub>factor</sub>.

**V<sub>factor</sub>}** The numerical value of the factor to modify the particle velocity. There is no default value.

#### EXAMPLES

---

TRACking FACTor for the particle is 0.5!! to account for retardation

TRACking FACTor: is 0.5 in the currently SELEcted subregion

TRACking FACTor is 2.5 in the subregion with ID=FAST

TRACking FACTor: DIVIde particle velocity by 2.0!! To account for retardation

## 5.116 TRANSFER

**PURPOSE** To specify the transfer of mass flux and fluid properties from one subregion to another. This command is effective only for the ANSWER® Software Tool.

**MODE 1:** Direct Transfer of Computed Flux

**SYNTAX** TRANSFER [FLUX] [LOCA] [subrgn]

**FLUX** The flux of mass is transferred from one surface to another. This type of transfer can only be defined by a LOCATE command with MATCHed pair option. The purpose of this command is to simulate embedded objects in the flow field such as pipes, pumps, fans, propellers and swirlers which are not explicitly resolved by the computational grid. The gross effect of such embedded objects is to transfer fluid flux and properties (with or without modification) from one location to another. For effective use of this option the object should be declared as a blocked region (BLOCK command) or the values of all the variables within the object should be fixed with the FIX command.

The flux computed at the 1<sup>st</sup> surface (“donor”) of a matched pair of surfaces is transferred to the 2<sup>nd</sup> surface (“receptor”). Both surfaces must be internal to computational domain; external surfaces are not allowed. There must be a one-to-one correspondence in the number of segments between the two surfaces.

The mass flux is computed at each segment of the donor surface from velocity components at the element located just upstream of the surface. This is then assumed to be the mass flux at the corresponding receptor surface. Any mass flux computed internally at the receptor surface is ignored. The momentum loss at the donor surface is equal to mass flux multiplied by the velocity components. At the receptor surface, by default, the flow enters normal to the surface. However, this can be changed by the LOCAL modifier below.

The loss (sink) of a conserved variable at the donor surfaces is computed as mass flux multiplied by the value of the variable at the element just upstream of the surface. An equal amount (source) is then injected at the element just downstream of the receptor surface.

**LOCA** By default the flow at the receptor surface is assumed to enter with a velocity vector that is normal to the surface. If the LOCAL modifier is present, then it is assumed that the velocity components of the flow entering the receptor are equal to the locally prevailing values. Thus there is no net effect of the entering flow on the local momentum.

**subrgn** The subregion for flux transfer. The subregion must have been previously defined by a LOCATE MATCH command. A subregion must be defined; there is no default value.

**EXAMPLES**

TRANSfer FLUXes from 1<sup>st</sup> surface (entrance) of subregion ID=Q\_TRANSFER to the 2<sup>nd</sup> surface (exit)  
 TRANSfer FLUXes from 1<sup>st</sup> surface of SELEcted subregion to the 2<sup>nd</sup> with LOCAL momentum

<b>MODE 2:</b>	<b>Transfer of Computed Amount as Flux or Source</b>
<b>SYNTAX</b>	<b>TRANSFER</b> { <b>FLUX</b>   <b>SOUR</b> } [ <b>func</b> ( $\xi$ )] { <b>N</b> <sub>1</sub> , <b>N</b> <sub>2</sub> , ..., <b>N</b> <sub>n</sub>   <b>fname</b> } [ <b>LOCA</b> ] [ <b>subrgn</b> ] [ <b>TOTA</b> ] [ <b>VOLU</b>   <b>AREA</b> ] [ <b>INTE</b> ]
<b>FLUX</b>	Same as Mode 1 except that the user specifies the mass flux rate at the matched surfaces. Fluid is extracted from elements just upstream of the 1 <sup>st</sup> of a matched pair of surfaces and transferred to the corresponding elements just downstream of the 2 <sup>nd</sup> surface. This is the default mode.
<b>SOUR</b>	Similar to <b>FLUX</b> modifier except that the mass is transferred as a source rather than as flux.
<b>func</b>	One of the modifiers listed in Table 4.2.1 that denotes the functional form of the source. If no function is specified, the value is assumed to be constant.
$\xi$	One of the independent variables listed in Table 4.2.2 if no independent variable is specified, the variable is assumed to be time.
<b>N</b> <sub>1</sub> ,..., <b>N</b> <sub>n</sub>	The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.
<b>fname</b>	The name of the file containing the numerical values <b>N</b> <sub>2</sub> through <b>N</b> <sub>n</sub> . This option is available only for selected functions. See Section 3.3 for additional information.
<b>LOCA</b>	See Mode 1 Specification.
<b>subrgn</b>	The sub region for flux or source transfer. If the <b>SOURCE</b> modifier without <b>AREA</b> is present, then the sub region must have been previously defined by a <b>LOCATE MATCH</b> or a <b>LOCATE CORRelated</b> command. Otherwise the sub region must have been previously defined by a <b>LOCATE MATCH</b> command. A subregion must be defined; there is no default value.
<b>TOTA</b>	By default, the amount of source specified, or computed from <b>func</b> ( $\xi$ ), is applied to each element of the <b>subrgn</b> . If this modifier is present then the amount is assumed to be the total quantity over whole of the <b>subrgn</b> . In this case, the source is distributed equally to all the elements of the <b>subrgn</b> unless the <b>VOLUME</b> or <b>AREA</b> modifiers are present.
<b>VOLU</b>	If the <b>TOTAL</b> modifier is absent, the source for each element is computed as: $Q=q\delta V$ . Here <b>q</b> is computed from <b>func</b> ( $\xi$ ) and <b>N</b> <sub>1</sub> ,..., <b>N</b> <sub>n</sub> and, $\delta V$ is the volume of the element. If the <b>TOTAL</b> modifier is present, the amount for each element is equal to: $Q=q\delta V/V$ , where <b>V</b> is the volume of the total <b>subrgn</b> .
<b>AREA</b>	If <b>TOTAL</b> modifier is absent, the source for each element is computed as: $Q=q\delta A$ , if <b>TOTAL</b> modifier is present, the source is computed as: $Q=q\delta A/A$ . Here $\delta A$ is the area of the element boundary and <b>A</b> is the total area of at the 1 <sup>st</sup> (inlet) surface.
<b>INTE</b>	By default wall function treatment at the walls of the object is deactivated and the diffusive flux at the wall is set to zero. If the <b>INTERNAL</b> modifier is present then the wall treatment is retained and the wall diffusive flux is included in the computations.

**EXAMPLES**

---

Generic examples for this command are given in Section 4.4. The keyword **TRANSFER** must replace the keyword and the dependent variable ( $\Phi$ ) used in these examples. Some additional examples that illustrate the use of the attributes specific to this command are given below.

**TRANSFER** FLUX = 0.0256 1<sup>st</sup> surface (entrance) of subregion ID=Q\_TRANSFER to the 2<sup>nd</sup> surface (exit)

**TRANSFER** SOURCE TOTAL amount of 0.0256 for SELEcted subregion

**TRANSFER** SOURCE = 0.0256 per unit VOLUME for SELEcted region; treat as INTERNAL source

**TRANSFER** FLUX = 0.0256 per unit AREA for SELEcted region

**TRANSFER** SOURCE TOTAL amount of 0.0256 and per unit VOLUME for SELEcted region

**TRANSFER** SOURCE is TABLE per unit AREA: 3 sets (TIME, value) (0, 0.01), (100, 0.10), (200, -0.20)

**TRANSFER** FLUX is TABLE per unit AREA: 3 sets (TIME, value) from file 'TRANSFER.FIL'

**MODE 3:** Transfer Based Upon Drag Coefficient and Pressure Difference

**SYNTAX** TRANSFER {DRAG} [FLUX|SOUR] [LOCA] [subrgn] [VOLU|AREA] [REVE] [INTE] {C<sub>D</sub>} [D, β]

**DRAG** The amount of mass transferred from one location to another is computed from a pressure gradient based drag law. It is assumed that flow occurs as a result of pressure difference across the object defined by the two surfaces. The fluid is extracted from elements just upstream of the 1<sup>st</sup> of a matched pair of surfaces and transferred to the corresponding elements just downstream of the 2<sup>nd</sup> surface. In all other respects command mode is identical to Mode 1. The flow rate is computed as:

$$q = \rho \delta A \left[ \frac{1}{2 C_D} \frac{P_{in} - P_{out}}{\rho} \left( \frac{\rho D}{\mu} \right)^\beta \right]^{\frac{1}{2-\beta}},$$

In the equation above:

- ρ** is the fluid density
- δA** is area of a segment (each element boundary) of the 1<sup>st</sup> surface through which the flow is extracted,
- C<sub>D</sub>** is the drag coefficient,
- P<sub>in</sub>** is the average pressure across the inlet surface (determined by taking an area-based average of the pressure at each segment of the surface) of the sub region,
- P<sub>out</sub>** is the average pressure across the outlet surface of the sub region,
- D** is the hydraulic diameter that defines an appropriate Reynolds number,
- μ** is the dynamic viscosity, and
- β** is an exponent that represents the dependence of the drag on Reynolds number (For laminar flow β is unity; for turbulent flow β is a weak function of Reynolds number. At high Reynolds number, for smooth pipes, the empirical value of C<sub>D</sub> is 0.0791 and that of β is 0.25. Often for fully turbulent flow, the effect of Reynolds number on β is ignored and its value taken as zero).

**FLUX** See Mode 2 Specification.

**SOUR** See Mode 2 Specification.

**LOCA** See Mode 1 Specification.

**subrgn** The sub region for mass transfer which must have been previously defined by a LOCATE MATCH command. There is no default value.

**VOLU** See Mode 2 Specification.

**AREA** See Mode 2 Specification.

**INTE** See Mode 2 Specification.

**REVE** By default flow can only occur from 1<sup>st</sup> surface to the 2<sup>nd</sup> surface irrespective of the pressure gradient. If P<sub>2</sub> exceeds P<sub>1</sub> the source is set to zero. If this modifier is present then reverse flow is allowed if the pressure gradient is from the 2<sup>nd</sup> to the 1<sup>st</sup> surface.

**C<sub>D</sub>** The drag coefficient, C<sub>D</sub>. There is no default value for this input.

**D** The hydraulic diameter, D, that defines the Reynolds number. This value is required only if the exponent β is specified. There is no default value for this input.

**β** The exponent β for the drag coefficient. The default value is zero.

**REMARKS**

---

This mode of the **TRANSFER** command is automatically selected if the **DRAG** modifier is present and there are no more than 3 numerical values on the command.

