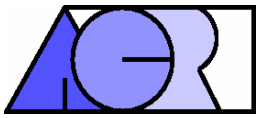
ANSWER

a software tool for navier-stokes, energy, and mass transport in turbulent reacting flows

USER'S MANUAL

VERSION 6.0

Rev: 1



1931 Stradella Road, Bel Air, CA 90077, USA • 01-310-471-3023 • FAX: 01-310-471-0797 • www.ACRiCFD.com
Cincinnati, OH 513-794-0888 • Sophia Antipolis, France • 33-492 96 75 00 • www.acri.fr • New Delhi, India

NOTIFICATION OF COPYRIGHT

September 1, 2005

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.

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.
12. 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

This ACRi Software Tool contains a number of 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 25 years, **ANSWER™** has evolved from a simple mathematical model for flow and heat transfer into a comprehensive **C**omputational **F**luid **D**ynamics software tool for analysis of a wide range of problems. It provides for coupled transport of fluid, heat and multiple chemical species in complex three-dimensional geometry. It is able to simulate the transient or steady state behavior of laminar or turbulent flow in compressible or incompressible fluids with chemical reactions, combustion, liquid sprays, droplet burning, soot formation, and radiation. It has grown from a simple computer code with few modules and fewer options to a software package with almost 1000 modules and a versatile set of options that can accommodate almost any user requirements.

ANSWER™ 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.

In the process, **ANSWER™** has evolved with the user's needs. At every stage, the flexibility and generality of the software were maintained while options were added to address user requirements. As a result, today **ANSWER™** provides a flexible format that is bound neither to a specific algorithm, nor to a particular methodology. Rather, it provides a framework that facilitates experimentation. The user can change numerical schemes, solution method, matrix inversion algorithms, or any of the physical or mathematical features.

Of all features of **ANSWER™**, two deserve special mention: generality of applications over a diverse range of problems, and ease of use provided by the conversational FREEFORM™ command language. These have enabled **ANSWER™** to emerge as a leading software in its field of application.

ANSWER™ is distinguished by the diversity of its users. It is being used by commercial, research and educational organizations in 6 countries. Among its users are Aerospatiale (France), Allison Gas Turbine, ASCI, S.A. (Spain), BAe-Sema (U.K.), C.N.I.M. (France), Department of Education (Mexico), GERPY (FRANCE), General Electric Company, James M. Montgomery, Lam Research Corporation, Marquardt Company, National Aeronautical and Space Administration, Renault (France), SNECMA (France), University of California, US Air Force, Watkins-Johnson, WS Atkins (UK), and a number of other commercial organizations. Over 50 publications and project reports on the benchmarking, verification and application of **ANSWER™** are currently available.

ANSWER™ relies on the numerical solution of complex mathematical equations. Some familiarity with the strengths and weaknesses of such mathematical and numerical algorithms is highly recommended. Every attempt has been made to provide the necessary information for satisfactory use of **ANSWER™** 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, Analytic & Computational Research, Inc. also provides training and support in the use of the software. For additional questions and inquiries, please contact:

Analytic & Computational Research, Inc.
1931 Stradella Road ● Bel Air, California, 90077● Phone: 310-471-3023, ● Facsimile: 310-471-0797
Web: www.ACRiCFD.com ● www.acri.fr ● ***Email:*** sales@ACRiCFD.com

Akshai Runchal
Bel Air, California, September 1, 2005

ACKNOWLEDGEMENTS

The **ANSWER™** software package 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

LIST OF TABLES

LIST OF FIGURES

NOTATION

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

... 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_U	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	$M L^{-3} t^{-1}$	$kg/(m^3 s)$	$lbm/(ft^3 s)$
s_R	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_a	Convert temperature to absolute units	T	K	R
T_c	Critical temperature of the fluid	T	K	R
uu	Autocorrelation of U	$L^2 t^{-2}$	m^2/s^2	ft^2/s^2
U	Fluid velocity in x-direction	$L t^{-1}$	m/s	ft/s
vv	Autocorrelation of V	$L^2 t^{-2}$	m^2/s^2	ft^2/s^2
V	Fluid velocity in y-direction	$L t^{-1}$	m/s	ft/s
V_i	Velocity in i^{th} direction	$L t^{-1}$	m/s	ft/s
ww	Autocorrelation of W	$L^2 t^{-2}$	m^2/s^2	ft^2/s^2
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
α	A coefficient	various	various	various
β	A coefficient	various	various	various
γ	Ratio of specific heats	-----	-----	-----
Γ	Diffusion coefficient	$M L^{-1} t^{-1}$	kg/(m s)	lbm/(ft s)
ε	Rate of dissipation of turbulence energy	$L^2 t^{-3}$	J/(kg s)	BTU/(lbm s)
ε	A small quantity	-----	-----	-----
θ	Angular coordinate	radian	radian	radian
κ	von Karman Constant	-----	-----	-----
λ	Decay rate constant	t^{-1}	1/s	1/s
μ	Fluid viscosity	$M L^{-1} t^{-1}$	kg/(m s)	lbm/(ft s)
ρ	Fluid density	$M L^{-3}$	kg/m^3	lbm/ft^3
σ	Prandtl or Schmidt number	-----	-----	-----
σ	Stefan-Boltzman constant	$M t^{-3} T^{-4}$	$J/(m^2 s K^4)$	$Btu/(ft^2 s R^4)$
τ	Shear stress	$M L^{-1} t^{-2}$	N/m^2	lbf/ft^2
ϕ	Density-related variable	$M L^{-3}$	kg/m^3	lbm/ft^3
ϕ_E	Equivalence ratio	-----	-----	-----
χ	Fuel fraction in fine or ambient scale of flow	-----	-----	-----

SUBSCRIPTS

SYMBOL	PERTAINING TO
e	Effective value of a property
F	Fluid property F
i	The i^{th} coordinate direction or i^{th} phase
inj	Injected fluid
j	The j^{th} coordinate direction or j^{th} phase
k	The k^{th} coordinate direction or k^{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 θ direction

SUPERSCRIPTS

SYMBOL	PERTAINING TO
F	Pertaining to property F
j	Pertaining to the j^{th} coordinate direction or j^{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^{th} fluid or the n^{th} chemical species
k	Pertaining to the k^{th} time step

(This page left intentionally blank)

CHAPTER 1
OVERVIEW AND INTRODUCTION

The ANSWER™ is a general-purpose software package for solution of the fluid flow, heat and mass transfer problems with or without chemical reactions. It is a highly flexible, modular and user-oriented. Due to the modular nature of the software, the user may readily customize the software for specific needs.

The software employs the FREEFORM™ command language and CFDStudio™, GUI pre-processor to provide a flexible, simple to use, and format-free user-interface. It interfaces with the CFDStudio™, acrPLOT™ and a number of 3rd party post-processors to display the computed results as 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.

This document describes Version 4.00, which can be operated on a broad range of micro, mini, main frame and super computers. An outline of the theory, a complete description of the input and output options, and examples of its use are given in the following chapters. The theory on which the software is based is described in more detail by publications listed in the REFERENCES section and APPENDIX A of this manual. This chapter describes the various capabilities and applications of ANSWER™.

1.1 OVERVIEW OF CAPABILITIES

The **ANSWER™** software package is a comprehensive mathematical model for simulation of fluid flow, heat and mass transport processes in laminar or turbulent, compressible or incompressible, flows at any speed. A sophisticated pre-processor allows the user to communicate with the software through the conversational, FREEFORM™ command language developed by Analytic and Computational Research, Inc. (ACRi) of Los Angeles, California. This language is described in detail in Appendix B. The pre-processor is designed to allow control of data input and output through simple, format-free, English-like commands. Only a few input commands need to be frequently used and, in the absence of specification by the user, built-in default values are assumed; this enables a newcomer to use **ANSWER™** easily without extensive training.

ANSWER™ can be used to simulate transient or steady state problems in Cartesian or cylindrical geometry. It provides a unified theoretical treatment of concepts relevant to fluid flow and transport. The physical processes incorporated in the software are shown in Figure 1.1.1. As can be seen from this figure, various levels of interaction and coupling exist between the different components of the flow system. In the **ANSWER™** software package, these components may be employed either in a coupled or uncoupled mode. Some of the important features are:

- *Cartesian or cylindrical, structured or unstructured geometry*
- *Transient or steady state simulation*
- *Option to add new variables and arbitrary number of transport equations*
- *Option to solve any or all of the governing equations*
- *Dynamic coupling between flow, heat and mass transport*
- *Powerful built-in library functions for input of physics of the problem*
- *Arbitrary user-defined functions for problem specification*
- *Laminar or Turbulent flow*
- *Compressible or incompressible fluid*
- *Subsonic, transonic or supersonic flow*
- *Multi-species chemical reaction and combustion*
- *Liquid sprays, soot formation and radiation*
- *Inherently mass-conservative numerical method*
- *Inclusion of buoyancy effects due to density variations*
- *Alternate algorithms for calculation of fluid properties*
- *Time-dependent options for physical input*
- *Format-free, conversational input with built-in default values*
- *Flexible operational and output control*
- *Choice of 'basis functions' for integration of equations*
- *Alternate methods for solution of the matrix of equations*
- *Arbitrary sources, body forces and boundary conditions*