**EXAMPLES**

---

**TRANS**fer DRAG C\_drag = 0.10 for subregion ID=Q\_TRANSFER

**TRANS**fer as FLUX DRAG C-drag = 0.5 Length scale = 0.2 beta=0.25 ID=Q\_TRANSFER

**TRANS**fer DRAG as SOURCE: CD = 0.10 Hydraulic Dia = 0.2 beta =0.25 by VOLUME ID=Q\_TRANSFER

**TRANS**fer DRAG as FLUX: CD = 0.10 Hydraulic Dia = 0.2 beta =0.25 by AREA ID=Q\_TRANSFER

**TRANS**fer DRAG SOURCE: CD=0.10 Hyd\_Dia=0.2 beta =0.25 allow REVERSE flow ID=Q\_TRANSFER

**MODE 4:** Transfer due to Pressure Difference across Sub-Grid Scale Holes

**SYNTAX** TRANSFER {DRAG} [FLUX | SOUR] [LOCA] [subrgn] [VOLU | AREA] [REVE] [INTE] {  
C<sub>D</sub>, D, N, L} [β]

**DRAG** The amount of mass transferred from one location to another is computed from a pressure gradient based drag law. This simulates flow through small holes that are not explicitly resolved by the grid. It is assumed that flow occurs as a result of pressure difference across the holes. The fluid is extracted from elements just upstream of the 1<sup>st</sup> of a matched pair of surfaces and transferred to the corresponding elements just downstream of the 2<sup>nd</sup> surface. The flow rate is computed as:

$$q = \rho N \frac{\pi D^2}{4} \left[ \frac{1}{2 C_D} \frac{P_{in} - P_{out}}{\rho} \left( \frac{\rho D}{\mu} \right)^\beta \frac{D}{L} \right]^{\frac{1}{2 - \beta}},$$

In the equation above:

- ρ** is the fluid density
- N** is the number of holes,
- D** is the diameter of holes,
- C<sub>D</sub>** is the drag coefficient,
- P<sub>in</sub>** is the average pressure across the inlet surface (determined by taking an area-based average of the pressure at each segment of the surface) of the sub region,
- P<sub>out</sub>** is the average pressure across the outlet surface of the sub region,
- D** is the hydraulic diameter that defines an appropriate Reynolds number,
- μ** is the dynamic viscosity,
- L** is the length of each hole, and
- β** is an exponent that represents the dependence of the drag on Reynolds number (For laminar flow β is unity; for turbulent flow β is a weak function of Reynolds number. At high Reynolds number, for smooth pipes, the empirical value of C<sub>D</sub> is 0.0791 and that of β is 0.25. Often for fully turbulent flow, the effect of Reynolds number on β is ignored and its value taken as zero).

**FLUX** See Mode 2 Specification.

**SOUR** See Mode 2 Specification.

**LOCA** See Mode 1 Specification.

**subrgn** See Mode 3 Specification.

**VOLU** See Mode 2 Specification.

**AREA** See Mode 2 Specification.

**INTE** See Mode 2 Specification.

**REVE** See Mode 3 Specification.

**C<sub>D</sub>** The drag coefficient, **C<sub>D</sub>**, for the holes. There is no default value for this input.

**D** The diameter, **D**, of the holes through which mass is transferred in response to a pressure gradient. There is no default value for this input.

**N** Total number of holes, **N**. There is no default value for this input.

**L** Average length of the holes, **L**. There is no default value for this input.

**β** The exponent **β** for the drag coefficient. The default value is zero.

**REMARKS**

---

This mode of the **TRANSFER** command is automatically selected if the **DRAG** modifier is present and there are at least 4 numerical values on the command.

**EXAMPLES**

---

**TRANS**fer DRAG C\_drag= 0.10 dia= 0.001, number=200, Length= 0.005, for subregion ID=Q\_TRANSFER

**TRANS**fer DRAG C\_drag= 0.10 dia= 0.001, number=200, L= 0.005, beta=0.25 ID=Q\_TRANSFER

**TRANS**fer DRAG C\_drag= 0.10 dia= 0.001, n=200, L= 0.005, distribution by VOLUME ID=Q\_TRANSFER

**TRANS**fer DRAG C\_drag= 0.10 dia= 0.001, n=200, L= 0.005, distribution by AREA ID=Q\_TRANSFER

**TRANS**fer DRAG C\_drag= 0.10 dia= 0.001, n=200, L= 0.005, Allow REVERSE flow; ID=Q\_TRANSFER

## 5.117 TRANSPORT

**PURPOSE** To specify the transport properties of the host porous matrix, or of planar or linear features. This command is effective only for the **PORFLOW®** Software Tool.

**MODE 1:** **Distribution Coefficient, Diffusivity and Dispersivity for Species**

**SYNTAX** **TRAN** { $\Phi$ } { $\alpha$ } [ $D_M$ ,  $\alpha_L$ ,  $\alpha_T$ ] [**SECO**|**SOLI**] [**WET**] [**subrgn**]

**$\Phi$**  **One or more** symbols that denote the dependent variables for which the transport properties are specified for the selected subregion. **Up to 100 symbols may be specified with one command.** The valid symbols are those pertaining to the species that are listed in Table 2.7.1. **If no symbol is specified, the values are assumed to be for all species variables.**

**$\alpha$**  The partition coefficient,  $k_d$  ( $\geq 0$ ), that relates the species concentration in the solid to that in the fluid phase. However, if a **PROPERTY** command with the same symbol and **EFFECTIVE** modifier was previously specified, the input value is interpreted as the retardation factor,  $R_D$  ( $> 0$ ). The equations below explain the interpretation of this input. **The default value for  $k_d$  is 0 and that for  $R_D$  is 1.** Please see the **STORAGE** command (**REMARKS** Section) for a full discussion of the interpretation of  $k_d$  and  $R_D$

**$D_M$**  Molecular diffusivity,  $D_M$  ( $\geq 0$ ) for the species. **The default value is 0.**

**$\alpha_L$**  The coefficient of longitudinal dispersivity:  $\alpha_L$  ( $\geq 0$ ). **The default value is 0.**

**$\alpha_T$**  The coefficient of transverse dispersivity:  $\alpha_T$  ( $\geq 0$ ). **The default value is 0.**

**SECO** **By default the species transport occurs in the 1<sup>st</sup> phase of the fluid.** If **SECO** is present, then the species transport is in the 2<sup>nd</sup> fluid phase. This means that the convective fluxes, saturation and fluid properties for the 2<sup>nd</sup> phase are used to solve the governing equations. The pressure equation for the 2<sup>nd</sup> phase must be solved for this modifier to be effective.

**SOLI** **By default the transport of mass species occurs in a fluid phase.** If this modifier is present then the transport is in the solid phase of the porous matrix and the governing transport equation is:

$$\frac{\partial}{\partial t} \left[ (1 - \theta_E) \rho_S C^k \right] = \frac{\partial}{\partial x_i} \left( \rho_S \Gamma_{ij}^c \frac{\partial C^k}{\partial x_j} \right) + S_c^k - (1 - \theta_E) \rho_S \gamma^k$$

**WET** The modifier is effective only if the **SOLID** modifier is also present on the command. By default, the accumulation term for the solid phase transport assumes that the total amount of solid is available for transport of the species. If this modifier is present, then only the wetted part of the solid participates in transport. That is the accumulation term of the above equation and is further multiplied by the saturation,  $S$ , of the first fluid phase.

**subrgn** The subregion for which the input is specified. See Section 3.4 for more details. **If no subregion is specified, then entire computational domain is selected.**

**APPLICABILITY**

This command is available only for the **PORFLOW®** Software Tool.

**REMARKS**

More complex functional relations for  **$\alpha$**  can be specified by the **DISTRIBUTION** command, those for the diffusivity by the **CONDUCTIVITY** command and for the dispersivity by the **DISPERSIVITY** command.

**REMARKS**

This command is a compact input mode to specify constant values for the transport properties for mass species. If these properties vary as functions of other variables, then the individual components of the input can be specified (or overwritten) by the **STORAGE**, **CONDUCTION** and **DISPERSION** commands.

**EXAMPLES**

---

**TRAN**sport properties: for all kd=1, Dm=5.24, alpha L=10, alpha T=1  
**TRAN**sport properties for C2: kd=1, Dm=5.24, alpha L=10, alpha T=1  
**TRAN**sport foer C2 Rd=21, dm=0, alpha\_L=10, alpha\_T=1  
**TRAN**sport for C2 in SOLId phase kd=0, dm=1.E-5  
**TRAN**sport for C2 only in WET part of SOLId phase.

**MODE 2:** Exponential Diffusivity Relation for a Species

**SYNTAX** TRAN [ $\Phi$ ] {EXPO} { $\alpha$ } [ $D_M$ ,  $\beta$ ,  $\gamma$ ] [SECO|SOLI] [WET] [subgrn]

$\Phi$  A symbol to denote the dependent variable for which the transport properties are specified. The valid symbols are those pertaining to transport equations of Table 2.7.1. By default, the specification is assumed to be for the 1<sup>st</sup> transport equation or chemical species.

**EXPO** The diffusion coefficient for the species,  $\Gamma$ , is computed from the exponential relation:

$$\Gamma = D_M \beta \exp(\gamma S \Theta_D),$$

where  $S$  is the phase saturation and  $\Theta_D$  is the diffusional porosity. Other symbols are described below.

$\alpha$  See mode 1 of command.

$D_M$  Molecular diffusivity  $D_M (\geq 0)$  for the chemical species. The default value is 0.

$\beta$  The coefficient  $\beta (\geq 0)$  of the diffusional relation given above. The default value is 0.

$\gamma$  The coefficient  $\gamma (\geq 0)$  of the diffusional relation given above. The default value is 0.

**SECO** See mode 1 of command.

**SOLI** See mode 1 of command.

**WET** See mode 1 of command.

**subgrn** See mode 1 of command.

#### APPLICABILITY

---

This command is available only for the **PORFLOW®** Software Tool.

#### EXAMPLES

---

TRANsport EXPOnential form: kd=1.1E-3, dm=1.2, a1=10, a2 = 5

## 5.118 TURBULENCE

**PURPOSE** To select the appropriate turbulence model and specify the empirical constants. **This command is available only for the ANSWER® Software Tool.**

**MODE 1:** Selection of Two Equation Turbulence Model

**SYNTAX** **TURB** {**model**} [**LOW**] [**VELO**] ] [**AFTE** | **BEFO**] [**COMP**] [**PASS**] [**DELA**=**N<sub>delay</sub>**]

**model** The turbulence model to be used. By default the standard k-ε model is used.

model	INTERPRETATION
<b>CHIE</b>	Chien (1982) low Reynolds number model is used.
<b>CUBI</b>	Cubic <b>k-ε</b> model of Shih et al. (1997) is employed.
<b>LRR</b>	Launder, Reece and Rodi (1975) Reynolds stress model is used.
<b>LS</b>	Launder-Sharma (1974) low Reynolds number model is used.
<b>WILC</b>	Wilcox (2006) <b>k-ω</b> model is used.
<b>KOME</b>	Wilcox (2006) <b>k-ω</b> model is used but the $\frac{\partial k}{\partial x_j} \frac{\partial \omega}{\partial x_j}$ term is omitted
<b>MENT</b>	Menter (2003) <b>k-ω</b> model is used.
<b>QUAD</b>	Quadratic <b>k-ε</b> model of Shih et al. (1997) is employed.
<b>RNG</b>	RNG enhancements to the <b>k-ε</b> model are employed.
<b>RSM</b>	Reynolds stress model of Craft et al. (1996) & Iacovides et al. (1996) is used
<b>SSA</b>	So and Sarkar (1997) low Reynolds number model is used.
<b>SST</b>	Menter (1996) low Reynolds number shear-stress transport model is used.
<b>YSM</b>	Yang and Shih (1993) low Reynolds number model is used.

**LOW** This modifier is effective in the absence of any explicit selection of a low Reynolds number model. For the standard **k-ε** model, the Launder and Sharma (1974) low Reynolds number option is used. For the **QUADratic** and **CUBIC** models, the Yang and Shih (1993) low Reynolds option is used.

**VELO** By default, the near-wall omega for the **k-ω** models is computed from the shear stress computed from the near-wall turbulence kinetic energy. If this modifier is present then the shear stress for the near wall omega is computed from the near-wall velocity component.

**AFTE** By default, the turbulence quantities are computed before the flow computations except for those simulations that have dissolving salt blocks (see **BLOCK SALT** command) or **SOLVE** command with **CFL** or **AUTO** modifiers. If this modifier is present then the turbulence quantities are computed after the flow equations.

**BEFO** By default, the turbulence quantities are computed after the flow computations if **BLOCK SALT** command or **SOLVE** command with **CFL** or **AUTO** modifiers is present. If this modifier is present then the turbulence quantities are computed BEFORE the flow equations.

**COMP** By default, for incompressible flow, the contribution of the divergence term to the kinetic energy production is ignored. This term is typically negligible for subsonic flows. If this modifier is present, then this contribution is included.