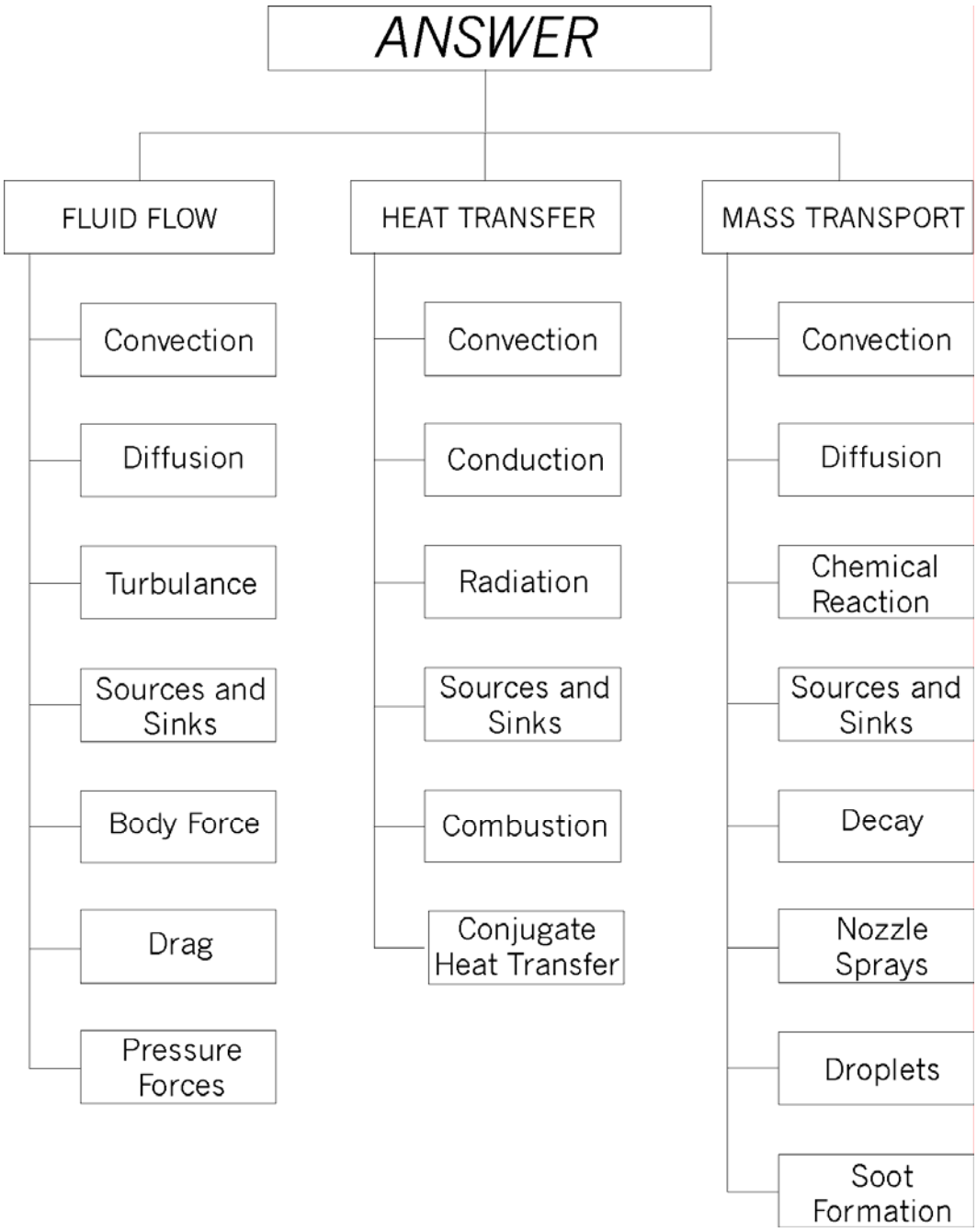


FIGURE 1.1.1: PROCESSES INCORPORATED INTO ANSWER™

1.2 VALIDATION AND PREVIOUS APPLICATIONS

ANSWER™ has benefited by more than twenty years of experience gained from computer codes for fluid dynamic applications written for the industrial, academic and research organizations. The critical elements of the software, such as the discretization schemes and the solution methods have all been independently verified during this period. The primary contribution of **ANSWER™** is in integrating these disparate elements into a software package that is general, flexible, economical, and easy to use. **ANSWER™** has been extensively verified by comparison of its results with analytic solutions, experimental and field data, and other numerical models. A large number of publications and technical reports on the applications of **ANSWER™** are currently available. A partial list of these publications is given in Appendix A. Specific applications of the software have included:

♦ AEROSPACE & DEFENCE

- *Ramjet Performance and Design Analysis*
- *Aircraft Engine Performance Analysis*
- *Afterburner Analysis*
- *Fuel Nozzle Design Optimization*
- *Turbo-jet Diffuser Design*
- *Aerofoil Analysis*
- *Rocket Exhaust Flow Analysis*
- *Missile Drag, Lift and Flight Performance*
- *Torpedo Launch Tube Analysis*

♦ AUTOMOTIVE

- *Flow Distribution Around a Car Body*
- *Aerodynamic Drag Analysis*
- *Interior Ventilation and Air Distribution*
- *Clutch Performance Analysis*

♦ ELECTRONICS

- *CVD Reactor Analysis*
- *Component Cooling and Placement Design*
- *Plasma Etch Reactor Design*
- *Voltage Potential Distribution in Components*
- *Control of Submerged Microscopic Impurities*
- *Low Pressure Film Deposition*

♦ **PROCESS INDUSTRY**

- *Pachuca Process Tank Design*
- *Flow in Pipes, Ducts and Bends*
- *Flow Past Orifices and Obstacles*
- *Heat Transfer Augmentation with Swirling Flows*
- *Boiler Design & Performance Analysis*
- *Non-Newtonian Flow in Reactor Vessels*

♦ **GENERAL APPLICATIONS**

- *Submerged Body Shape & Signal Optimization*
- *Flow in Centrifugal Pumps*
- *Ventilation of Toxic Fumes*
- *HVAC Design Optimization*
- *Lubrication of Bearings*
- *Water Flume and Tunnel Design*
- *Flow in Pipes, Channels and Flumes*
- *Flow in Pipes with Steps and Obstructions*
- *Flow in Diffusers*
- *Driven Cavity Flow*
- *Recirculating Flows in Solid Body Rotation*
- *Swirling Flow in Pipes and Annuli*
- *Boundary Layer and Free Jet Flow*
- *Mixing of Jets*

1.3 STRUCTURE OF ANSWER™

ANSWER™ is designed to achieve four major objectives:

1. ***User-friendliness***
2. ***Generality***
3. ***Flexibility***
4. ***Economy of computation.***

To achieve the first objective, the software employs a friendly user interface that allows specification of input and output requirements through the conversational, English-like, **FREEFORM™** command language developed by **ACRi**. The output from the software may be displayed in convenient tabular or graphical form.

To achieve the second of these objectives, a simple and convenient method is employed to define the geometry and the physics of the system. The nature of the geometry, itself, can be specified to be Cartesian, cylindrical or body-fitted. The grid may be structured or unstructured. The geometrical arrangement for a general problem is considered to be composed of six types of 'elements'; these allow easy placements of specific features such as inlets, outlets and walls The values of dependent variables, such as velocity, pressure, temperature and species concentration, may be specified for any grid element. Arbitrary sources or sinks may be located in the interior of the domain of interest. The physics of the problem is either inferred from the specified initial and boundary conditions or is easily selected by simple user commands.

The last two objectives are achieved by employing a highly modular structure for the software which, for the current version, consists of more than 300 individual modules. In general, each module is dedicated to a single function and can be replaced by an alternate module without significant change to the remaining modules or to the structure of **ANSWER™**. The code provides options for alternate discretization schemes, solution methods, matrix solvers, and fluid property calculation algorithms. In addition, to accommodate special user requirements, options are provided for user-defined input modules for physical and material properties, boundary conditions, and sources. Any option not required for a given problem can easily be bypassed by simple index specifications, this allows efficient and economical solution of a wide range of problems of varying complexity.

The generality, economy and efficiency of **ANSWER™** is demonstrated by the fact that it *is fully operational on a large number of computer systems ranging from IBM-PC microcomputers to CRAY supercomputers*.

1.4 SCOPE AND LIMITATIONS

1.4.1 System of Equations

ANSWER™ numerically solves a variable set of general transport equations for mass conservation, momentum, energy, turbulence, chemical species, droplet transport, thermal radiation and soot formation. These equations are supplemented by constitutive equations, equations of state, and initial and boundary conditions, and are coupled through convection, pressure and viscous forces, fluid density and buoyancy, temperature, and chemical kinetic effects. These equations may be solved individually or simultaneously in a coupled or uncoupled manner, depending on the nature of a specific problem and options selected by the user.

The current version of **ANSWER™** provides for a arbitrary number of chemical species with multi-step reaction kinetics. The droplet transport is lagrangian with liquid-vapor phase change.

1.4.2 Spatial Dimensionality

The code is designed primarily to solve two-dimensional (2D) or three-dimensional (3D) problems. It can accept user input in either a 2D or 3D mode. Full details of these input modes are given in Chapters 6 and 7. One-dimensional problems can be solved by specifying a grid size of 1 element (3 nodes) in the direction(s) that is (are) to be omitted.

1.4.3 Problem Geometry

The problem geometry can be defined in terms of either cartesian (x, y, z) or cylindrical (x,r,θ) coordinates. In the 2D mode, only the (x, y) or (x, r) planes are allowed. One- or two-dimensional problems, of course, can be simulated in the 3D mode in any combination of coordinates by assigning 3 nodes in each direction to be omitted.

Orientation of axes with respect to the gravitational acceleration vector is arbitrary. The gravitational vector may be aligned with any coordinate direction or it may be at any arbitrary angle to the coordinate direction.

1.4.4 Temporal Dependence

Either transient or steady state problems can be solved. Except for the geometry and spatial grid, all problem parameters can change with time. The source terms, the boundary conditions, the values of field variables, and the fluid properties can all change either continuously or abruptly with time. Such quantities can be specified as analytic or tabular functions or through user-specified options.

1.4.5 Spatial Dependence

The values of most parameters are allowed to vary over the spatial grid. The fluid density, viscosity, specific heat, sources etc. can all change from one location to another.

1.4.6 Methods for Solving Governing Equations

The method of Nodal Point Integration (NPI) is employed for integration of the governing differential equations by temporal and spatial discretization over each control volume (element) of the physical domain. It leads to solutions that automatically conserve fluid, heat, and mass locally within every grid element, as well as for the entire flow domain. The storage terms are approximated by a modified Newton-Raphson method. The dependent variable or its change from the current state approximates the flux terms. The

elements used to define the problem geometry can vary in size, but their shape is restricted to that of a quadrilateral, hexahedral or segment of a cylinder. Three distinct discretization schemes are available: the hybrid, CONDIF and QUICK schemes.

The resulting matrix of algebraic equations can be solved by one or more of several matrix inversion algorithms. The available options include the Point Successive Over-Relaxation, the Alternating Direction Implicit (ADI), the Conjugate Gradient, Cholesky Decomposition and Gaussian Elimination. In addition, the software provides the flexibility to use any other matrix inversion technique through coupling with an external matrix-inversion algorithm.

1.4.7 Boundary Conditions

Varied types of boundary conditions can be specified. Dirichlet (specified values of pressure, temperature, or concentration), Neumann (specified fluxes of fluid, heat, or mass), or mixed (combination of specified values and fluxes) boundary conditions can be stipulated. Different types of boundary conditions can be designated at various segments of a boundary. Combined with the time-dependence feature discussed in Section 1.4.4, this feature can be used to solve a large variety of problems that have space- and time-dependent boundary conditions.

1.4.8 Sources and Sinks

Several options are provided in **ANSWER™** for incorporating sources and sinks of fluid, heat, and mass. Fluid injection or withdrawal, and sources (or sinks) of heat or chemical species may occur anywhere in the interior of the domain of interest. Spatially variable sources and sinks can be specified. The magnitudes of the sources or sinks either can be constant or vary with time. For chemical species, the sources can be limited by their inventory, solubility, or both.

1.4.9 Natural Convection and Body Forces

Current versions of **ANSWER™** do not have a provision for generalized body forces. However, buoyancy and natural convection due to the effects of variations in fluid density is included as an integral part of the software. This option can be modified to incorporate other body forces such as electromagnetic forces.

1.4.10 Non-Newtonian Flow

A non-linear relation between fluid stress and strain tensors characterizes a non-Newtonian flow. This effect is included through viscous forces. Current versions of **ANSWER™** provide for alternate options to compute fluid viscosity. For Newtonian flows, viscosity is independent of fluid velocity. A non-Newtonian flow may be treated in a linearized manner by defining a viscosity that is a function of velocity gradients. This option has been used to simulate non-Newtonian flow. Turbulent flow, which is a built-in option, is a special case of a non-Newtonian flow.

1.4.11 Chemical Reaction and

ANSWER™ incorporates a multi-step chemical kinetic algorithm for chemical reaction and combustion. A default 4-step, 8 species, reaction scheme optimized for hydrocarbon reactions is also incorporated.

1.4.12 Soot and Particulate Matter Calculations

ANSWER™ incorporates an option to compute soot nuclei and soot particle density as a result of incomplete hydrocarbon combustion. The computation of soot is based upon the solution of three transport equations: one for soot nuclei and two for soot particle sizes. This option can also be modified to solve for other particulate matter.

1.4.13 Radiation Heat Transfer

The thermal radiation in **ANSWER™** is computed on the basis of a three-equation, six-flux model. In this model, the net radiation flux in any coordinate direction, at any location in space, is described by the sum of two fluxes: one aligned with the positive and the other in the negative direction of the coordinate.