**PASS** The **QUADratic**, **CUBIC**, **RSM** and **LRR** model add source terms to the momentum equations. If this modifier is present, then the contribution to the momentum equation is ignored.

**DELA** By default, for momentum source term contribution starts at the very beginning. If this flow, the contribution of the divergence term to the kinetic energy production. If this modifier is present, then start of the source terms is delayed.

**N<sub>delay</sub>** The number of steps after which the source term are included. If this modifier is present, then source term will be included after **N<sub>delay</sub>** steps.

**REMARKS**

---

Turbulence modeling still remains more of an art than science. The standard **k-ε** model is widely used and has a measure of acceptability. However, it is known to have many shortcomings. In particular, its use for flows with high swirl, high streamline curvature or strong pressure gradients may not prove satisfactory. Many of the other models should be considered experimental at this stage. Some of these perform well for certain class of flows. Caution should be exercised in evaluating the results obtained from all turbulence models. **ACRi makes some of the models available only by special arrangement.**

**EXAMPLES**

---

**TURBULENCE** model standard k-e

**TURBULENCE** model k-e with **LOW** Reynolds number extensions

**TURBULENCE** model k-e with **CHIEN** low Reynolds number extensions

**TURBULENCE** model **LRR** version of rsm

**TURBULENCE** model **RSM** of lauder & Craft

<b>MODE 2:</b>	<b>LES Turbulence Models: Smagorinsky and Deductive</b>
<b>SYNTAX</b>	<b>TURB {LES} [SMAG   DEDU DYNA  DBS] [Cs] [A1] [DELA=N<sub>Delay</sub>] [DIAG=N<sub>Diag</sub>]</b>
<b>SMAG</b>	Smagorinsky (1963) Large Eddy Simulation Model is used. This is the default option.
<b>DEDU</b>	The 2 <sup>nd</sup> order Deductive LES of Lee and Meecham (1996) is used.
<b>DYNA</b>	The deductive dynamic LES model of Lee, Kim & Runchal (2003) is used. In the default mode of the dynamic model, the constant (Cs) that is used to compute the turbulent eddy viscosity for this model is a function of space and time that is automatically computed. The specified value of the <b>C<sub>s</sub></b> is ignored. However, if the modifier <b>SMAGorinsky</b> is simultaneously present, then specified value of the <b>C<sub>s</sub></b> constant is added to the value computed by the model.
<b>DBS</b>	The LES model of D. B. Schein (2003) is used.
<b>C<sub>s</sub></b>	The Smagorinsky constant C <sub>s</sub> (>0). The default value is 0.03.
<b>A1</b>	The constant A1 of Equation 34 of D. B. Schein (2003). This constant determines the fraction of the second order terms of the turbulence stress tensor that are included as the velocity source terms. If present this constant must be the 2 <sup>nd</sup> numerical value on the command. The maximum value is 1. The default value is 0.0
<b>DELA</b>	This modifier is currently effective only in the presence of the <b>DEDU</b> and <b>DBS</b> modifiers. If this modifier is present, then onset of the velocity source terms (see A1 above) is delayed.
<b>N<sub>Delay</sub></b>	The number of steps starting at which the velocity source terms are included.
<b>DIAG</b>	This modifier is currently effective only in the presence of the <b>DBS</b> modifier. If this modifier is present, then detailed diagnostic output of all gradients and stress tensor that form the velocity source terms is printed out.
<b>N<sub>Diag</sub></b>	The number of steps at which the output of gradients and stress tensor is produced.

## REMARKS

---

Turbulence modeling still remains more of an art than science. The LES models are widely used and have a measure of acceptability. However, these models are known to have many shortcomings. In particular, the use of such models for flows where wall effects may dominate the flow may not prove satisfactory. These models should be considered experimental at this stage though they may perform well for certain class of flows. Caution should be exercised in evaluating the results obtained from all turbulence models. **ACRi makes some of the models available only by special arrangement.**

## EXAMPLES

---

**TURBULENCE** model is **LES**

**TURBULENCE** model is **LES DYNAMIC** with **SMAGORINSKY** constant Cs=**0.04**

**TURBULENCE** model is **LES DBS** with **SMAGORINSKY** constant Cs=**0.05**, A1=**0.1**, DELA=**50**, DIAG=**50**

**MODE 3:** Empirical Constants for Turbulent Flows

**SYNTAX** TURB [ $\kappa$ , E,  $C_\mu$ ,  $C_{\epsilon 1}$ ,  $C_{\epsilon 2}$ ]

- $\kappa$  The constant  $\kappa$  ( $>0$ ) of the log law of wall. The default value is 0.4187.  
E, The constant E ( $>0$ ) of the log law of wall. The default value is 9.793.  
 $C_\mu$  The constant  $C_\mu$  ( $>0$ ) of the **k- $\epsilon$**  model. The default value is 0.09.  
 $C_{\epsilon 1}$  The constant  $C_{\epsilon 1}$  ( $>0$ ) of the **k- $\epsilon$**  model. The default value is 1.44.  
 $C_{\epsilon 2}$  The constant  $C_{\epsilon 2}$  ( $>0$ ) of the **k- $\epsilon$**  model. The default value is 1.92.

#### REMARKS

---

The default values are for the 'standard' k- $\epsilon$  model of turbulence. Some of the other models have their own built-in default values. This mode of the command may be invoked in a single mixed command with the Mode 1 of the command.

#### EXAMPLES

---

TURBulence kappa=0.4, E=9.

TURBulence kappa=0.4, E=9., CMU=0.09, Ce1 = 1.4, Ce2 = 2.0

TURBulence model RSM kappa=0.4, E=9., CMU=0.09 !! Differential Reynolds stresses Model

**5.119 UPTAKE**

**PURPOSE** To specify uptake and precipitation reactions of dissolved species. This command is effective only for the **PORFLOW®** Software Tool.

**REMARKS** \_\_\_\_\_

This command is identical to the **PRECIPITATION** command.

**EXAMPLES** \_\_\_\_\_

See **PRECIPITATION** command

**5.120 USER**

**PURPOSE** To specify the user identification for the problem.

**SYNTAX** **USER** followed by character information

**REMARKS**

---

The specification must be restricted to one 80 character record. User identification is included in all output files generated by the ACRi Software Tool.

**EXAMPLES**

---

**USER** ACRi - AKR demonstration

## 5.121 VELOCITY

**PURPOSE** To specify the direct computation of Darcy velocity from pressure field. This command is effective only for the **PORFLOW®** Software Tool.

**SYNTAX** **VELO** {**PRES**}

**PRES** By default the representative Darcy velocity at a grid node is computed by dividing the arithmetic average of the flux by the average directed area of the faces. For example, for the hexahedral element shown in Figure 3.2.1, the U, V and W velocity components are computed as the averages of flux at each of the faces divided by the total face area and multiplied by the directed area in the x, y and z direction, respectively. If **PRESSURE** modifier is present, then the velocity components are directly computed from the hydraulic conductive and the local pressure gradient.

**REMARKS**

---

The **PORFLOW®** Software Tool directly computes the fluxes from the continuity equation at cell boundaries (see Figure 3.2.1). The fluxes are computed with the hydraulic conductivity computed at the cell faces. The cell-face value of hydraulic conductivity is different from the value at the node depending on the method of averaging (see the **PROPERTY** command). This command computes the Darcy velocity from the hydraulic conductivity and the local pressure gradient at the nodal point located within the cell. Thus strictly speaking, the Darcy velocity computed by this command represents the local cell velocity. On the other hand, the fluxes, and the implied velocity, at the cell face are more representative of the numerical solution which satisfies the governing equation.

**EXAMPLES**

---

**VELO**city directly from **PRESSURE** gradient

## 5.122 VISCOSITY

**PURPOSE** To specify the viscosity and to select associated options.

**MODE 1:** **Generic Functional Form for Viscosity**

**SYNTAX** **VISC** {func{ $\xi$ }} { $N_1, N_2, \dots, N_n$ |fname} [EFFE] [operation] [ALWA] [phase] [subrgn]

**func** One of the modifiers listed in Table 4.2.1 that denotes the functional form of the input. For this input, the function specifies the value of the viscosity for the corresponding phase. If no function is specified then the value is assumed to be constant.

**$\xi$**  One of the independent variables listed in Table 4.2.2. If no independent variable is specified, the variable is assumed to be time.

**$N_1, \dots, N_n$**  The numerical constants and coefficients for the selected function. See Section 4.4 for more details. There are no default values for this input.

**fname** The name of the file containing the numerical values  $N_2$  through  $N_n$ . This option is available only for selected functions. See Section 3.3 for additional information.

**EFFE** By default for turbulent flow in ANSWER® Software Tool, the specification is assumed to be for the molecular dynamic viscosity. The turbulent eddy viscosity is added to the molecular viscosity. If this modifier is present, then the specified value becomes the total viscosity for the fluid. For PORFLOW® Software Tool this modifier is assumed to be present by default.

**operation** Available modifiers are REPLACE, ADD, SUBTRACT, MULTIPLY, DIVIDE, ABS, ABSOLUTE, POSITIVE and NEGATIVE. See Section 4.3 for further details.

**ALWA** By default, for PORFLOW® Software Tool the VISCOSITY command is implemented **immediately and only once** – as soon as the command is encountered. If this modifier is present then the command is executed immediately as well as **repeatedly** at the beginning of each time step (or iterative step in steady state mode) of the solution procedure. For ANSWER® Software Tool this modifier is assumed to be present by default.

**phase** The phase for which the input is specified. See Section 3.6 for available options. By default the input pertains to the 1<sup>st</sup> phase of the fluid. This modifier is currently available only for the ANSWER® and PORFLOW® Software Tools.

**subrgn** The subregion for which the input is specified. See Section 3.4. If no subregion is specified, the entire computational domain is selected.

**REMARKS**

The effective viscosity for the fluid (combined molecular and turbulent components) may also be set by the SET Command with MU modifier. In this case the modifier ALWAYS must be specified if the viscosity is to be set as the specified function throughout the solution process. Otherwise the values will be set only at the time of specification and may be subsequently overwritten by other computations. The VISCOSITY command is recommended as the preferred mode of input.

The effective viscosity may also be set by the CONDUCTIVITY command with U, V or W modifiers. This option allows the specification of anisotropic viscosity; that is the coefficient of viscosity can be specified as a vector so that the components in the 3 principal directions are different from each other. In this mode, the viscosity vector for each velocity component can be independently specified for each velocity component.

The VISCOSITY, SET and CONDUCTIVITY commands may be specified in combination. In this case, the CONDUCTIVITY commands will take precedence over all other commands.

**REMARKS**

If a constant value is specified for whole of the field, then it is used as reference value. The reference value is used in ANSWER® to compute reference Reynolds number for the flow and to account for the effect of wall in turbulent flow through wall functions.

In PORFLOW® the reference value is used to compute the effect of varying viscosity on the hydraulic conductivity of the matrix. The effective hydraulic conductivity,  $K_{ij}$ , is computed as:

$$K_{ij} = K_{ij}^* \frac{\mu^*}{\mu}$$

Where  $K_{ij}^*$  is the reference value of hydraulic conductivity specified through the **CONDUCTIVITY** or **HYDRaulic** commands, and  $\mu$  and  $\mu^*$  are, respectively, the local and reference values of viscosity.

#### EXAMPLES

---

**VISC**osity for primary fluid:  $\mu^* = 1.002E-3$  (water at 20 deg C)

**VISC**osity UNIFORM SECONdary phase:  $\mu^* = 8E-2$  (Engine oil at 20 deg C)

**VISC**osity for primary fluid:  $\mu^* = 1.002E-3$  for region ID=LOWER\_Domain

**VISC**osity REFERENCE value = 0.544E-3

Generic examples for this command are given in Section 4.4. The command keyword (**VISC**) must replace the keyword used in these examples.