1.4.14 Two-Phase Flow and Droplet Transport

Current version of **ANSWER™** does not have a full two-phase flow capability. However, limited two-phase capability is available for special flow situations. Soot equations may be modified to compute transport of particulate material, such as slurries. For gas-liquid mixtures, an option is available to account for injection of a liquid spray in a gaseous stream through one or more nozzles.

1.4.15 Conjugate Heat Transfer

In most fluid dynamic applications, the fluid is the focus of concern. However, some applications involve heat transfer both in the fluid and in the surrounding (or embedded) solid components. Conjugate heat transfer is accommodated by a simple user command.

1.4.16 Drag from Sub-Grid Scale Objects

At times, there are solid objects that are too small to be economically resolved by the computational grid. Certain other objects may be of a porous nature which present additional and distributed drag to the flow. **ANSWER™** provides an option to account for the momentum drag effect form such embedded objects.

1.4.17 Moving Boundaries

Any of the external boundaries of the domain of interest may be specified to be moving at an arbitrary velocity. By default, it is assumed that all boundaries of the fluid are stationary. However, simple user specification, through generalized initial and boundary commands may be used to specify moving boundaries.

1.4.18 Moving Embedded Objects

A special feature of **ANSWER™** is its ability to accommodate a object moving at an arbitrary velocity through the field of interest. The object may be surrounded on one or more sides by fluid (such as a torpedo or missile moving inside its launch tube) or it may act as a moving barrier (such as a piston moving in a cylinder). The rate of movement of such objects may be varied with time.

1.4.19 Operational and Output Control

Through design of the input, the user can exert extensive control over the operation of the software. For example, the execution of the code can be stopped to change boundary conditions at any convenient point and restarted later from the point at which it was stopped. The user also has considerable control over the extent and nature of output. Output can be obtained in a tabular form or written in a file for post-processing in a graphic form. The variables to be tabulated, the size of the tables, and the times at which they are to be obtained can all be controlled by input commands.

(This page left blank intentionally)

CHAPTER 2

GOVERNING EQUATIONS

ANSWER™ solves a set of coupled transport equations for fluid, heat and mass transport processes in laminar or turbulent, compressible or incompressible, transient or steady, flows at any speed. It is able to accommodate chemical reactions, combustion, thermal radiation, soot formation, droplet transport, embedded moving boundaries, complex fluid property relations, and a number of other fluid flow phenomena. The equations to be solved can be easily varied to accommodate specific user requirements. The equations may be solved in their two- or three- dimensional, transient or steady state form.

The governing equations are based on the conservation principles of continuum mechanics, expressed in the form of Navier-Stokes, energy and transport equations. A brief summary of the governing equations is provided in this chapter. The principal processes incorporated in these equations were summarized earlier in Figure 1.1.1.

2.1 THE GENERAL TRANSPORT EQUATION

The general equation for the transport of a property F of the fluid in the Cartesian tensor notation is written as:

$$\frac{\partial}{\partial t}(\alpha F) + \frac{\partial}{\partial x_i}(\beta V_i F) = \frac{\partial}{\partial x_i}(\Gamma_e^F \frac{\partial F}{\partial x_i}) + m_{inj} F_{inj} + S_F \quad s_F \alpha F \ , \tag{2.1.1}$$

where

- t is the time coordinate,
- α is a coefficient for the accumulative term,
- ρ is the mass density of the fluid,
- F is the transported property,
- β is a coefficient for the convective term,
- V_i is the velocity component of the fluid in the i^{th} direction,
- x_i is the coordinate in the i^{th} direction,
- Γ_e^F is the diffusivity tensor,
- m_{inj} is the mass injected per unit volume, per unit time,
- F_{inj} is the amount of property in the injected mass,
- S_F is the source term of F, and
- s_F is the rate constant for reaction or removal of property F.

In the more familiar Cartesian and cylindrical coordinate notation, this equation is written as:

$$\frac{\partial}{\partial t}(\alpha F)+\frac{\partial}{\partial x}(\beta UF)+\frac{1}{r}\frac{\partial}{\partial y}(r\beta VF)+\frac{1}{r}\frac{\partial}{\partial z}(r\beta WF)=\frac{\partial}{\partial x}(\Gamma_e^F\frac{\partial F}{\partial x_i})+\frac{1}{r}\frac{\partial}{\partial y}(r\Gamma_e^F\frac{\partial F}{\partial y})+\frac{1}{r}\frac{\partial}{\partial z}(\Gamma_e^F\frac{1}{r}\frac{\partial F}{\partial z})+m_{inj}F_{inj}+S_F-s_F\alpha F \tag{2.1.2}$$

For the Cartesian coordinate system, x, y and z are the Cartesian coordinates and r is set to unity. For cylindrical polar coordinate system, x, y and z represent, respectively, the axial coordinate (x), the radius (r), and the angular coordinate (θ).

2.2 THE BASIC SET OF EQUATIONS

The basic set of equations consists of the equation of continuity, the three components of the momentum equations, the energy conservation equation, the equations for turbulence energy and its rate of dissipation, and one of more general transport variables which define a particular application. These basic variables are defined in Table 2.2.1.

TABLE 2.2.1: THE BASIC VARIABLES OF ANSWER™

No.	Variable	Description
1	ρ	Mass density of fluid
2	U	Velocity component in the x (or axial) direction
3	V	Velocity component in the y (or radial, r) direction
4	W	Velocity component in the z (or circumferential, θ) direction
5	h_s	Stagnation enthalpy
6	k	Kinetic energy of turbulence
7	ε	Rate of dissipation of turbulence energy
8	Φ	One or more of the other transport variables required for a specific problem. These variables may include conserved scalars, chemically active species or Reynolds stresses if a differential Reynolds stress model for turbulence is used. **These variables may be defined by the user through the ALLOCATE command or may be seleced from the built-in dependent variables that are listed later in Table 6.7.1.**

Two of the dependent variables that appear in this Table are derived from the primitive variables for computational convenience. These are the kinetic energy of turbulence, k, and the stagnation enthalpy, h_s. These are defined as follows:

$$k = \frac{1}{2}(uu + vv + ww) , \tag{2.2.1}$$

$$h_s = \sum_j m_j h_j + \frac{1}{2}(U^2 + V^2 + W^2) + k , \tag{2.2.2}$$

$$h_j = h_j^o + \int_T C_{pj} dT \tag{2.2.3}$$

where

$$\sum_j m_j = 1 , \tag{2.2.4}$$

uu is the autocorrelation of the turbulent fluctuations of U,

vv is the autocorrelation of the turbulent fluctuations of V,

ww is the autocorrelation of the turbulent fluctuations of W,

m_j is the mass fraction of j^{th} chemical species,

h_j is the enthalpy of the j^{th} chemical species,

h is the heat of formation of the j^{th} species,

C_{pj} is the specific heat of the j^{th} species at constant pressure, and

T is the fluid temperature.

All these basic variables are governed by the general transport equation, Equation 2.1.1. The coefficients and source terms of the transport equation for these variables are summarized in Table 2.2.2. The S_i and P_k, which appear in this Table are are given by:

$$S_i = -\frac{\partial p}{\partial x} + \rho g_i + \frac{\partial V_j}{\partial x_i} - \frac{4}{3}\frac{\partial \mu_e}{\partial x_i} - \frac{1}{3}\mu_e \frac{\partial D}{\partial x_i} \tag{2.2.5}$$

$$D = \frac{\partial V_i}{\partial x_i}, \tag{2.2.6}$$

$$P_k = \mu_e\left(\frac{\partial V_j}{\partial x_i} + \frac{\partial V_i}{\partial x_j}\right)\frac{\partial V_j}{\partial x_i} - \frac{2}{3}(\rho k + \mu_e D)D - \frac{\mu_e}{\rho^2}\frac{\partial \rho}{\partial x_i}\frac{\partial P}{\partial x_i} \tag{2.2.7}$$

TABLE 2.2.2: COEFFICIENTS AND DEFAULT SOURCE FOR BASIC VARIABLES

F	α	β	Γ	S_F	s_F
ρ	1	1	0	0.	0.
U	ρ	ρ	μ_e	S_1	0.
V	ρ	ρ	μ_e	$S_2 - (\partial r/\partial y)\{\mu_e(V + 2\,\partial W/\partial z)/r^2 - \rho W^2/r\}$	0.
W	ρ	ρ	μ_e	$S_3 - (\partial r/\partial y)\{\mu_e(W - 2\,\partial V/\partial z)/r^2 + \rho VW/r\}$	0.
h_s	ρ	ρ	Γ	$P_k + 2(R_x+R_y+R_z - 3E)$	0.
k	ρ	ρ	μ_e	P_k	ε/k
ε	ρ	ρ	Γ	$C_{\varepsilon 1} P_k \varepsilon /k$	$C_{\varepsilon 2} \varepsilon /k$
ϕ	ρ	ρ	Γ	Problem specific	Problem specific

2.3 DERIVATION OF A MODIFIED CONTINUITY EQUATION

The equation of continuity for the conservation of a compressible fluid mass in Cartesian tensor notation is written as:

$$\frac{\partial \rho}{\partial t} + \frac{\partial V_i}{\partial x_i} = m_{inj} \ . \tag{2.3.1}$$

In **ANSWER™**, the equation of continuity is transformed into an equation for computation of density (compressible flow) or pressure (incompressible flow). At any stage of the solution procedure, previously computed (or initial) values for density, pressure and velocity components are available. These values are then used to compute a new approximation to the velocity field, V, from the momentum equations while the pressure and density are held at their previous (initial) state. Thus, in essence, the initial pressure and density values are used as approximate values to compute the new velocity field. In reality, of course, the coupled system of equations implies that all these variables must be simultaneously adjusted. The correct fluid velocity, density and pressure that satisfy the governing equations are given by:

$$V_i = V_i^* + V'_i \ , \tag{2.3.2}$$

$$\rho = \rho^* + \rho' \ , \tag{2.3.3}$$

$$p = p^* + p' \ , \tag{2.3.4}$$

where the quantities with a prime are corrections to the approximate values. Substitution of these relations in Equation 2.3.1, after neglecting second order terms, leads to:

$$\frac{\partial \rho'}{\partial t} + \frac{\partial}{\partial x_i}(\rho' V_i^*) + \frac{\partial}{\partial x_i}(\rho^* V'_i) = m_v \ , \tag{2.3.5}$$

$$m_v = m_{inj} - \frac{\partial}{\partial x_i}(\rho^* V_i^*) \ . \tag{2.3.6}$$

Following Chorin (1967), it can be shown that the approximate velocity field differs from the final by the gradient of a scalar field, ϕ; so that:

$$V'_i = - c_v \frac{\partial \varphi}{\partial x_i} \ . \tag{2.3.7}$$

We now assume that this scalar field is related to the density correction by:

$$\rho' = c_\rho \varphi \ . \tag{2.3.8}$$

Equation 2.3.5 then transforms to:

$$\frac{\partial}{\partial t}(c_\rho\ \varphi)+\frac{\partial}{\partial x_i}(c_\rho\ V_i^*\ \varphi)=\frac{\partial}{\partial x_i}(\rho^*\ c_v\ \frac{\partial \varphi}{\partial x_i})+m_v\ . \tag{2.3.9}$$

From the equation of state for a fluid, the pressure change due to a change in density can be computed as:

$$p' = \rho'\ \frac{\partial p}{\partial \rho}\ , \tag{2.3.10}$$

which for the perfect gas law (see Equation 3.1.4) becomes:

$$p' = \rho'\ C\ R\ T\ , \tag{2.3.11}$$

where C is equal to unity for an isothermal process, and is equal to, γ, the ratio of the specific heat at constant pressure to that at constant temperature, for an isentropic process.

This approach of working directly with the density variable is termed DEFCON for **D**ensity **E**quation **F**ormulation of **CON**tinuity equation. For incompressible flow this approach can be made identical to the SIMPLE algorithm (Patankar & Spalding, 1972) by a suitable selection of the c_V (derived from the momentum equation).

The DEFCON offers considerable advantages over the SIMPLE approach for compressible flows. First, the pressure is now determined from the equation of state rather than from the constraint that it must satisfy the continuity equation; that role is directly taken over by the density variable which is naturally suited for this function. Second, and more important, in the SIMPLE algorithm the "convective" term (the second term on the left hand side of Equation 2.3.9) contains the $\partial\rho/\partial p$ term which must be evaluated at cell faces. Since density and pressure are only defined at grid nodes, this term must be obtained either from averages or from upwind assumptions. The former leads to non-physical effects for supersonic flows (changes in such flows can not travel upstream) while the latter causes significant false numerical diffusion. Karki (1986) has discussed this in detail in his thesis. In the DEFCON approach, the convective terms contain only the constant c_ϑ and no inter-nodal values are required.

One potential disadvantage of the DEFCON approach is that for incompressible flows, $\partial p/\partial\rho$ is infinite (in fact, a large number on the order of the square of sonic speed). This can lead to numerical accuracy problems due to round off (see, e.g. Karki, 1986). This is, however, easily avoided by setting this quantity to an arbitrary number since pressure-related changes in density are unimportant for incompressible flow. This is akin to using the artificial compressibility method. In the DEFCON approach, the density correction is omitted for incompressible flows and only pressure is corrected. Strictly speaking, in such fluids, it can be shown that the constant c_ρ is on the order of the inverse of the square of the sonic speed. In practice, a value equal to the inverse of the sonic speed has proved satisfactory for a very wide variety of problems.

(This page left intentionally blank.)

CHAPTER 3

AUXILIARY EQUATIONS

Several auxiliary relations are required to solve the governing transport equations. Solution of these equations requires the specification of fluid properties, constitutive relations, source terms, and initial and boundary conditions. The mathematical framework of ANSWER™ is sufficiently general to accommodate virtually any mathematical relation for these quantities and many options are available. Some of the commonly employed expressions that are provided as built-in options are described in this chapter.

3.1 FLUID DENSITY

Changes in the fluid density significantly effect the flow and pressure equations through the density term in the continuity and momentum equations. Both analytic and arbitrary user-defined functions for calculation of fluid density are included. The currently available analytic functions are:

$$\rho = \rho^* \left[\frac{T_c - T}{T_c - T^*} \right]^{a1} \tag{3.1.1}$$

$$\rho = \rho^* [1 + a_1 (T^* - T) + a_2 (T^* - T)^2 + a_3 (T^* - T)^3] \tag{3.1.2}$$

$$\rho = \rho^* [1 + a_1 (T^* - T) + a_2 (C^* - C)] \tag{3.1.3}$$

$$\rho = \frac{p + p^*}{R_u (T + T_a) \sum_j \frac{m_j}{M_j}} \tag{3.1.4}$$

where:

ρ^* is the reference density for the fluid,
T_c is the critical temperature of the fluid,
T^* is the reference temperature,
a_1, a2, a_3 are empirical constants,
C^* is the reference mass concentration,
p^* is the reference datum for thermodynamic pressure,
R_u is the universal gas constant,
T_a is a constant to convert temperature to absolute units,
m_j is mass fraction of the j^{th} chemical species in the gas, and
M_j is the molecular weight of the j^{th} species.

The first three equations are used primarily for liquids. The last is the equation of state for a perfect gas. For water, Equation 3.1.1 provides a good fit to the experimental data. With ρ^* = 996.59 kg/m^3, T_c = 647.3 K, T^* = 300 K and a_1 = 0.20, the water density computed from Equation 3.1.1 differs from the reference values by less than 1 percent for 4 $^oC < T < 200\ ^oC$, and by less than 2.5 percent for 0 $^oC < T < 350\ ^oC$ (Perry and Chilton, 1973, pp. 3-230).

3.2 FLUID VISCOSITY

Three different options for computing fluid viscosity are included. Other options, such as those for non-newtonian fluids, may be easily added, if required. Of the built-in options, the simplest is the specification of a uniform and constant viscosity. The second option is that of a user specified set of arbitrary values. The third option is that for computation of effective viscosity for a turbulent fluid. In this case, the effective viscosity, μ_e, is assumed to be given by:

$$\mu_e = \mu + \mu_t \ , \tag{3.2.1}$$

where μ and μ_t are the coefficients of molecular and turbulent viscosity, respectively.

For the k-ε model, the turbulent viscosity is given by:

$$\mu_t = C\mu \, \rho \, k^2 / \varepsilon \ , \tag{3.2.2}$$

where $C\mu$ is an empirical constant.

3.3 DIFFUSION COEFFICIENTS

The quantity, Γ_e^F of Equation 2.1.1 represents a diffusion coefficient which is given by:

$$\Gamma_e^F = \mu / \sigma^F + \mu_t / \sigma, \tag{3.3.1}$$

where

σ^F is the Prandtl or Schmidt number for the property F,
σ is the turbulent Prandtl or Schmidt number for the property F.

Equation 3.3.1 may be rearranged as:

$$\Gamma_e^F = \mu_e / \sigma_e^F , \tag{3.3.2}$$

where σ_e^F , the effective Prandtl or Schmidt number for the fluid, is given by:

$$\sigma_e^F = \mu_e \sigma^F \sigma_t^F / (\sigma^F \mu_t + \sigma_t^F \mu) . \tag{3.3.3}$$

For most flows, turbulent viscosity is much larger than the molecular viscosity, and both σ^F and σ_t^F are on the order of unity; in this case:

$$\sigma_e^F \approx \sigma_t^F . \tag{3.3.4}$$

3.4 SPECIFIC HEAT

Three built-in options for calculation of the mean specific heat of fluid are provided. The first and second of these are a constant value, and an arbitrary set of user specified values which may vary over the field of interest. The third formulation is that of Gordon and McBride (1971. In this case, for each chemical species, j:

$$C_{p_j} = a_{o_j} + a_{1_j} T + a_{2_j} T^2 + a_{3_j} T^3 + a_{4_j} T^4 \ , \tag{3.4.1}$$

where a's are empirical constants. The values for the 8 chemical species described in Section 3.6 are provided as default values. These are valid over a temperature range from 300K to 2000K under atmospheric conditions. These may be replaced by other suitable values, if required.

The mean specific heat for the fluid is then computed as a mass-weighted average for all the species; that is:

$$C_p = \sum_j \int_T \frac{C_{p_j} \, m_j \, d\,T}{T - T_o} \ , \tag{3.4.2}$$

where T_o is the datum temperature from which the component specific heats are defined. Most often the datum is absolute zero.

3.5 CALCULATION OF TEMPERATURE FROM ENTHALPY

The energy equation is solved in terms of the fluid stagnation enthalpy. The fluid temperature must therefore be computed from enthalpy from its definition (Equation 2.1.2). The equation for temperature is written is:

$$T = T_o + \frac{1}{C_p}\left[h_s - \sum_j m_j \, h_j^o - \frac{1}{2} \left(U^2 + V^2 + W^2 \right) - k \right], \qquad (3.5.1)$$

where C_p is a mean specific heat as defined by Equation 3.4.2.

3.6 THE CHEMICAL REACTION

3.6.1 Reaction Mechanism

The ANSWER™ provides for a chemical reaction set consisting of up to 4 reactions which may involve up to 8 chemical species. Each of the species obeys a general transport equation of the form given by Equation 2.1.1.

Though the reaction set can be modified to adapt to any species, the built-in system is optimized for hydrocarbon reactions. It is assumed that the basic mechanism is that of the oxidation of a hydrocarbon fuel, with the composition, C_nH_m, by an oxidant consisting of oxygen, nitrogen and other species. The 8 species taking part in the reaction are C_nH_m, C_nH_{m-2}, CO, H_2, O_2, CO_2, H_2O and N_2. The reaction mechanism consists of 4 steps as follows:

$$C_nH_m \quad \text{----> } C_nH_{m-2} + H_2 \ , \tag{3.6.1}$$

$$C_nH_{m-2} + \tfrac{1}{2}\, n\, O_2 \quad \text{----> } n\, CO + \tfrac{1}{2}\,(m-2)\, H_2 \ , \tag{3.6.2}$$

$$CO + \tfrac{1}{2}\, O_2 \quad \text{----> } CO_2 \ , \tag{3.6.3}$$

$$H_2 + \tfrac{1}{2}\, O_2 \quad \text{----> } H_2O \ . \tag{3.6.4}$$

The rate expression for this scheme, developed primarily from propane oxidation (Hautman et al., 1981 is:

$$S_{mj,k} = C_A\, S_o \exp(-C_E/T)\, \rho^{(a+b+c)}\, m_j^{\,a}\, m_k^{\,b}\, m_l^{\,c} \tag{3.6.5}$$

where $S_{mj,k}$ is the kinetic rate of reaction (S_F term of Equation 2.1.1) of j^{th} species, m_j, m_k and m_l are mass fractions of the participating species, and 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 for which it is given by:

$$S_o = \text{minimum } \{7.93 \exp(-2.48\, \Phi_E),\ 1.\} \ , \tag{3.6.6}$$

where Φ_E is the initial equivalence ratio. Default parameters that define this reaction set are summarized in Table 3.6.1

TABLE 3.6.1: DEFAULT CONSTANTS FOR THE REACTION RATE TERM

m_j	m_k	m_l	$C_A x 10^{-18}$	$C_E x 10^{-4}$	a	b	c
m_{CnHm}	m_{O2}	m_{CnHm-2}	20893	2.48	0.50	1.07	0.40
m_{CnHm-2}	m_{O2}	m_{CnHm}	50.117	2.50	0.90	1.18	-0.37
m_{CO}	m_{O2}	m_{H2O}	39.811	2.00	1.00	0.25	0.50
m_{H2}	m_{O2}	m_{CnHm-2}	3.3113	2.05	0.85	1.42	-0.56

For turbulent flows the reaction process may be significantly affected by intermittency. For such flows the reaction rate is calculated from a variation of the 'eddy-break-up' (EBU) model proposed by Spalding (1971 as:

$$S_{mj,t} = C_{EBU}\, \rho\, m^{ms}\, \varepsilon / k\,, \qquad (3.6.7)$$

where C_{EBU} is an empirical constant and m^{ms} is a measure of the r.m.s. value of the local fluctuations in the j^{th} species. As a first approximation, the rms measure is obtained as:

$$m^{ms} = \text{minimum}\,\{m_j,\, \alpha_j\, m_{O2}\}\,, \qquad (3.6.8)$$

where $\alpha_j\, m_{O2}$ is the maximum value of the j^{th} species that is able to react with the local oxygen. The actual rate of chemical reaction is assumed to be limited by the smaller of the two values given by Equation 3.6.6 and 3.6.7; that is:

$$S_{mj} = \text{minimum}\,\{S_{mj,k},\, S_{mj,t}\}\,. \qquad (3.6.9)$$

A better estimate of m^{rms} can be obtained by solving one or more partial differential equations which are identical to the general transport equation (Equation 2.1.1). This option, though potentially desirable from a theoretical point of view, adds to the cost of computations. Therefore, in view of the general uncertainties inherent in the interaction between turbulence and chemical reaction, this option is not selected. This is acceptable for most turbulent flows. It should be noted that interaction of turbulence and chemistry is a field of very active research. A number of alternative models have been proposed and tested. Among these are the PDF (Pope and Correa, 1986 and flamelet (Bray et al., 1985) models. Due to the modular nature of **ANSWER™**, the chemistry modules can be easily modified or replaced to incorporate an alternative reaction model. One example of such efforts is the work by Candel et al. (1990).