**MODE 2:** Pre-Defined Viscosity Functions

**SYNTAX** VISC {SUTH|POWE|EXPO} [phase] [ $\mu_0$ ] [ $\alpha$ ] [ $\beta$ ] [ $T_a$ ]

**CONS** The fluid viscosity is constant. This is the default option.

**SUTH** Sutherland equation for viscosity of gases is used:

$$\mu = \mu_0 \frac{\alpha + \beta}{T + \beta} \left[ \frac{T}{\beta} \right]^{3/2}$$

For this relation, T must be in absolute units. If no numerical values are specified; then the default values are  $\mu_0=1.827E-5$ ,  $\alpha=291$ ,  $\beta=120$  that provide the viscosity of gaseous air in units of [N.s/m<sup>2</sup>].

**POWE** The special version of empirical Seeton fit (C.J. Seeton, UIUC, 2006) for viscosity of liquids is used:

$$\mu = \mu_0 10^{\left( \frac{\alpha}{T + T_a - \beta} \right)}$$

The default values are  $\mu_0=2.414E-5$ ,  $\alpha=247.8$ ,  $\beta=140$  that provide the viscosity of water in units of [N.s/m<sup>2</sup>] with an accuracy of 2.5% in the temperature range from 0 to 370 °C.

**EXPO** Exponential variation in fluid viscosity according to the equation:

$$\mu = \mu_0 \exp \left[ \frac{\alpha}{T + T_a} \right]$$

If no numerical values are specified; then the default values are  $\mu_0=7.20912E-6$ ,  $\alpha=1436$ . These provide the viscosity of water in units of [N.s/m<sup>2</sup>].

**EFFE** See Mode 1 of Command.

**ADD** See Mode 1 of Command.

**phase** See Mode 2 of Command.

**$\mu_0$**  The 1<sup>st</sup> viscosity constant. See above for default values.

**$\alpha$**  The 2<sup>nd</sup> viscosity constant. See above for default values.

**$\beta$**  The 3<sup>rd</sup> viscosity constant. See above for default values..

**$T_a$**  The absolute temperature base. If no value is specified, the default value is 273.15 except if the compressible mode of ANSWER® Software Tool is invoked. In that case, the temperature must be in absolute units and the default value is 0.

## REMARKS

If these viscosity relations are invoked and no other VISCOSITY command is given to set the reference value (see Mode 1 of command), then the reference value is set to 8.90439E-4 for the POWER and EXPONENTIAL options and equal to the value of  $\mu_0$  for the SUTHERLAND option.

**EXAMPLES**

---

**VISC**osity EXPOnential option

**VISC**osity from SUTHERLAND equation

**VISC**osity EXPOnential option:  $\mu^* = 0.544E-3$ ,  $b = 1450$  K for FIELD values

## 5.123 WALL

**PURPOSE** To specify wall boundary or boundary conditions for a problem. This command is effective only for the ANSWER® Software Tool.

**MODE 1:** Specify Walls at Undefined Outer Boundaries

**SYNTAX** WALL [ORTH] [MOVI] [ROUG=Z<sub>rough</sub>] [TYPE=C<sub>rough</sub>]

**ORTH** By default it is assumed that walls may not be orthogonal to the grid coordinates. Therefore the computation of wall functions and turbulence energy production term requires extensive arithmetic operations. If this modifier is present, then a simpler set of formulae are used on the assumption that the walls are aligned with coordinate directions.

**MOVI** By default all walls and solid blocks are assumed to be stationary. Irrespective of the user input, the initial values of velocity components and, energy, length scale and dissipation of turbulence are all set to zero at all walls and obstacles. This modifier disables this feature so that all non-zero initial values for these variables are retained as such. Independent of this modifier, external moving walls may be specified by the BOUNDARY command.

**ROUG** This mode of the command is operational only for the ANSWER® software. If this modifier is present, then the wall of the block is assumed to be hydro-dynamically rough for turbulent flow. The height and type of roughness is specified by the Z<sub>ROUGH</sub> and C<sub>ROUGH</sub> constants

**Z<sub>ROUGH</sub>** The average roughness height for the wall. It should be noted that the roughness should be small compared to the grid size in the vicinity of the wall. Further the roughness theory is only valid for the range of Y+ values that put it in fully turbulent region of the log-law of the wall.

**TYPE** This mode of the command is operational only for the ANSWER® software. This modifier specifies the nature of the roughness. Roughness can be of many types including those of sand-grain, ribbed, finned and random. A factor in the corresponding roughness relations (C<sub>ROUGH</sub>) accounts for these differences. This modifier is effective only if the modifier ROUG is also present.

**C<sub>ROUGH</sub>** The factor that accounts for roughness type. The default value is 0.5 which is generally considered to be suitable for sand-grain type of roughness. The recommended values generally lie between 0.5 and 1.0.

**REMARKS**

If this command is present then all the outer boundaries of the domain of computation that are not explicitly defined through the INLET, OPEN, OUTLET, PERIODIC and SYMMETRY commands are defined to be walls. At a wall, the turbulence variables are forced to conform to certain preset conditions, the velocities and temperatures are fixed and gradients of the species are set to zero. The defaults for wall temperature and species can be changed by explicit user commands (ADIABATIC and WALL). The wall is assumed to coincide with the element boundary.

**EXAMPLES**

**WALL** by default at external boundaries

**WALL** by default at undefined external boundaries; l walls ORTHogonal

**WALL** by default at undefined external boundaries; ORTHogonal and MOVIng

**WALL** by default at external boundaries ROUGHness height=0.001; TYPE=0.6

**MODE 2:** Specify Boundary Conditions at Walls

**SYNTAX** WALL { $\Phi$ } [FIX|ADIA|GRAD|EXTR]

**$\Phi$**  One or more symbols that denote the dependent variables for which the wall boundary conditions are specified. Valid symbols are listed in Table 2.7.1. There is no default value; a symbol must be specified

**FIX** The wall values of the specified variables are fixed. The actual value may be fixed by the **BOUNDARY**, **INITIAL**, **READ** or **SET** commands

**ADIA** The gradient of the specified variables at the wall is set to zero.

**GRAD** Same as **ADIABATIC**; the gradient of the specified variables at the wall is set to zero.

**EXTR** The specified variables at the wall are computed from linear extrapolation of the values from the nearest in-field locations.

#### REMARKS

---

This command allows the user to set the boundary conditions at the walls globally. Multiple commands may be used to accommodate different requirements for different variables. The wall boundary conditions can also be set by **BOUNDARY** and **ADIABATIC** commands. By default, at a wall, the turbulence variables are forced to conform to certain preset conditions, the velocities and temperatures are fixed and gradients of the species are set to zero

#### EXAMPLES

---

**WALL** present at the SELEcted region; determine orientation from subregion

**WALL** for SELEcted region with orientation of X-

**WALL** for SELEcted region with orientation of Z+

**MODE 3: Specify Individual Wall Location and Boundary Conditions****SYNTAX** WALL {dir} [subrgn] [FIX|ADIA] [ROUG=Z<sub>rough</sub>] [TYPE=C<sub>rough</sub>]**dir** The orientation index for the wall. See Section 3.5 for available choices. There is no default value; a value must be specified.**subrgn** The subregion to be identified as a wall. See Section 3.4. If no subregion is specified, the outermost **dir** oriented boundary of the entire computational domain is selected.**FIX** The wall is assumed to be at fixed temperature. For the external wall, the temperature may be specified by **BOUNDARY**, **INITIAL**, **READ** or **SET** commands. For an internal wall, temperature is assumed to be the average of the fluid temperature prevailing at the two nearest elements that straddle the wall. This specification takes precedence over any global boundary specification by **WALL (Mode 1)** or **BOUNDARY** command.**ADIA** The gradient of the specified temperature at the wall is set to zero.**ROUG** See Mode 1 of command.**Z<sub>ROUGH</sub>** See Mode 1 of command.**TYPE** See Mode 1 of command.**C<sub>ROUGH</sub>** See Mode 1 of command.**REMARKS**

This command may be used either for external walls or infinitesimally thin internal walls. Any thick walls should be specified by the **BLOCKage** command. For an external wall the values of the variables at the boundary can be set by the **BOUNDARY**, **INITIAL**, **READ** or **SET** commands. **For an internal wall, normal gradients of all variables except velocity and temperature are set to zero.** It is assumed that all internal walls are stationary; hence all velocity components are set to zero. The temperature boundary condition may be specified as indicated above.

**EXAMPLES****WALL** with orientation of X-**WALL** at Y+**WALL** for SELEcted region with orientation of Z+**WALL** with orientation of Y+ at ID=OUTEr boundary**WALL** with orientation of Y+ at ID=OUTEr boundary with ADIA boundary

**MODE 4:**      **Open a Segment of a Previously Specified Wall**

**SYNTAX**      **WALL {OPEN} {dir} [subrgn]**

**OPEN**            All previously specified walls are searched and if the boundary segments identified by the **dir** and **subrgn** on the command match any previous wall segments, then the wall segment is removed. This essentially replaces a wall segment by an opening which may be subsequently used for specification of, say, an inlet or outlet.

**dir**                The orientation index for the boundary. See Section 3.5 for available choices. *There is no default value; a value must be specified.*

**subrgn**            The subregion to be removed from an existing wall. See Section 3.4. *If no subregion is specified, the outermost **dir** oriented boundary of the entire computational domain is selected.*

#### REMARKS

---

This command is intended to provide a compact method of opening a hole or slot in a previously specified wall. **It should be noted, that at the time of specification, only the previously specified walls are searched to remove wall segments.** The same segments may be subsequently re-specified by, say, a WALL command.

#### EXAMPLES

---

**WALL OPEN** any previously specified walls at Y+ of the SELEcted region

**WALL OPEN** any previous wall for Y+ of ID=HOLEINWALL

<b>MODE 5:</b>	<b>Drill a Hole in a Previously Specified Wall</b>
<b>SYNTAX</b>	<b>WALL {HOLE} {dir} [subrgn] {d<sub>hole</sub>} [N<sub>1</sub>, . N<sub>n</sub>] [IJK   ELEM] [N<sub>vec1</sub>, . N<sub>vecm</sub>] [ε]</b>
<b>HOLE</b>	Wall segments falling within the range specified by the input are removed from the wall previously specified by the same <b>dir</b> and <b>subrgn</b> modifiers. The hole is circular or elliptic based on the specification of the direction normals for the plane.
<b>dir</b>	The orientation index for an existing wall. See Section 3.5 for available choices. <b>There is no default value; a value must be specified.</b>
<b>subrgn</b>	The subregion designation of an existing wall. See Section 3.4. <b>If no subregion is specified, the outermost dir oriented boundary of the entire computational domain is selected.</b>
<b>d<sub>hole</sub></b>	The diameter of the hole to be “drilled” in the specified wall.
<b>N<sub>1</sub>, . N<sub>n</sub></b>	In the absence of the <b>IJK</b> or <b>ELEM</b> modifier, these specify the (x, y) or the (x, y, z) coordinates of the center of the circular hole. In the presence of <b>IJK</b> or <b>ELEM</b> modifier these are interpreted as given below.
<b>IJK</b>	The numerical input [N <sub>1</sub> , . N <sub>n</sub> ] specifies the grid indices (I, J) or (I, J, K) of the element. The center of the hole is assumed to coincide with the center (node) of the element. 2 values must be specified for 2D and 3 for 3D geometry. <b>This option can be used only for structured grids. Only the internal elements can be specified.</b>
<b>ELEM</b>	The numerical input [N <sub>1</sub> , . N <sub>n</sub> ] specifies the element which is at the center of the hole. Only 1 value must be specified. <b>This option can be used for structured or unstructured grids. Only the internal elements can be specified.</b>
<b>N<sub>vec1</sub>, . N<sub>vecm</sub></b>	The components of a vector normal to the plane in which the center of the hole is located. The values may be in arbitrary units and are internally converted to unit normals. Up to 3 values may be specified. <b>At least one value must be specified. An elliptic hole can be drilled by appropriate choice of the normal vector.</b>
<b>ε</b>	The tolerance in the direction normal to the plane of the hole. The logic employed computes the normal and tangential distance of the existing wall element centers (nodes) from the center of hole in the specified plane. If the tangential distance is ≤ to the radius of the hole, then the wall segment is removed <b>provided its normal distance from the plane is less than the tolerance</b> . By default the tolerance is set to 10 <sup>30</sup> so that all elements in the projected plane of the hole are captured. This input is used to specify a different tolerance to capture only wall segments within a certain distance from the plane. <b>If specified, the intervening values for all 3 components of the unit normal (even if zero) must be specified.</b>

## REMARKS

---

This command is intended to provide a compact method of opening a hole or slot in a previously specified wall. **It should be noted, that only the specified wall is searched to remove wall segments.** The removed segments may be subsequently re-specified by, say, a **WALL** command.