3.6.2 Mass Fractions of Chemical Species

Of the 8 species involved in the reaction process, 4 can be determined from the transport equations for C_nH_m, C_nH_{m-2}, CO and H_2. Three additional transport equations are obtained for elemental carbon (C), oxygen (O) and hydrogen (H). Since these quantities are all individually conserved, the source terms (S_F and s_F) for these equations are zero. The 8th equation, for nitrogen (N_2), is obtained from the constraint that the sum of all mass species must equal unity. If m_C, m_H and m_O are the respective values of the elemental carbon, hydrogen and oxygen, then, the mass fractions of the chemical species CO_2, H_2O, O_2 and N_2 can be obtained from:

$$m_{H2O} = \alpha_1 (m_H - m_{CnHm} - \alpha_2 m_{CnHm-2} - \alpha_3 m_{H2}) , \quad (3.6.10)$$

$$m_{CO2} = \alpha_4 (m_C - m_{CnHm} - \alpha_5 m_{CnHm-2} - \alpha_6 m_{CO}) , \quad (3.6.11)$$

$$m_{O2} = m_O - \alpha_7 m_{CO} - \alpha_8 m_{H2O} - \alpha_9 m_{CO2} , \quad (3.6.12)$$

$$m_{N2} = 1 - m_{CnHm} - m_{CnHm-2} - m_{CO} - m_{H2} - m_{H2O} - m_{CO2} - m_{O2} , \quad (3.6.13)$$

where:

$$\alpha_1 = ½ m M_{H2O} / M_{CnHm} , \quad (3.6.14)$$

$$\alpha_2 = (1-2/m) M_{CnHm} / M_{CnHm-2} , \quad (3.6.15)$$

$$\alpha_3 = (2/m) M_{CnHm} / M_{H2} , \quad (3.6.16)$$

$$\alpha_4 = n M_{CO2} / M_{CnHm} , \quad (3.6.17)$$

$$\alpha_5 = M_{CnHm} / M_{CnHm-2} , \quad (3.6.18)$$

$$\alpha_6 = (1/n) M_{CnHm} / M_{CO} , \quad (3.6.19)$$

$$\alpha_7 = ½ M_{O2} / M_{CO} , \quad (3.6.20)$$

$$\alpha_8 = ½ M_{O2} / M_{H2O} , \quad (3.6.21)$$

$$\alpha_9 = M_{O2} / M_{CO2} . \quad (3.6.22)$$

where M's are the molecular weights of the respective species.

3.7 SOOT EMISSIONS

The particulate emission of primary concern in the combustion of hydrocarbon fuels is soot, which is evident in the form of exhaust smoke. The model adopted for computing soot emissions is derived from the COM3D (Srivatsa, 1983) computer code developed for NASA. The COM3D model, in turn, derives heavily from the work of Magnussen et al. (1978), and Nagle and Strickland-Constable (1962).

The processes governing the formation and subsequent oxidation of soot are of a particularly complex nature; and, as such, quantitative models of soot production are still being developed. Soot is not an equilibrium product of combustion; and therefore, its formation is influenced as much by the physical processes of atomization, evaporation, and fuel/air mixing as by reaction kinetics. Soot is generally produced anywhere within the combustor where the temperature is high and fuel/air mixing is inadequate, resulting in high-temperature, oxygen-deficient zones.

Detailed discussions of the many mechanisms proposed to explain the chemical and physical processes governing soot formation are available in reviews by Haynes and Wagner (1981) Palmer and Culliss (1965), Gaydon and Wolfhard (1979), Homann (1967) and Bittner and Howard (1978). Based on the information available, the process of soot formation can be considered to occur in three distinct stages:

- *Soot-particle nucleation*
- *Agglomeration and surface growth*
- *Coagulation.*

The computation of soot emissions involves the solution of three additional transport equations; one equation for the particle density of soot nuclei and two for the concentrations of soot particles for two different particle diameters. The smaller of the soot particle diameters is assumed to result from nucleation while the larger results from fuel droplet pyrolysis and char formation. By default these particle are assumed to be 0.025 and 1 micron diameter and their relative rates of formation are 90 to 10 percent. However, these default values can be changed by the user.

The soot formation is assumed to occur only if the temperature and the local carbon-to-oxygen ratio are above certain threshold values. These threshold values are called incipient temperature and carbon-to-oxygen ratio and are inputs to the calculation procedure.

For soot nuclei, the general formation rate expression is:

$$S_{sn} = C_{n1}N_{sn} + \sum_j \{C_{n2}\, m_{CnHm} \exp(C_{Es}/T_j) - C_{n3}N_{sn} \sum_i N_{si}\}\, \chi_j \rho / \rho_j \qquad (3.7.1)$$

where:

S_{sn} is the formation rate of soot nuclei (particles/m^3s),
N_{sn} is the density of soot nuclei (particles/m^3),
N_{si} is the concentration of i^{th} soot particles size (kg/m^3),
T_j is the temperature of the fluid in the j^{th} state,
χ_j is the relative mass fraction of fuel in the j^{th} state, and
ρ_j is the density of the fluid in the j^{th} state.

The C's in this equation are empirical constants and the subscript, j, denotes the j^{th} state of the fluid. The states of the fluid are used here to account for the influence of turbulence on soot formation. Based on the

work of Magnussen et al. (1978), it is assumed that a turbulent fluid exists in two states: a fine structure with the characteristic Kolmogorov length scale where the energy dissipation and reaction take place and the surrounding ambient fluid with larger scales. For laminar flow, of course, no such distinction exists and all fluid is assumed to be in one state: that of the ambient fluid.

The summation of Equation 3.7.1 is thus to be carried out over both the fine structure and the ambient fluid states. Following Magnussen et al. (1978), it is assumed that the relative mass fraction of the reacted fuel in the fine structure, χ_f, and that in the ambient fluid, χ_a, are given by:

$$\chi_f = 9.7\ Re^{3/4} / \{1 + (1+i)\ m_{CnHm}/m_{pr}\}, \tag{3.7.2}$$

$$\chi_a = 1 - \chi_f\ , \tag{3.7.3}$$

where

Re_t is the turbulence Reynolds number,

i is the stoichiometric ratio, and

m_{pr} is the mass fraction of products of combustion.

The temperature of the fluid in the fine structure, T_f, and that of the surrounding ambient fluid, T_a, are given by:

$$T_f = T + \Delta T\ , \tag{3.7.4}$$

$$T_a = T - \Delta T\ \chi_f / \chi_a\ , \tag{3.7.5}$$

$$\Delta T = H_R\ m_m / (\rho\ C_P)\ , \tag{3.7.6}$$

where H_R is the heat of reaction, m_m is the r.m.s. value of fuel species that can react (Equation 3.6.8), and C_P is the mean specific heat of fluid.

The actual rate of nuclei generation is computed on the assumption that it must be the smaller of that for a turbulent fluid and that for well-stirred reactor conditions. Thus:

$$S_{sn} = \text{minimum}\{S_{sn},\ S_{sn}(\chi_f=0)\}\ , \tag{3.7.7}$$

The oxidation rate for soot nuclei, the s_F term of Equation 2.1.1, is given as:

$$s_{sn} = S_{CnHm} / m_{CnHm}\ , \tag{3.7.8}$$

where S_{CnHm} is the reaction rate for the fuel.

Without accounting for turbulence, the source term for the i^{th} soot particle concentration equations, according to Edelman et al. (1973) is written as:

$$S_{si,k} = C_{so}\ T^{\alpha} m_{CnHm}\ m_2\ \exp(-C_{Ess}/T)\ , \tag{3.7.9}$$

whereas for a turbulent fluid, it is assumed to be given by:

$$S_{si,t} = m_{pi} \sum_j \{ (C_{s1} - C_{s2}\ N_{sn})\ N_{si}\ \chi_j\ \rho / \rho_j \}\ , \tag{3.7.10}$$

where m_{pi} is the mass of the i^{th} soot particle, and α and C's are empirical constants.

The actual source rate is taken to be the minimum of these two; so that:

$$S_{si} = \text{minimum}\{ S_{si,k}, S_{si,t} \} . \quad (3.7.11)$$

The oxidation rate for soot particles is assumed to be the smaller of the two given by Equation 3.7.8 and the semi-empirical formula of Nagle and Strickland-Constable (1962):

$$s_{si} = 12 P_{O2}[\psi K_1 /(1+K_2P_{O2}) + (1-\psi) K_3) A_s , \quad (3.7.12)$$

where P_{O2} is the partial pressure of oxygen in atmospheres, A_s is the total surface area available for oxidation, and:

$$\psi = 1 /\{1 + K_4/(K_3P_{O2})\} , \quad (3.7.13)$$

$$K_j = C_{sj} \exp(-C_{Esj}/T) , \quad (3.7.14)$$

with j standing for 1, 2, 3 or 4, and C's as empirical constants. The default values of these coefficients are summarized below.

TABLE 3.7.1: DEFAULT CONSTANTS FOR SOOT SOURCE TERMS

j	1	2	3	4
C_{sj}	2000	21.3	0.446	$1.51 \bullet 10^6$
C_{Esj}	15,100	-2060	7,640	48800

3.8 RADIATION HEAT TRANSFER

The thermal radiation is computed on the basis of the six-flux version of the Schuster-Hamaker (Hamaker, 1947) model. In this model, the net radiation flux in any coordinate direction, at any location in space, is described by the sum of two fluxes: one aligned with the positive and the other in the negative direction of the coordinate. As noted by Sidall (1972), other flux models, such as the Milne-Eddington and Schuster-Schwarzchild approximations, can be represented by the same form of equations with different coefficients. The differential equations describing the variations of the net radiation flux, R_i, in the i^{th} direction, can then be written as;

$$\frac{1}{r}\frac{\partial}{\partial x_i}(r^n \Gamma_i^R \frac{\partial R_i}{\partial x_i}) = (a_R + s_R)R_i - \frac{1}{3}s_R \sum_j R_j - a_R E \qquad (3.8.1)$$

where no summation is implied by the i index and the exponent n is 1 except for the θ coordinate when it is equal to -1. In this equation, a_R is the absorption coefficient defined as fraction of radiation absorbed per unit length, s_R is the scattering coefficient defined as fraction of radiation scattered per unit length, and E is the black body emissive power. The Γ, E and R_i, in turn, are defined as:

$$\Gamma_i^R = \frac{1}{aR + sR + \frac{1}{r}\frac{\partial r}{\partial x_i}} \qquad (3.8.2)$$

$$E = \sigma T^4 , \qquad (3.8.3)$$

$$R_i = \frac{1}{2}(I_{i+} + I_{i-}) \qquad (3.8.4)$$

where σ is the Stefan-Boltzman constant, and I_{i+} and I_{i-} are, respectively, the energy fluxes along the positive and negative directions of the i^{th} coordinate.

It is assumed that soot, carbon dioxide and water vapor all influence radiation heat transfer. Contributions due to other constituents are considered to be of secondary importance for heat transfer. The net influence of these constituents is accounted for by modifying the absorptivity and emissivity of the gas. The approach followed is adopted from Modak (1979). Modak, in turn, relied upon the work of Edwards & Balakrishnan (1973), DeRis (1979), Leckner (1972) and others.