## EXAMPLES

---

```
WALL HOLE dia = 0.018 at (0.012, 0.3) normal (1, 0) in X+ wall of ID=WALLHOLE
WALL HOLE dia = 0.018 at (0.012, 0.3) normal (1, 2) in X+ ID=WALL tol=0.1
WALL HOLE dia = 0.018 at (0.012, 0.3, 0.1) normal (0, 1, 0) in Y+ wall of ID=HOLEINWALL
WALL HOLE dia = 0.01 at IJK (10, 15, 5) normal (0, 1, 0) in Y+ wall of ID=WALL
WALL HOLE dia = 0.01 at ELEMent 989 normal (0, 1, 0) in Y+ wall of ID=WALL
```

**5.124 WIND**

**PURPOSE** To specify wind conditions for a problem. This command is effective only for the TIDAL® Software Tool.

**MODE 1:** Specification of Wind Stress Coefficients and Reference Wind Speed

**SYNTAX** WIND [**C<sub>w1</sub>**, **C<sub>w2</sub>**, **w<sub>0</sub>**]

**C<sub>w1</sub>** Linear component of wind stress coefficient, C<sub>w1</sub>, in the equation given below. The default value is 1.0E-6.

$$C_w = C_{w1} + C_{w2} \left\{ \max \left[ 0, \left( 1 - \frac{w_0}{W} \right)^2 \right] \right\},$$

**C<sub>w2</sub>** Non-linear component of wind stress coefficient, C<sub>w2</sub> in the equation given above. The default value is 1.4E-6.

**w<sub>0</sub>** Reference wind speed, w<sub>0</sub> in the equation given above. The default value is 7.0.

**EXAMPLES**

---

**FRIC**tion parameters for WIND: C<sub>w1</sub> = 2.4E-6, C<sub>w2</sub> = 0.00, w<sub>0</sub> = 0 m/s

**FRIC**tion - WIND parameters: all values are set to zero

**MODE 2:** Specification of Hurricane Wind Distribution According to Holland (2010) model

**SYNTAX** WIND [HOLL] [P<sub>Ext</sub>, P<sub>c</sub>, dP, R<sub>m</sub>] [LATI= φ] [TEMP= T<sub>ss</sub>] [DENS= ρ<sub>v</sub>] [ANTI|CLOC]

**HOLL** Holland (2010) model is used to compute the wind distribution for a Hurricane. The wind velocity at surface level (10m) is computed as:

$$v_s = \sqrt{\xi_s e^{(1-\xi_s)}}; \quad \xi_s = \left(\frac{r_{vms}}{r}\right)^{b_s}$$

$$v_{ms} = \sqrt{\frac{b_s}{e} \frac{100 \Delta p_s}{\rho_{ms}}}; \quad \Delta p_s = p_{ns} - p_{cs}$$

$$b_s = 1 - 4.4 \times 10^{-5} \Delta p_s^2 + 0.01 \Delta p_s + 0.03 \frac{\partial p_{cs}}{\partial t} - 0.014 \varphi + 0.15 v_t^x$$

$$x = 0.6 \left(1 - \frac{\Delta p_s}{215}\right)$$

$$\rho_s = \frac{100 p_s}{R T_{vs}}; \quad T_{vs} = (T_s + 273.15)(1 + 0.61 q_s); \quad T_s = T_{ss} - 1$$

$$p_s = p_{cs} + \Delta p_s e^{\xi_s}; \quad q_s = RH_s \left(\frac{3.802}{100 p_s}\right) e^{\frac{17.67 T_s}{243.5 + T_s}}$$

In the equation above,  $v_s$  is the surface wind speed (m/s),  $b_s$  is the shape parameter for the radial profile,  $\Delta p_s$  is the pressure drop (hPa),  $p_{ns}$  is the external pressure (hPa),  $p_{cs}$  is the central pressure (hPa),  $\partial p_{cs} / \partial t$  is the intensity change in (hPa per hour),  $\varphi$  is the latitude in degrees,  $v_t^x$  is the cyclone translation velocity (m/s),  $R$  is the gas constant for dry air (J/kgK),  $T_{vs}$  is the virtual surface temperature (K),  $q_s$  is the surface moisture in (g/kg),  $RH_s$  is surface relative humidity,  $T_s$  is the surface temperature (°C),  $r_{vm}$  is the radius at maximum winds (km) and  $T_{ss}$  is the sea surface temperature (°C).

**P<sub>c</sub>** The pressure at the center (eye) of the hurricane (hPa)

**dP** The pressure intensity (hPa per hour),

**R<sub>m</sub>** The radius to maximum winds (km)

**LATI** If this modifier is present, then the first numerical value immediately following this modifier specifies the latitude of the hurricane.

**φ** The latitude in degrees

**TEMP** If this modifier is present, then the first numerical value immediately following this modifier specifies the sea surface temperature

**T<sub>ss</sub>** The sea surface temperature (°C) if the **TEMP** modifier is present.

**DENS** If this modifier is present, then the first numerical value immediately following this modifier specifies the density of air at the maximum velocity.

**ρ<sub>v</sub>** The density of air at the maximum velocity (kg/m<sup>3</sup>) If the **DENS** modifier is present.

**ANTI** The winds are generated in an anti-clockwise rotation; **this is the default option.**

**CLOC** The winds are generated with a clockwise rotation around the hurricane eye.

**REMARKS**

---

This model for calculating the wind speed at a given radius is taken from the paper by G.J.Holland, J.I.Belanger and A.Fritz (2010): A Review Model for Radial Profiles of Hurricane Winds, American Meteorological Society, 2010, p 4393-4401.

**EXAMPLES**

---

**WIND HOLLAND** pext=1005, Pc=950, dpdt=3, Rm=20, LATI=20, TEMP=20, DENS=1.18

**MODE 3:**      **Specification of Trajectory of Hurricane for Holland (2010) model**

**SYNTAX**      **WIND [TRAJ] [V<sub>Eye</sub>,  $\theta$ , X<sub>o</sub>, Y<sub>o</sub>]**

**TRAJ**            Trajectory of the hurricane is specified.

**V<sub>Eye</sub>**            The eye speed of the hurricane (m/s)

**$\theta$**                 The angle that the trajectory makes with the positive direction of the x-axis (degrees).

**X<sub>o</sub>, Y<sub>o</sub>**          The starting (x, y) coordinates of the eye of the hurricane (m)

**EXAMPLES**

---

**WIND TRAJECTORY** Veye=**5**, theta=**45**, starting (x,y)=(**0, 0**)

## 5.125 WRITE

**PURPOSE** To generate output of selected variables on demand.

**MODE 1:** Write Selected Variables to Standard Output File or User-Specified File

**SYNTAX** WRIT { $\Phi$ }[ [MULT] | [SCALE|STAC|UNIT|ELEM] ] [N][ $\alpha$ ] [ $\beta$ ] [fname] [subrgn]

**$\Phi$**  One or more of the symbols that represent the variables for which output is desired. The valid symbols are listed in Tables 2.8.1-3 and 2.8.4. There is no default value.

**MULT** By default the output for each  $\Phi$  appears in a separate table. Each record in the table consists of node number in the 1<sup>st</sup> column, followed by the node coordinate values (x, y for 2D) or (x, y, z for 3D) and then the value of the variable. If this modifier is present, then multiple variables are listed in the same table. The first 3 or 4 columns list the node number and the x, y and z coordinates as described above. These are followed by one column for each of the  $\Phi$ 's. Up to 17 columns (including the node number and the coordinates) appear in the same table. If more than 14 (for 2D) or 13 (for 3D) variables are specified then the remaining variables appear in supplemental tables in the same format.

**SCAL** All output variables,  $\Phi$ 's, are scaled with  $\alpha$ ,  $\beta$  according to the relation:

$$\Phi_{\text{out}} = \alpha \Phi_{\text{computed}} + \beta$$

**STAC** The output variable is scaled by coefficients **a** and **b** specified on a previous **STACK WRITE** command according to:

$$\Phi_{\text{out}} = a \Phi_{\text{computed}} + b$$

In this mode each variable can be scaled by its own **a**, **b** as specified by previous **STACK WRITE** commands. See also comments below for further clarification.

**UNIT** If this modifier is specified then the each output variable is scaled to the range of zero to unity according to the relation:

$$\Phi_{\text{out}} = \Phi_{\text{min}} + (\Phi_{\text{computed}} - \Phi_{\text{min}}) / (\Phi_{\text{max}} - \Phi_{\text{min}})$$

Here  $\Phi_{\text{min}}$  and  $\Phi_{\text{max}}$  are, respectively, the minimum and maximum computed values of the  $\Phi$  variable in the whole of the computational domain.

**ELEM** If this modifier is specified then the each output variable is in reference to the value of that variable at the element number specified by the input **N**; that is:

$$\Phi_{\text{out}} = \Phi_{\text{computed}} - \Phi_{\text{N}}$$

Here  $\Phi_{\text{N}}$  is the value of  $\Phi$  at element number denoted by **N**

**N** The element number for the reference element location if **ELEM** modifier is present. This input must be omitted if the **ELEM** modifier is not present.

**$\alpha$**  The scaling factor for the variables if the **SCAL** modifier is specified. This input must be omitted if the **SCAL** modifier is not present.

**$\beta$**  The additive factor for the variables if the **SCAL** modifier is specified. This input must be omitted if the **SCAL** modifier is not present.

**fname** The name of the file to which the output is directed. See Section 3.3 for additional information. If no file name is specified, then the variables are written to the Standard Output file

**subrgn** The subregion for computations. If no subregion is specified, the entire domain is selected.

**REMARKS**

This command provides an alternative mode of output in comparison with the **OUTPUT** command. The output is obtained in the form of a list. Where possible the list is organized by element or node number. Output is provided as soon as the command is encountered.

The **SCAL**, **STAC**, **UNIT** and **ELEM** modifiers are effective only for **real** variables. These are ignored for integer variables. The **STAC** modifier is also ignored if no previous **STACK WRITE** command was specified.

**EXAMPLES**

---

**WRITe** V and T

**WRITe** V and T to 'file.VT'

**WRITe** U, V, P and T in SELEcted subregion to 'FLOW.FIL'

**WRITe** U, V, P and T in region ID=SUBSET1 to 'FLOW.FIL'

**WRITe** U, V, P and T after scaling with STAC for region ID=SUBSET1 to 'FLOW.FIL'

**WRITe** V, T, NFACE, NBRS, MTYP and FC to 'output.mix'

**MODE 2:** Write Boundary Specific Variable Values and Statistics.

**SYNTAX** WRIT { $\Phi$ } [option] [walltype] [SUMM] [SCALE|STAC|UNIT|ELEM] [N| $\alpha$ ] [ $\beta$ ] [subrgn] [dir] [fname]

$\Phi$  One or more of the symbols that represent the variables for which output is desired. The valid symbols are those listed in Table 2.8.1-3 and Table 2.8.4. There is no default value.

option The boundary type for which output is required. Only one option may be selected for each command.

option	INTERPRETATION
<b>INLE</b>	Only the boundaries specified by the <b>INLET</b> command are selected.
<b>OUTL</b>	Only the boundaries specified by the <b>OUTLET</b> command are selected.
<b>OPEN</b>	Only the boundaries specified by the <b>OPEN</b> command are selected.
<b>IO</b>	All boundaries specified by <b>INLET</b> , <b>OUTLET</b> or <b>OPEN</b> commands are selected.
<b>BOUN</b>	All external boundaries of the specified (or default) subregion are selected.
<b>SYMM</b>	Only the boundaries specified by the <b>SYMMETRY</b> command are selected.
<b>WALL</b>	Only the walls are selected.

walltype The type of wall to be selected for output if the **WALL** modifier is present.

walltype	INTERPRETATION
<b>ALL</b>	All walls are selected. This is the default option.
<b>EXTE</b>	Only the exterior walls of the computational domain are selected.
<b>INTE</b>	Only the walls located in the interior of the computational domain are selected.
<b>BLOC</b>	Only the walls of the blocks ( <b>BLOCK</b> command) are selected.

**SUMM** By default, the output generated by this command includes the detailed information for each element of the boundary as well as an overall summary for the boundary as a whole. If this modifier is present, then only the summary information is generated; the element by element details are suppressed.

**SCAL** See Mode 1 Specification.

**STAC** See Mode 1 Specification.

**UNIT** See Mode 1 Specification.

**ELEM** See Mode 1 Specification.

**N** See Mode 1 Specification.

$\alpha$  See Mode 1 Specification.

$\beta$  See Mode 1 Specification.

**subrgn** The subregion for computations. If no subregion is specified, the entire domain is selected.

**dir** The orientation index for the boundary. See Section 3.5 for available choices. If no input is given, then the output is obtained for all boundaries of the selected type.

**fname** The file name for output. By default the output is printed only to the standard output device. If a file name is specified, then the output is printed to the named file.

**EXAMPLES**

---

**WRITE** T and RHO for all INLEt boundaries

**WRITE** T and RHO only for INLEt boundaries; scale output with STACK coefficients

**WRITE** T and RHO only for INLEt boundaries in X- direction to file 'INLETX.FIL'

**WRITE** T and RHO only for INLEt boundaries defined by ID=INLEt in X- direction.

**WRITE** SUMMery for U, T and RHO for INLEt boundaries defined by ID=INLEt in X- direction.

**WRITE** U, V, P, T and RHO for all OUTLEt boundaries in X+ direction to 'OUTLET.VAL' file

**WRITE** T and RHO only for WALLs to the file 'WALLS.PRB'

**WRITE** T and RHO only for EXTErIOR WALLs to the file 'WALL\_EXT.PRB'

**WRITE** T and RHO only for axis of SYMMetry in Y- direction.

**WRITE** T for all external BOUNDaries.

**MODE 3:** Write a Profile of Values at Selected Locations

**SYNTAX** WRIT {Φ} [INTE] [method] [STAC] [subrgn] [fname]

**Φ** One or more of the symbols that represent the variables for which output is desired. Up to 9 symbols may be specified on one command. The valid symbols are those listed in Table 2.8.1-3 and Table 2.8.4. There is no default value.

**INTE** The variable values at the selected locations are computed by linear or inverse distance squared interpolation from the computed values at the nearest neighboring nodes.

**method** The method of interpolation for computing values at the specified (x, y, z) locations. Only one method may be selected for each command.

method	INTERPRETATION
<b>LINE</b>	The values are computed by inverse linear distance interpolation from the computed values at nearest neighbors. This is the default option.
<b>SQUA</b>	The values are computed by inverse distance squared interpolation from the computed values at nearest neighbors.
<b>AVER</b>	The values are computed by arithmetic average of computed values at nearest neighbors.
<b>NEAR</b>	The value is set equal to that at the nearest neighbor.
<b>X</b>	Same as <b>LINE</b> except that the distance is set equal to the separation in the x-coordinate.
<b>Y</b>	Same as <b>LINE</b> except that distance is set equal to the separation in the y-coordinate.
<b>Z</b>	Same as <b>LINE</b> except that the distance is equal to the separation in the z-coordinate.

**STAC** The output variable is scaled by coefficients **a** and **b** specified on a previous **STACK WRITE** command according to:

$$\Phi_{out} = a \Phi_{computed} + b$$

The **STAC** modifier is effective only for **real** variables. It is ignored if no previous **STACK WRITE** command was specified.

**subrgn** The sub region for output. The sub region must have been previously specified by a **LOCATE LIST COORDINATE** command. Any other sub region specification will lead to an error condition. There is no default value.

**fname** The file name for output. By default the output is printed only to the standard output device. If a file name is specified, then the output is printed to the named file.

**EXAMPLES**

```
WRITe U, V, P and T by INTErpolation for ID=L_PROFILE
WRITe U, V, P and T by SQUAred INTErpolation for ID=L_PROFILE
WRITe U, V, P and T by INTErpolation for ID=L_PROFILE
WRITe U by AVERAGE INTErpolation for ID=L_PROFILE on file: 'PROFILE.U'
WRITe U, V, P and T by X direction INTErpolation for ID=L_PROFILE on file: 'PROFILEX.VAR'
```

**MODE 4:** Write Selected Variables to User-Specified File in Block Mode

**SYNTAX** **WRITE** { $\Phi$ } {**BLOC**} [**HEAD**] [**FIEL**] {fname}

**$\Phi$**  One or more of the symbols that represent the variables for which output is desired. The valid symbols are those listed in Table 2.8.1-3, and the node or element-based variables listed in Table 2.8.4. There is no default value.

**BLOC** The variables are written in the block format. One record is written for each variable. If the grid is structured then the variable is written in the manner of the FORTRAN DO loop (over the grid indices I, J, K). If the grid is unstructured, then the record for the variable is sequential over element numbers starting with the 1<sup>st</sup> element.

**HEAD** By default only the numeric values for the selected variable(s) are written to the file without any header information. If the **HEADER** modifier is present, then a two line header in the standard ACRi **SAVE** file format appears before each set of variable values.

**FIEL** Only the inner field nodes or elements are written to the output file. The boundary nodes are omitted. This modifier is active only if the **BLOCK** modifier is simultaneously present. By default both the field and the boundary elements are written to the record.

**fname** The name of the file to which the output is directed. See Section 3.3 for additional information. There is no default file name; a file name must be specified.

#### EXAMPLES

---

**WRITE** T in BLOCK format to file named 'value.T'

**WRITE** U, V and W in BLOCK format to file 'value.UVW'

**WRITE** U, V, W, T, C, FU in BLOCK format to file named 'mixed.val'

**WRITE** **FIELd** values of U and X in BLOCK format 'UandX.val'

**WRITE** in BLOCK format values of X Y Z and MYTP to 'geometry.val'

**MODE 5:** Write Vertex Coordinates to User-Specified File

**SYNTAX** WRIT {VERT} {fname}

**VERT** A file containing the (x, y, z) coordinates of the vertices is generated. The file contains header information followed by a table of vertex numbers and the corresponding grid coordinates. The vertex data in the file is generated by the FORTRAN STATEMENTS:

```
DO N = 1, Total_Vertices
  WRITE (IFILE,*) N, (XV (K), K=1, N23D)
ENDDO
```

Here IFILE is an internally assigned file unit number, XV are the coordinates of the vertices and, N23D is 2 for 2D and 3 for 3D geometry.

**fname** The name of the file to which the output is directed. See Section 3.3 for additional information. There is no default file name; a file name must be specified.

#### EXAMPLES

---

WRITe VERTices on file='VERTICES.XYZ'

**MODE 6:** Write Element Connectivity to User-Specified File

**SYNTAX** WRIT {CONN} {fname}

**CONN** A file containing the element to vertex connectivity is generated. **The command must be given before the first SOLVE command otherwise an empty file will be generated.** The file contains header information followed by a table of element numbers and the corresponding vertices that define the element. The connectivity data in the file is generated by the FORTRAN STATEMENTS:

```
DO M = 1, Total_Elements
  MVLO = NVRTXLO (M)
  MVHI = NVRTXHI (M)
  WRITE (IFILE,*) M, (NVRTX (K), K=MVLO, MVHI)
ENDDO
```

Here IFILE is an internally assigned file unit number and NVRTXLO and NVRTXHI are the starting and ending locations of the vertices for the m<sup>th</sup> element.

**fname** The name of the file to which the output is directed. See Section 3.3 for additional information. *There is no default file name; a file name must be specified.*

#### EXAMPLES

---

WRITe CONNectivity on file='VERTICES.CNC'

**MODE 7:** Write Element Vertex Coordinates to User-Specified File

**SYNTAX** WRIT {CORN} {fname}

**CORN** A file containing the (x, y, z) coordinates of the vertices for each element is generated. **The command must be given before the first SOLVE command otherwise an empty file will be generated.** The file contains header information followed by a table of element number, vertex number connected to that element, and the corresponding coordinates for the vertex. The vertex data in the file is generated by the FORTRAN STATEMENTS:

```
DO M = 1, Total_Elements
  DO N = 1, Vertices_for_this_Element
    WRITE (IFILE,*) M, N, XC (N), YC (N), ZC (N)
  ENDDO
ENDDO
```

Here IFILE is an internally assigned file unit number, and XC, YC and ZC are, respectively, the (x, y, z) coordinates of the vertex. The vertex numbers for the element are in the same order as those on the **CONNECTIVITY** command.

**fname** The name of the file to which the output is directed. See Section 3.3 for additional information. **There is no default file name; a file name must be specified.**

#### EXAMPLES

---

WRITe CORNers of the element vertices on 'CORNER.XYZ'

## 5.126 ZONE

**PURPOSE** To specify the material or zone type for the soil or rock formation. This command is effective only for the **PORFLOW®** Software Tool. This command has been retained only for backward compatibility. It is no longer necessary to use this command. Its function has been superseded by the **LOCATE** command.

**MODE 1:** Specification of Material Type

**SYNTAX** ZONE {N<sub>1</sub>} {N<sub>2</sub>, ..., N<sub>n</sub>} [N<sub>n+1</sub>, N<sub>n+2</sub>]

**MODE 2:** Material Type Data Input from a File

**SYNTAX** ZONE {fname}

**REMARKS**

---

This command is retained primarily to provide compatibility for data sets prepared with previous versions of **PORFLOW®**. Mode 1 and Mode 2 of the **MATERIAL**, **ROCK** or **SOIL** commands, together with **LOCATE** command, have replaced its function. These latter commands should be used..

**EXAMPLES**

---

See Mode 1 of the **MATER**ial command.