3.9 TWO-PHASE FLOW

Currently, the software does not have a full two-phase flow capability. However, limited two-phase capability is available for special flow situations. As described in Section 3.7, **ANSWER™** is able to solve three transport equations for soot nuclei and soot particles. With appropriate modifications to the source terms, these equations can be used for transport of any particulate material, such as slurries.

For gas-liquid mixtures, an option exists to inject a liquid in a gaseous stream through one or more nozzles. This liquid is subdivided into a number of rays and sub-rays. Liquid droplets, of specified size distribution, are then individually tracked by an explicit Lagrangian method. During the transport of these particles, boiling, phase change, chemical reaction, droplet drag, and interaction between the droplet and the surrounding fluid are computed by analytic and empirical expressions. The complete details of the droplet transport are available in Mongia and Reynolds (1978).

3.10 FLUID, HEAT OR MASS SOURCES OR SINKS

The source terms for the fluid, heat and mass transport equations may be positive or negative in value. For example, if fluid is injected into a region, the source term for that region is positive. On the other hand, if fluid is withdrawn, the source term is negative; in such cases, the source term is more appropriately referred to as the 'sink' term. Both constant and variable sources or sinks are permitted. The variability of a source or a sink may be a function of time, space or any of the field variables. The variable source may be input from a table or as an analytic function. The tabulated values permit arbitrary variation of the source or sinks. The analytic options allow use of several common functional forms; the general form of these functions is:

$$S_F = S_F(\xi), \tag{3.10.1}$$

where S_F represents any of the flow, heat or mass sources or sinks, and ξ represents the independent variable of time, space or one of the field variables. An extensive library of such functions is integrated into the software and new functions are easily added. Polynomial, power law, trigonometric, exponential and logarithmic functions are available; these functions are described in Chapter 7.

A flow-induced and solubility-limited source or sink may also be specified. For a flow-induced source or sink, as the fluid is injected or withdrawn, other quantities, such as heat and mass and mass, are also removed or added along with the fluid mass. The formulation for this type of source is:

$$S_F = q_m F_{inj}, \tag{3.10.2}$$

where q_m is the rate of injection or withdrawal of fluid and F_{inj} is the value of F in the injected fluid. If fluid is injected, then F_{inj} is specified as part of the input. On the other hand, if fluid is withdrawn, then F_{inj} is the prevailing local value of the property and is computed as part of the solution process.

The solubility-limited option is available only for the concentration equations. In this option, it is assumed that a finite inventory of the source material, such as a chemical or radionuclide species, is initially present. The fluid then dissolves this species such that the maximum concentration in the fluid phase does not exceed the solubility of the species. Specifically, the rate of dissolution is given by:

$$S_C^k = f_S(C_S^k - C^k), \tag{3.10.3}$$

where S is the rate of species dissolution, f_S is a dissolution frequency, and C is the saturation limit of the species in the fluid. All of the functional forms that apply to S_F of Equation 3.10.1 are available for the C_S term; the latter may be either a tabulated or analytic function of time, space or one of the other variables. In implementing this algorithm, the amount of solute inventory is computed at every time step and the frequency term is given a large value as long as the source of the solute has not been exhausted; otherwise, it is given a value of zero.

The radioactive decay or a first-order Arrhenius chemical reaction of a species is given by:

$$\frac{\partial C}{\partial t} = -\lambda_k C, \tag{3.10.4}$$

where λ_k is the reaction-rate constant. With $t_{1/2}$ as the half-life of the k^{th} species, the λ_k is given by:

$$\lambda^k = \frac{\ln(2)}{t_{1/2}^k} \tag{3.10.5}$$

A fraction of the decay of the k^{th} species may generate the next species in the chain; this fraction is denoted by σ^{mk} and, along with $t_{/2}$, is specified as part of the input by the user.

3.11 BOUNDARY CONDITIONS

3.11.1 General Boundary Condition

The most general form of boundary condition for the transport equation is:

$$-\Gamma \frac{\partial F}{\partial N} = h_F (F - F_o) + q_F \tag{3.11.1}$$

where F represents any of the dependent variables, N is the direction coordinate normal to the boundary, and Γ, h_F, F_o, and q_F are specified quantities. Appropriate selection of these quantities results in Dirichlet, Neumann, or mixed (also called radiation, Robbins or "third kind") type of boundary conditions. The quantities F_o, q_F and h_F may be functions of time, space or any one of field variables. They may vary as tabulated analytic or user-defined functions. The wide ranges of functional forms that are available are described in Chapter 7.

3.11.2 Dirichlet Boundary Condition

In this type of boundary condition, the value of the dependent variable at the boundary is directly specified as:

$$F = F_o . \tag{3.11.2}$$

3.11.3 Neumann Boundary Condition

The Neumann boundary condition consists of specification of the normal gradient (or the flux) of the dependent variable at the boundary; that is:

$$-\Gamma \frac{\partial F}{\partial N} = q_F \tag{3.11.3}$$

Here, Γ is unity if the gradient of the variable is specified or it represents the appropriate component of the effective diffusion and dispersion tensor (Γ_e of Equation 2.2.1) if the flux of the variable is specified. It is internally computed from other input data and separate specification is not required. The quantity q_F is the specified boundary gradient or flux of fluid, heat, or chemical species.

3.11.4 Mixed Boundary Condition

The mixed boundary condition specifies a linear relationship between the boundary value and the normal gradient (or the flux) of the dependent variable at the boundary; that is:

$$-\Gamma \frac{\partial F}{\partial N} = \pm h_F (F - F_o) \tag{3.11.4}$$

In this equation, Γ has the same meaning as for the Neumann condition; h_F is the fluid, heat, or mass transfer coefficient, and F_o is the equilibrium value of F. The plus sign is used when the unit-outward-normal vector is aligned with the coordinate direction; otherwise, the negative sign is used. This sign convention assures that when F_o exceeds F, the flux is added to the system

3.11.5 The Default Boundary Conditions

INLET BOUNDARY: For the boundary nodes through which inflow occurs, the values of all dependent variables, except pressure for incompressible flow, must be specified by the user. In the absence of such a specification, the specified or default initial conditions are assumed to be the boundary conditions for these nodes. For incompressible flow, DEFCON method does not require the specification of pressure. Boundary

pressure is generated from the known fluid density and the constraint of the continuity equation. The built-in default treatment is equivalent to the assumption of Neumann boundary conditions.

OUTLET BOUNDARY: The outflow boundary conditions for all the variables, with one exception, are assumed to be of the Neumann kind. The exception is the velocity component normal to the outflow boundary for steady incompressible flow. For such flows, convergence is enhanced if the equation of continuity is explicitly satisfied at a global scale (that is for all the control volumes taken together). Mathematically, for a steady incompressible flow, the requirement of continuity is expressed as:

$$\rho \, U \, dA = M_{in} \, , \tag{3.11.5}$$

where ρ, U and A are, respectively, the density, the outward normal velocity and the outflow area for the outlet boundary, and M_{in} is the total inflow into the domain of interest. The integral is to be carried out for all the outflow nodes. The assumption made is that the velocity profile at outflow is similar to the velocity profile just inside the domain of interest. For example, if the outflow boundary values are denoted by o and those at the immediately upstream as, u, then it is assumed that:

$$U_o \; = \; C \, (U_u + |U_u|) \, , \tag{3.11.6}$$

where C is a constant scale factor for all the outflow nodes and the outflow velocity component is defined to be positive if the flow is leaving the domain of interest. By reference to Equation 3.11.5, C is then obtained from:

$$C \; = \; M_{in} \, / \int \rho_o \, (Uu + |Uu|) \, dA \, . \tag{3.11.7}$$

The outflow velocity can now be calculated from Equation 3.11.6. For multiple outflow ports, the requirement of global mass conservation is applied to each of the ports separately. If more than one outflow port exists, then it is necessary for the user to explicitly specify the fraction of total mass inflow rate (M_{in}) through each of these. This mass flow fraction is then used in Equation 3.11.7 to determine the correction factor for the appropriate velocity component at each outflow node. By default, all outflow nodes are assumed to belong to the same port and one value of the correction factor is applied to all the nodes.

SOLID WALL OR OBSTACLE BOUNDARY: It is a requirement of the no-slip condition at a solid wall that the fluid velocity at the wall be equal to the velocity of the wall. By default, the velocity components at all walls (including solid boundaries and internal obstacles) are set to zero. This implies that all walls are stationary. Of the other variables, temperature is also assumed to be specified. If no specification is made then the initial value is taken to be boundary value. For all other variables, the Neumann conditions are assumed to prevail at the wall; that is the normal flux at the wall is zero.

For laminar flow these conditions prove adequate. For turbulent flows, however, additional modifications are required. These are described below.

OPEN BOUNDARY: The default boundary condition for an open boundary is the Neumann boundary condition with the assumption that the flux of the variable across the boundary is zero.

AN AXIS OR A PLANE OF SYMMETRY: The default boundary condition at an axis (or a plane) of symmetry is the Neumann boundary condition with the assumption that the flux of the variable across the axis (or plane) of symmetry is zero. There is however an exception to this specification for the radial velocity component, V, which must be zero at the axis of symmetry.

3.11.6 Near Wall Regions for Turbulent Flow

INTRODUCTION: Turbulent flow near a wall is distinguished in two ways from flow far from the wall: the effect of molecular viscosity becomes prominent because of the damping effect of the wall on turbulence; and, some properties show steep variations in the vicinity of the wall. An economical way to incorporate the effects of wall proximity is by way of 'wall functions'. Wall functions are essentially embodied in algebraic expressions which ensure that the numerical solution behaves in a desired manner in the vicinity of the wall. An example of such a specification is that of 'the log-law' variation of velocity for turbulent flow in the vicinity of a wall. The practices adopted are outlined below; a more detailed treatment of this topic is available, for example, in Launder and Spalding (1972).

THE MOMENTUM VARIABLES: Consider the adjacent grid nodes P and N as shown in Figure 3.11.1. The wall is located midway between the nodes P and N at a normal distance from the node P (equal to half the distance between nodes P and N).