## REFERENCES

- F. Adenot and M. Buil, 1992.** Modelling of the corrosion of the cement paste by deionized water. *Cement and Concrete Research* **22**, (2-3), p. 489
- B. Abramzon and W.A. Sirignano, 1989.** Droplet Vaporization Model for Spray Combustion Calculations, *International Journal of Heat and mass Transfer*, **32**, pp. 1605-18.
- Bary, Benoît and Antoine Bregeon, 2003.** Etude de la carbonatation des matériaux cimentaires en conditions saturées:: Modélisation simplifiée et simulations numériques. *Réf. CEA, NT DPC/SCCME 03-230-A Novembre 2003.*
- J. Bear, 1972.** *Dynamics of Fluids in Porous Media*. American Elsevier, **New York, New York.**
- J.D. Bittener, and J.B. Howard, 1978.** Role of Aromatics in Soot Formation, *Alternate Hydrocarbon Fuels: Combustion and Chemical Kinetics*, Ed. **C.T. Bowman and J. Birkeland**, Academic Press, **New York.**
- N.E. Bixler, 1985.** NORIA-A Finite Element Computer Program for Analyzing Water, Vapor, Air, and Energy Transport in Porous Media, *Report No. SAND84-2057, UC-70, Sandia National Laboratories, Albuquerque, New Mexico 87185.*
- K.N.C. Bray, P.A. Libby and J.B. Moss, 1985.** *Unified Modeling Approach for Premixed Turbulent Combustion - Part I: General Formulation*, *Combustion and Flame*, **61**, pp. 87-102.
- R.H. Brooks and A.T. Corey, 1966.** Properties of porous media affecting fluid flow. *J. Irrig. Drain. Div., Am. Soc. Civil Eng.* **92** (IR2), pp. 61-88.
- N.T. Burdine, 1953.** Relative permeability calculations from pore-size distribution data. *Petr. Tans., Am. Inst. Mining Metall. Eng.*, **198**, pp. 71-77.
- S. Candel, D. Veynante, F. Lacas, E. Maistret, N. Darabiha and T. Poinso, 1990.** Flamelet Descriptions of Turbulent Combustion. *Invited Lecture, 9th International Heat Transfer Conf., Aug. 19-24, 1990, Jerusalem, Israel.*
- K.Y. Chien, 1982.** Predictions of Channel and Boundary-Layer Flows with a Low-Reynolds-Number Turbulence Model, *AIAA*, **20**, 1, p 33-38.
- A.J. Chorin, 1967.** A Numerical Method for Solving Incompressible Viscous Flow Problems, *J. Computational Physics*, *Vol. 2*, pp. 12-26.
- T.J. Craft, N.Z. Ince and B.E. Launder, 1996.** Recent developments in second-moment closure for buoyancy-affected flows. *Dynamics of Atmospheres and Oceans*, **23**, p.99-114.
- J. DeRis, 1979.** Fire Radiation - A Review, *Seventeenth Symposium (International) on Combustion*, *The Combustion Institute, Pittsburgh, PA.*
- R. Edelman, J. Boccio and G. Weilerstein, 1973.** The Role of Mixing and Kinetics in Combustion Generator NO<sub>x</sub>, *AIChE Sym. on Control of NO<sub>x</sub> Emissions in Direct Combustion Power Sources.*
- D.K. Edwards and A. Balakrishnan, 1973.** Thermal Radiation by Combustion Gases, *Int. J. Heat Mass Transfer*, **16**, pp. 25-40.
- R.A. Freeze and J.A. Cherry, 1979.** *Groundwater*, **Prentice-Hall, Englewood Cliffs, New Jersey.**
- A.G. Gaydon and H.G. Wolfhard, 1979.** *Flames*, **Fourth Edition, Chapman and Hall Ltd., London.**
- William R. Goodin, D. Ian Austin and A. K. Runchal, 1980.** A Model Verification and Prediction Study of SO<sub>2</sub>/SO<sub>4</sub> Concentrations in the San Francisco Bay area. *Proc. Second Joint Conf. on Applications of Air Pollution Meteorology, AMC/APCA, New Orleans, LA, March 24-27, pp. 328-333.*
- W. R. Goodin, A. K. Runchal and G.Y. Lou, 1982.** The Random-Walk Advection and Dispersion Model (RADM). *AIChE, Environmental Progress*, **1**(3), p. 178-181.
- S. Gordon and B.J. McBride, 1971.** Computer Program for Calculation of Complex Chemical Equilibrium Compositions, Rocket Performance, Incident and Reflected Shocks and Chapman-Jouguet Detonations, *NASA SP-273.*
- H.C. Hamaker, 1947.** Radiation and Heat Conduction in Light-Scattering Material, *Phillips Research Report, Volume 2*, pp. 55-67.

- A. Harten, 1983.** High Resolution Schemes for Hyperbolic Conservation Laws. **Journal of Computational Physics**, Vol. 49, pp. 357-393.
- A. Harten, 1984.** On a Class of High Resolution Total-Variation-Stable Finite-Difference Schemes, **SIAM J. Num. Anal.**, Vol. 21, pp. 1-23.
- D.J. Hautman, F.L. Dryer, K.P. Schug and I. Glassman, 1981.** A Multiple-Step Overall Kinetic Mechanism for the Oxidation of Hydrocarbons, **Comb. Sci. Tech.**, 25, pp. 219-235.
- B.S. Haynes and H.G. Wagner, 1981.** **Progress in Energy and Combustion Science**, 7, 229 pp.
- K.H. Homann, 1967.** Carbon Formation in Premixed Flames, **Combustion and Flame**, 11, pp. 265-287.
- H. Iacovides, B.E. Launder, and Y. Li, 1996.** Application of a reflection-free DSM to turbulent flow and heat transfer in a square-sectioned U-bend. **Experimental Thermal and Fluid Science**, 13, 4, p 419-429.
- W.P. Jones and J. H. Whitelaw, 1982.** Calculation Methods for Reacting Turbulent Flows: A Review. **Combustion and Flame**, 48, p 1-26.
- K.C. Karki, 1986.** A Calculation Procedure for Viscous Flows at All Speeds in Complex Geometries, **Ph.D. Dissertation, Univ. of Minnesota.**
- W.O. Kemper and J. C. van Schaik, 1966.** Diffusion of Salts in Clay-Water Systems. **Soil Sc. Am. Proc.** 30:534-540.
- B.E. Lander and B.I. Sharma, 1974.** Application of the Energy Dissipation Model of Turbulence to the calculation of Flow Near a Spinning Disc. **Letters in Heat and Mass Transfer**, 1, 2, p 131-138.
- B.E. Launder, G.J. Reece and W. Rodi, 1975.** Progress in the development of a Reynolds Stress Turbulence Closure. **Journal of Fluid Mechanics**, 68,3, p. 537-566.
- B.E. Launder and D.B. Spalding, 1972.** **Lectures in Mathematical Models of Turbulence**, Academic Press.
- B. Leckner, 1972.** Spectral and Total Emissivity of Water Vapor and Carbon Dioxide, **Combustion and Flame**, 19, pp. 33-48.
- S. Lee, H-J. Kim and A.K. Runchal, 2003.** Large Eddy Simulation of Unsteady Flows in Turbomachinery, 2003, submitted to the **J. Power and Energy**.
- S. Lee and W.C. Meecham, 1996.** Computation of Noise from Homogeneous Turbulence and a Free Jet, **International Journal Acoustics and Vibration**, 1996, 1, pp. 35-47.
- B.P. Leonard, 1979.** A Stable and Accurate Convective Modeling Procedure based on Quadratic Upstream Interpolatio, **Comp. Methods Appl. Mech. Eng.**, 19,59-98.
- B.F. Magnussen, B.H. Hjertager, J.G. Olson and D. Bhaduri, 1978.** Effects of Turbulent Structure and Local Concentrations on Soot Formation and Combustion in C<sub>2</sub>H<sub>2</sub> Diffusion Flames, **Seventeenth Symposium (International) on Combustion**, The Combustion Institute, Pittsburgh, PA.
- M. Mainguy, C. Tognazzi, J.-M. Torrenti and F. Adenot, (2000).** Modelling of leaching in pure cement paste and mortar. **Cement and Concrete Research** 30, (1), p. 83
- F.R. Menter, J.C. Ferreira, T. Esch and B. Konno, 2003.** The SST Turbulence Model with Improved Wall Treatment for Heat Transfer Predictions in Gas Turbines. **IGTC2003-TS-059** , **Proceedings of the International Gas Turbine Congress 2003 Tokyo**.
- A.T. Modak, 1979.** Radiation from Products of Combustion, **Fire Research**, 1, pp. 339-361.
- H.C. Mongia and R.S. Reynolds, 1978.** Combustor Design Criteria Validation, Volume III - User's Manual, Report # USARTL-TR-78-55C, U.S. Army Res. and Tech. Lab., Fort Eustis, VA.
- Y. Mualem, 1976.** A new model for predicting the hydraulic conductivity of unsaturated porous media. **Water Resources Res.** 12, 3, pp. 513-522.
- J. Nagle and R.F. Strickland-Constable, 1962.** Oxidation of Carbon Between 1000-2000 °C, **Proc. Fifth Conf. on Carbon**, Volume I, Pergamon Press, New York.
- J. Nikuradse, 1937.** Laws of flow in rough pipes, **NACA Tech. Mem.**1292.
- H.B. Palmer and C.F. Culliss, 1965.** The Formation of Carbon from Bases, Chemistry and Physics of Carbon, Volume I, Ed. P.L. Walker, Marcell Dekker, New York.
- S.K. Park and K.W. Miller, 1988.** Random Number Generators: Good Ones Are Hard To Find.

Communications of the ACM, October 1988, pp. 1192-1201.

S.V. Patankar and D.B. Spalding, 1972. A Calculation Procedure for Heat, Mass and Momentum Transfer in Three-Dimensional Parabolic Flows, *Int.J.Heat Mass Transfer*, 15, p. 1787.

R.H. Perry and C.H. Chilton, 1973. Chemical Engineers' Handbook, 5th Edition, McGraw Hill, New York.

S.B. Pope and S.M. Correa, 1986. Joint PDF Calculations of non-Equilibrium Turbulent Diffusion Flame, Paper # 53, 21<sup>st</sup> Sym. (International) on Combustion, Munich, August 3-8.

W.E. Ranz and W.R. Marshall, 1952. Evaporation from Drops, *Chemical Engineering Progress*, 48, pp. 141-46 and pp. 173-180.

C. Richet, C. Gallé, P. Le Bescop, H. Peycelon, S. Béjaoui, I. Tovenà, I. Pointeau, V. L'Hostis and P. Lovera, (2004). Synthèse des connaissances sur le comportement à long terme des bétons. Applications aux colis cimentés. CEA-R-6050.

P. J. Roache, 1972. *Computational Fluid Dynamics*, Hermosa Publishers, Albuquerque, New Mexico.

A.K. Runchal, 1972. Convergence and Accuracy of Three Finite Difference Schemes for a Two-Dimensional Conduction and Convection Problem, *Int. J. Num. Meth. Eng.*, 4, pp. 541-550.

A.K. Runchal, 1977. Comparative Criteria for Finite Difference Formulations for Problems of Fluid Flow, *Int. J. Num. Methods in Eng.*, 11, pp. 1667-1679.

A.K. Runchal, 1980. A Random Walk Atmospheric Dispersion Model for Complex Terrain and Meteorological Conditions. 2nd AMS/APCA Joint Conf. on Applications of Air Pollution Meteorology, New Orleans, LA, March 24-27, p. 430-437.

A.K. Runchal, 1987a. Theory and Application of the PORFLOW Model for Analysis of Coupled Fluid Flow, Heat and Radionuclide Transport in Porous Media, in *Coupled Processes Associated with Nuclear Waste Repositories*, pp. 495-516, Academic Press, 1987, Ed. C-F. Tsang.

A.K. Runchal, 1987b. CONDIF: A Modified Central-Difference Scheme for Convective Flows, *Int'l J. Num. Methods in Engg.*, 24, 1593-1608.

A.E. Scheidegger, 1961. General Theory of Dispersion in Porous Media, *J. Geophys. Res.*, 66, pp. 3273-3278.

A. Sarkar and R.M.C. So, 1997. A Critical Evaluation of Near-Wall Two-Equation Models Against Direct Numerical Simulation Data, *Int. J. Heat & Fluid Flow*.