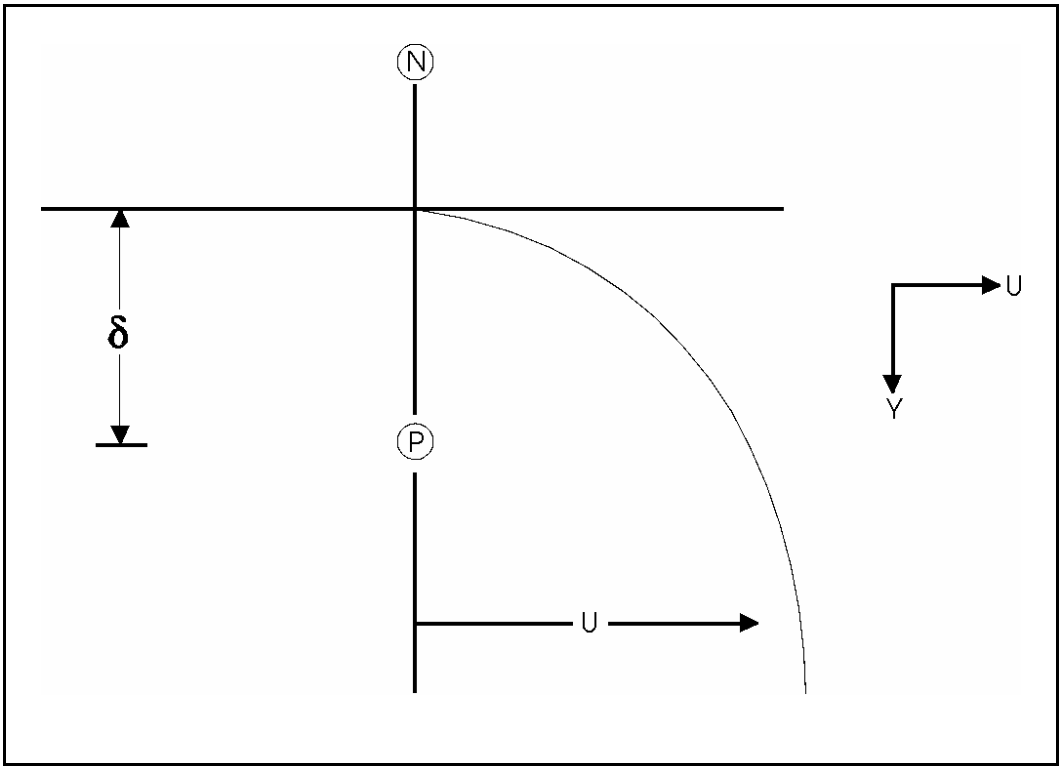


FIGURE 3.11.1: ILLUSTRATION OF NEAR-WALL REGION

The wall function modification for the velocity component is based on the assumption that 'log-law' prevails between the wall and the node P; that is:

$$U = U_w + (u^*/\kappa) \ln(E\, y^+) , \qquad (3.11.8)$$

where U is the velocity component parallel to the wall, U_w is its value at the wall, u^* is a friction velocity, κ is the von Karman constant, E is a constant in the log-law of the wall and, y^+ is a non-dimensional distance from the wall. With τ_w as the wall shear stress and y as the distance from the wall (Figure 3.11.1), the u^* and y^+ are defined by:

$$u^* = (\tau_w / \rho)^{1/2} , \qquad (3.11.9)$$

$$y^+ = \rho\, u^*\, y / \mu , \qquad (3.11.10)$$

A combination of Equations 3.11.8 and 3.11.9, yields an expression for the wall shear stress as:

$$\tau_w = \kappa\, \rho\, u^* (U - U_w) / \ln(E\, y^+) . \qquad (3.11.11)$$

For the k-ε model of turbulence, it can be shown that (Launder and Spalding, 1972) in the vicinity of the wall where the shear stress is constant:

$$u^* = C\, \mu^{1/4}\, k^{1/2} , \text{ and} \qquad (3.11.12)$$

$$\varepsilon = u^{*3} / (y\, \kappa) . \qquad (3.11.13)$$

The turbulent viscosity, μ_t, is obtained from Equations 3.2.2, 3.11.12 and 3.11.13 as:

$$\mu_t = \kappa\, \rho\, u^*\, y . \qquad (3.11.14)$$

From Equations 3.11.10 and 3.11.14, the non-dimensional distance y^+ can be expressed in terms of μ_t as:

$$y^+ = \mu_t / (\kappa\, \mu) . \qquad (3.11.15)$$

Substitution of the expression for the turbulent viscosity, Equation 3.11.14, in Equation 3.11.11 results in:

$$\tau_w = \mu_t (U - U_w) / \{y \ln(E\, y^+)\} . \qquad (3.11.16)$$

Since $(U - U_w)/ y$ represents the numerical equivalent of the velocity gradient normal to the wall, it is seen that the net effect of the wall functions is to replace this gradient with an effective value given by:

$$(\partial U/\partial y)_e = (\partial U/\partial y) / \ln(E\, y^+) , \qquad (3.11.17)$$

$$(\partial U/\partial y) = (U - U_w) / y , \qquad (3.11.18)$$

where the subscript e denotes the effective value of the velocity gradient in the vicinity of a wall.

The wall functions are employed only if the node P is situated well outside the viscous sub-layer. This is assumed to be the case if the value of y^+ at node P is greater than some reference value y; that is:

$$(y^+)_P > y . \qquad (3.11.19)$$

Provided that the criterion 3.11.19 is met, the effective value of the velocity gradient at node P (Equation

3.11.17) is employed to calculate the wall shear stress in the momentum equations; otherwise Equation 3.11.18 is employed. This modification is made for each of the velocity components parallel to a wall. For the velocity component normal to the wall no such modification is required since its gradient does not exhibit steep variations.

THE TEMPERATURE VARIABLE: For temperature (or enthalpy) variable, no modifications are necessary if the wall heat flux is specified. However, if wall temperature is specified then wall functions are required since the temperature, like the velocity component U, may vary steeply in the vicinity of the wall. For this case, the approach adopted is similar to that for the velocity components: the numerically computed value of the temperature gradient at the node P is replaced by an effective value. The wall heat flux, q_T, is then given by:

$$q_T = (\mu_t / \sigma_t^T) (\partial T/\partial y)_e , \qquad (3.11.20)$$

$$(\partial T/\partial y)_e = (T_P - T_w) / \{y [\ln(E\, y^+) + \kappa P_T]\} , \qquad (3.11.21)$$

$$P_T = 9.0 \{(\sigma_T/ \sigma_t^T) - 1\} (\sigma_t^T / \sigma^T)^{1/4} , \qquad (3.11.22)$$

where subscript e denotes the effective value of the temperature gradient at node P, P_T is a resistance to heat transfer, σ^T is the Prandtl number and, σ_t^T is the turbulent Prandtl number.

TURBULENCE KINETIC ENERGY AND DISSIPATION: For the kinetic energy k, it is assumed that the gradient normal to the wall is zero. In addition the velocity gradients normal to the wall which occur in the production term for k (P_k of Equation 2.2.6) are replaced by their effective values based on the log-law.

For the dissipation variable, ε, a somewhat different approach is used. The value of ε at the near-wall node, P, is calculated directly from Equation 3.11.13 above.

THE MASS SPECIES: For mass species, the current versions of **ANSWER™** assume that the gradients normal to wall are zero; no wall functions are necessary. This is equivalent to assuming that the flux across the wall is zero that is consistent with most applications. However, if the concentration of mass species is specified, then a treatment identical to that for the temperature variable can be easily incorporated.

3.12 INITIAL CONDITIONS

The governing equations require specification of a well-posed set of initial conditions for all field variables. These may be specified explicitly or are assigned a suitable value by default. Any field variable may be set initially, or during the solution process. It may be constant, or a function of time, space or any one of the other field variables. The values may be set by tabulated or analytic functions. The functional forms available for this purpose are described in detail in Chapter 7.

3.13 FIELD VARIABLE INTERRELATIONSHIPS

In certain instances an interrelationship exists between the field values of a dependent variable. The relationship may be with respect to an internal boundary between two immediate neighbors or it may be a more general, neighborhood interrelationship. The relationship between two immediate neighbors is that of a flux exchange and is governed by Equation 3.11.3 or 3.11.4. These relationships may be specified in terms of the boundary flux exchange, in a manner analogous to that for the boundary conditions for the model domain.

On the other hand, a more general form of the neighborhood interrelationship is given by:

$$F_P = F_0 + a_n \sum_{n=1}^{N} F_n , \qquad (3.13.1)$$

where the subscript P denotes the value of the variable at a location P and n in its immediate neighborhood. In this equation, F_o is the datum value and a_n are specified or derived coefficients. The user may explicitly specify these relations at any internal node in the flow domain.

3.14 SPECIAL FEATURES

3.14.1 Conjugate Heat Transfer

In most computational fluid dynamic applications, the fluid is the focus of concern. However, some applications involve heat transfer both in the fluid and in the surrounding or embedded solid components. Such conjugate heat transfer can be accommodated by a simple user command. In such a case, the thermal properties of the solid, which are assumed to be constant, must be specified by the user.

3.14.2 Nozzles and Sprays

ANSWER™ provides for injection of fluid in the form of a spray through one or more nozzles. The fluid in the spray is divided into a range of droplets of varying sizes. A representative selection of droplets is then tracked in a Lagrangian mode by dividing the nozzle into a number of rays and sub-rays. During the transport process, these droplets may heat, evaporate and undergo chemical reaction and mass transfer. Each droplet is tracked till it evaporates completely or leaves the region of interest.

3.14.3 Moving Boundaries

Any of the external boundaries of the domain of interest may be specified to be moving at an arbitrary velocity. By default, it is assumed that all solid boundaries are stationary. However, simple user specification, through generalized initial and boundary commands may be used to specify moving boundaries.

3.14.4 Moving Embedded Objects

A special feature of **ANSWER™** is its ability to accommodate a object moving at an arbitrary velocity through the flow field. The object may be surrounded on one or more sides by fluid (such as a torpedo or missile moving inside its launch tube) or it may act as a moving barrier (such as a moving piston in a cylinder). The rate of movement of such objects may be varied with time.

(This page left intentionally blank.)

CHAPTER 4

NUMERICAL BASIS

The governing partial differential equations are solved by the Nodal Point Integration method. A choice of 'integration profiles' or 'basis functions' is provided for spatial discretization of the equations. The choices are the hybrid , the CONDIF and the QUICK numerical schemes. The matrix of algebraic equations resulting from the discretization process is solved by one of several matrix solution methods. Numerical solution of the governing equations described in Chapter 2 requires three steps: (1) A grid arrangement to divide the domain of interest into discrete elements, (2) discretization and derivation of the algebraic analogues of the equations by a numerical scheme, and (3) solution of the resulting algebraic equations by a matrix solver. Like all numerical methods, the numerical stability and accuracy of the algorithms are functions of the discretization process. These numerical features are discussed in this chapter.

4.1 THE GRID ARRANGEMENT

The first step towards obtaining a numerical solution is spatial discretization of the domain of interest. The field of interest is divided into contiguous elements, each of which individually is a control volume. A nodal point, which represents the intersection of the three coordinate lines, is associated with each element. Examples of such a grid, in Cartesian and cylindrical coordinate systems, are respectively shown in Figures 4.1.1 and 4.1.2. In these figures each element encloses one grid node. The element may be any arbitrary quadrilateral in 2D or a hexahedral in 3D space. Such an element has 4 sides and 4 corners in 2D, and 6 sides and 8 corners in 3D space. All elements contain a single interior node that may or may not be located at the centroid of the element. An additional node is placed at the boundary of an element if that boundary is not connected to another element. As indicated in Figures 4.1.1 and 4.1.2, a unique node number denotes each node. For structured grids, each node also has a unique set of grid index numbers associated with it. For 2D domain the grid indices (I,J) denote the location in the (x,y) or (x,r) space. For 3D domain the node is denoted by a unique (I,J,K) index in the respective grid directions (x,y,z) or (x,r,θ).

A co-located grid approach is used. All the state variables except for the fluxes are defined at the grid nodes. The fluxes of heat and mass are internally computed at the element boundaries (solid lines). Details of this arrangement of variables and the integration element are shown in Figures 4.1.3 through 4.1.5. This leads to a natural description of the physical system in which fluxes are defined at the element boundaries and intrinsic properties are defined at the node points.