D.B. Schein, 2003. Computation of Exhaust Mixing Noise Using Large-Eddy Simulation Turbulence Modeling and Lighthill's Acoustic Analogy. Ph.D. Thesis, University of California, Los Angeles.

T.H. Shih, J. Zhu, W. Liou, K.-H. Chen, N.-S. Liu and J.L. Lumley, 1997. Modeling of Turbulence Swirling Flows. NASA Tech. Memo. 113112, ICOMP-97-08, CMOTT-97-03.

R.G. Siddall, 1972. Flux Methods for the Analysis of Radiant Heat Transfer, *Fourth Symp. on Flames and Industry*, British Flame Research Committee and the Institute of Fuel, Imperial College, London, September.

J. Smagorinsky, 1963. General Circulation Experiments with the Primitive Equations, Part I: the Basic Experiment, *Monthly Weather Rev.*, 91, pp. 99-164.

D.B. Spalding, 1971. Mixing and Chemical Reaction in Steady Confined Turbulent Flames, *Thirteenth Symposium (International) on Combustion*, The Combustion Institute, Pittsburgh, PA.

D.B. Spalding, 1972. A Novel Finite-Difference Formulation for Differential Expressions Involving Both First and Second Derivatives, *Int. J Num, Methods Eng.*, 4, p. 551-559

S.K. Srivatsa, 1983. Computation of Soot and NO<sub>x</sub> Emissions from Gas Turbine Combustors, NASA CR-167930, NASA, Cleveland, OH.

J.C. Street and A. Thomas, 1955. Carbon Formation in Pre-Mixed Flames, *Fuel*, 34, pp. 4-36.

C. Tognazzi, 1998. Couplage fissuration-dégradation chimique dans les matériaux cimentaires caractérisation et modélisation. Thèse de Ph D, INSA Toulouse.

J.M. Torrenti, O. Didry, J.P. Ollivier and F. Plas, 1999. La dégradation des bétons, couplage fissuration-dégradation chimique. HERMES Sciences Publications, Paris.

R. van Genuchten, 1978. Calculating the unsaturated hydraulic conductivity with a new closed-form

analytical model, **Report 78-WR-08, Water Resources Program, Department of Civil Engineering, Princeton University, Princeton, NJ 08540.**

**R.S. Varga, 1962. *Matrix Iterative Analysis*, Prentice-Hall, Englewood Cliffs, New Jersey.**

**N.B. Vargaftik, 1975. *Tables of Thermophysical Properties of Liquids and Gases*, 2nd Ed., John Wiley & Sons, New York, NY.**

**J.A. Wheeler, 1973.** Simulation of heat transfer from a warm pipeline buried in permafrost, **Paper 27b, 74th National Meeting, American Institute of Chemical Engineers, New Orleans, Louisiana, 11-15 March.**

**D.C. Wilcox, 2006. *Turbulence Modeling for CFD* (Third Edition). DCW Industries, Inc., La Canada, CA 91011**

**Z. Yang and T.H. Shih, 1993.** A New Time Scale Based k-e Model for Near Wall Turbulence, **AIAA, 31, 7, p 1191-1198.**

*(This page left intentionally blank.)*

**APPENDIX A: FILES PROCESSED DURING EXECUTION****A.0 INTRODUCTION**

ACRi Software Tools read a number of files during execution. These help in customizing the installation and managing the progress of the simulation. These are described below.

## A.1 ACRNAMES.ACR

ACRi Software Tools define a number of phase space variables by default. These names are contained in the file called **ACRINIT.ACR**. The user can define and add new phase space variables by the **ALLOCATE** command. If a number of variables need to be always defined for an installation or for a project then these can be included in a file named "**acrNAMES.ACR**". The installation and the working directories are searched for this file at the start of simulations. If this file is present, then these phase space variables are automatically defined. If the file is present in both the installation and the working directory, then the one in the working directory is included. The typical file format is given below.

```
T199 – Technetium 199
I129 - Iodine 129
VRBL1 – 1st Variable to be defined
VRBL0002 – 2nd variable to be defined
```

With these as the contents of the file, 4 new phase space variables will be automatically defined for each simulation by default as if these were included in the **ACRINIT.ACR** file. Following the variable symbol convention, the symbols for these variables will be constructed from the first "word" of up to 8 characters that appears in the definition before a separator (a space or a "-" in this case). Thus for the above list the symbols will be T199, I129, VRBL1 and VRBL0002.

## A.2 ACRDFLT.ACR

If a file named '**acrDFLT.ACR**' is present in the working directory, then this file is automatically included at the start of simulation. This file is assumed to consist of a number of **ACRi FREEFORM™** commands. It is automatically included at the head of the user input command file. As far as the **ACRi** Software Tool is concerned, this is equivalent to the user having inserted these commands into the input command file.

the purpose of this file is to allow project specific "macro" commands, such as for a given geometry of a reaction system to be automatically included. For example for project control, it may be desirable to have some common project specific files that define some basic components of the simulation and need to be standardized across all users. These can be spun off to a separate "read-only" file that is include for all users of that directory.

### A.3 ACRINSRT.ACR

If at time during the simulation, a file named '**acrINSRT.ACR**' is present in the working directory, then it is automatically inserted in the input command file. **ACRi** Software searches for this file at the start of the each time (or iteration) step of the solution process. As soon as this file is detected, it is inserted in the input command file and the file is deleted. The search continues however, if the file is detected again, it is again included.

This file provides a mechanism for including commands that were omitted for one or reason or another at the launch of the simulation or it can used to modify the input (such as relaxation factors, matrix solvers, boundary conditions or output requirements) during the simulation process.

#### A.4 ACRSTOP.ACR

If at time during the simulation, a file named '**acrSTOP.ACR**' is detected in the working directory, then the simulation is automatically terminated. All remaining commands in input command file are executed except the **SOLVE** command which is ignored. Any output requested is generated and the solution process is terminated.

The contents of this file are not scanned and the action is taken solely on the basis of the name of the file.

**A.5    ACRDONE.ACR**

A file named '**acrDONE.ACR**' is generated at the end of the simulations. The purpose of this file is to act as an indicator for other processes that may be running and waiting for the simulation to complete. For example, this file can be used to automatically trigger post-processing of the solution.



## APPENDIX B: FILE FORMATS

## B.1 INPUT FILE FORMATS

**B.1.1 Input file for NODE wise Time-Dependent Function (Section 4.1.3)**

A typical file is of the form:

<Optional number of header records>

These must not begin with TIME as the first word in the record>

< A record: such as>

TIME = <value> (TIME must be the first word; command follows FREEFORM input mode)

< One or more records with numerical values which are read according to the FORTRAN Format:

READ(NUNIT,\*) (VALUE(N), N=1, NVAL)

Here:

NUNIT is a unit number which is internally assigned

VALUE(N) are the user input values

NVAL is the number of nodes in the ID=<subdomain>. If the number of values specified on the file exceeds the NVAL; then the extra values are ignored. If the specified number of values is less than NVAL, then the run is terminated with an error message.

The values for the variable at following times follow the same format. An example of such a file is given below.

**FILE TITLE: Temperature at a 12 nodes in sub domain ID=FIX\_T as a function of Time.  
Temperature is given in absolute Kelvin units.**

**TIME = 0 (First data set)**

**300 301 303 298 305 299 301 302 305 304 307 296**

**Second data set**

**TIME = 10 (Second data set)**

**300 299 300 299 300 299 300 299 300 299 300 299**

**TIME = 25 (Third data set)**

**300 301 303 298 305 299 301 302 305 304 307 296**

**TIME = 30 (Second data set)**

**300 299 300 299 300 299 300 299 300 299 300 299**

**TIME = 50 (Third data set)**

**300 299 300 299 300 299 300 299 300 299 300 299**

**END of File**

## APPENDIX C: RADM STOCHASTIC TRANSPORT METHODOLOGY

## C.1 THEORETICAL DESCRIPTION

## C.1.1 Basic Transport Equation

## C.1.1 Computational Algorithm

## C.2 COMPUTATION OF CONCENTRATION

For each computational element (or cell) the concentration of species is equal to the sum of the mass of particles in the cell divided by the volume of the cell. However in the macro scale simulations, each particle represents a large collection of “molecules” that are distributed around the mean computed position of the particle. The concentration for any element is preferably computed from the intersecting “volume” of the particle with the element. The general relation is:

$$C_M = \sum_{n=1}^{n+N} q_n [(\delta V_n \cap \delta V_M) / \delta V_M] / \delta V_M \quad , \quad (C.2.1)$$

Where

- $C_M$  is the average concentration of the species in the grid element M,
- $q_n$  is the mass of the  $n^{\text{th}}$  particle
- $\delta V_n$  is the volume of the  $n^{\text{th}}$  particle
- $\delta V_M$  is the volume of the  $M^{\text{th}}$  grid element or cell
- $N$  is the number of particles in the vicinity of the cell within a given radius of influence, and
- $\cap$  stands for the intersection of the particle volume with the cell volume.

With  $\alpha_n^M$  is the fractional contribution of the  $n^{\text{th}}$  particle to the  $M^{\text{th}}$  element, the above formula can be written as:

$$C_M = \frac{1}{\delta V_M} \sum_{n=1}^{n=N} \alpha_n^M q_n$$

There are many ways to compute the intersecting volume and the radius of influence; some of these are described below.

## C.2.1 Species Concentration from Inverse-Distance-Square Algorithm

The distance of the particle,  $d_n$ , from the element M is given as:

$$d_n^2 = \sum_{i=1}^{i=3} (x_{Mi} - x_{ni})^2$$

If the particle mass is distributed according to the inverse of the distance squared; then::

$$\alpha_n^M = \frac{1/d_n^2}{\sum_k 1/d_k^2}$$

Here the subscript k denotes the immediate neighborhood that consists of the element in which the particle is located and its immediate neighbors. For a hex grid there will be 4 neighbors in 2D and 6 in 3D; therefore the total mass of the particle will be distributed over 5 elements for 2D and 7 for 3D.

## C.2.2 Species Concentration from Location of Particle

In this case, the total mass of the particle is assumed to be located in the element of its location: If there are a total of k particles in the element M, then:

$$\alpha_n^M = 1 \quad \text{if particle in M} \quad ; \quad \alpha_n^M = 0 \quad \text{otherwise}$$

**C.2.3 Species Concentration from Particle density**

With the assumption that each particle has a volume based upon the its mass and density, the characteristic lengths for the particle ( $R_n$ ), the element ( $R_M$ ) and the distance between the element and the particle ( $d_n$ ) are computed as:

$$R_n = \frac{1}{2} \left( \frac{q_n}{\rho} \right)^{1/3} ; R_M = \frac{1}{2} \delta V_M^{1/3} ; d_n = \sqrt{\sum_{i=1}^{i=3} (x_{Mi} - x_{ni})^2}$$

Where  $X_i$  are the grid coordinates and the subscripts M and n refer to the element and the particle. The particle contribution to this cell is then computed as:

$$\alpha_n^M = \frac{R_M + R_n - d_n}{R_M + R_n} \text{ if } d_n < R_M + R_n ; \alpha_n^M = 0 \text{ otherwise}$$

**C.2.4 Species Concentration from Radius of Influence**

The expressions used are:

$$R_M = \frac{1}{2} \delta V_M^{1/3} ; d_n = \sqrt{\sum_{i=1}^{i=3} (x_{Mi} - x_{ni})^2}$$

$$\alpha_n^M = 1 \text{ if } d_n \leq R_M ; \alpha_n^M = \frac{R_M^2}{R_n^2} \text{ if } R_M < d_n < \beta R_M ; \alpha_n^M = 0 \text{ if } d_n \geq \beta R_M$$