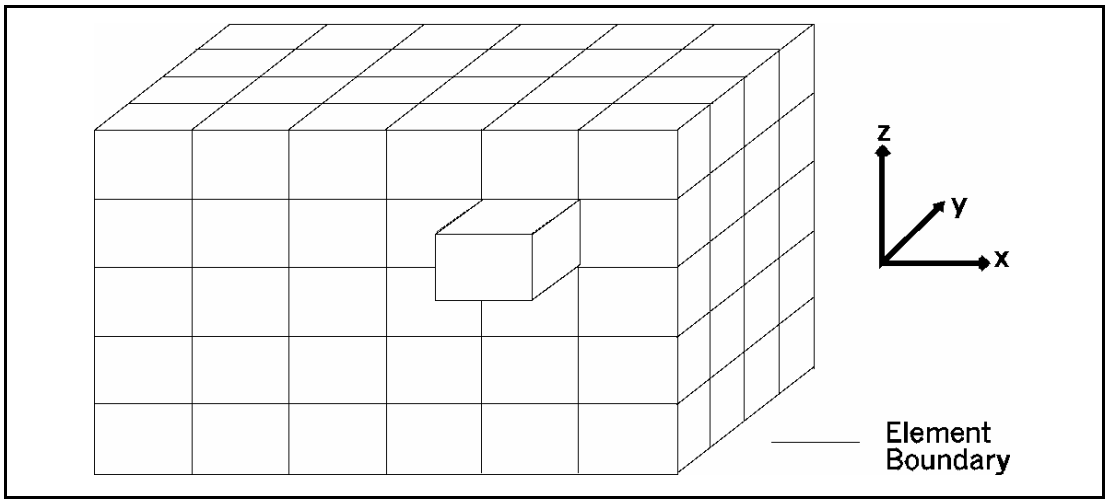


FIGURE 4.1.1: CARTESIAN GRID ARRANGEMENT

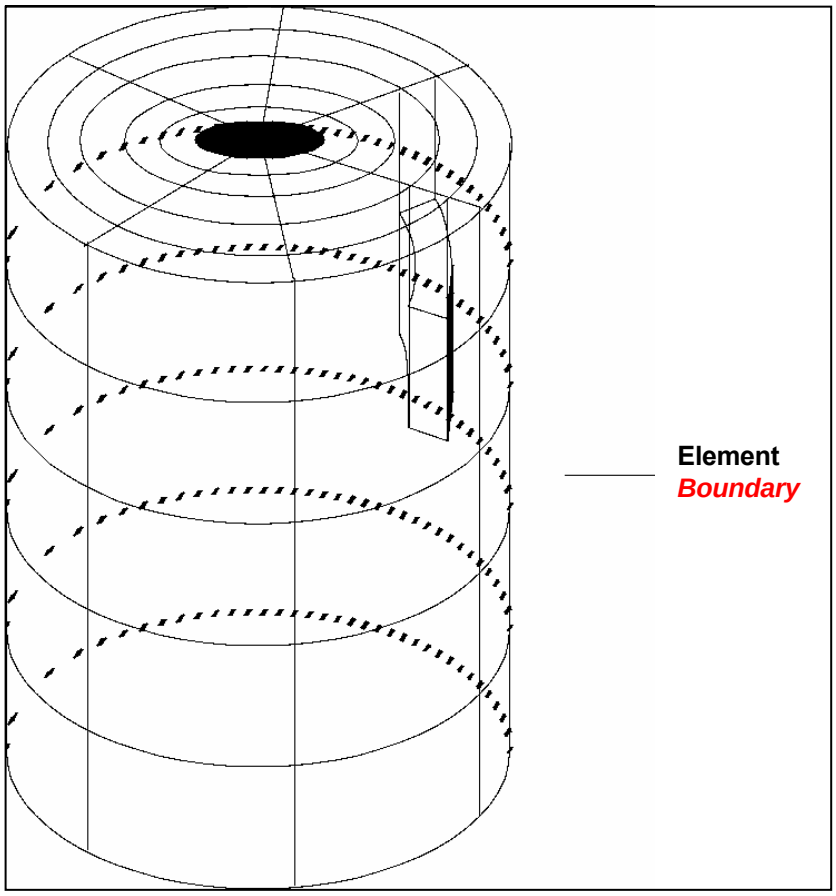


FIGURE 4.1.2: CYLINDRICAL GRID ARRANGEMENT

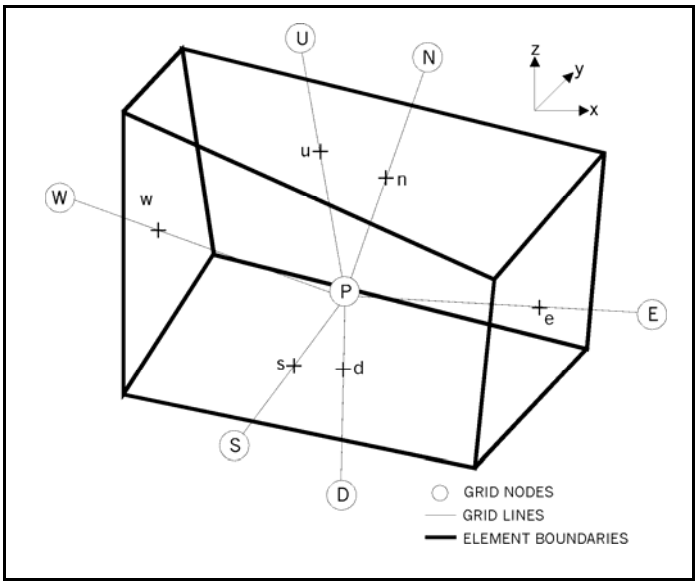


FIGURE 4.1.3: TYPICAL GRID ELEMENT IN CARTESIAN COORDINATES

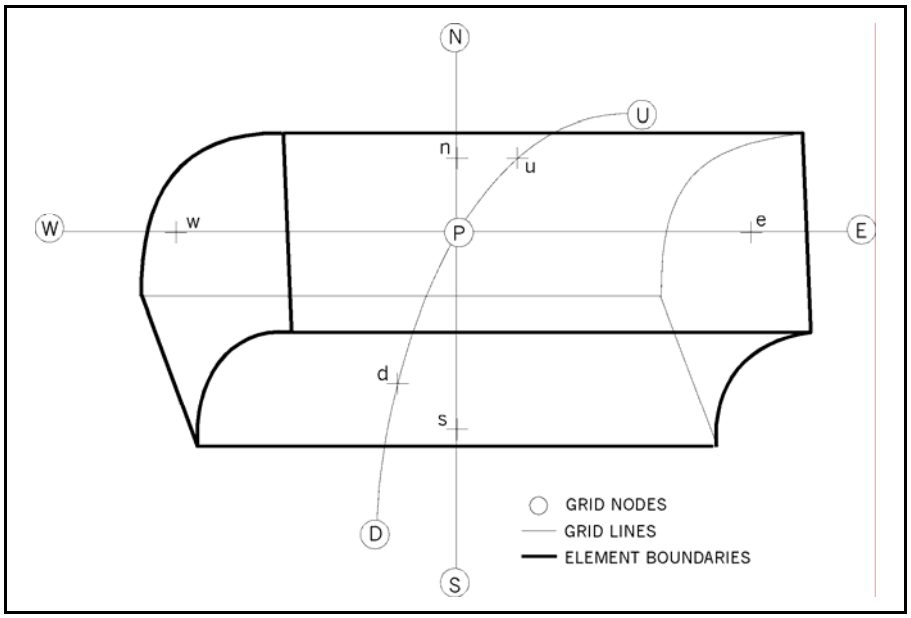


FIGURE 4.1.4: TYPICAL GRID ELEMENT IN CYLINDRICAL COORDINATES

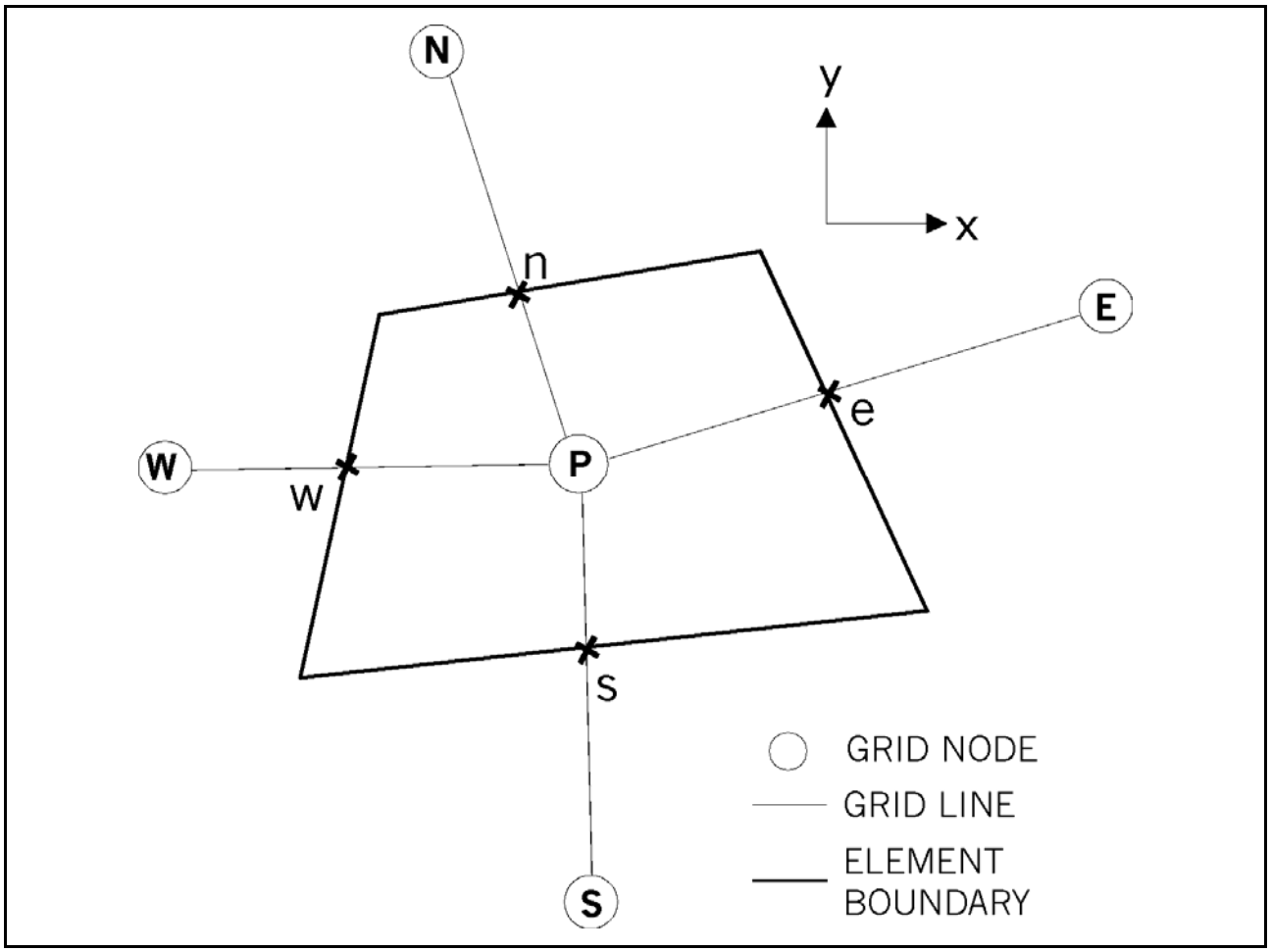


FIGURE 4.1.5: TYPICAL GRID ELEMENT IN X-Y PLANE

4.2 DISCRETIZATION SCHEME

The discretized form of the governing transport equation is obtained by integrating it over each element, such as the one shown in Figure 4.1.5. The integration method used is termed Nodal Point Integration (NPI). It is a variation of the `finite-volume' method and somewhat resembles the finite-element method. However, it is simpler and more economical, and a major advantage of this method is that it intrinsically preserves the mass, material and thermal fluxes both at local and global scales. Consequently it is inherently mass-conservative and typically results in more accurate and stable numerical formulations than finite-difference and finite-element methods. The details of the NPI method are available, for example, in Runchal (1987a).

The numerical approach employed for spatial integration results in unconditional linear stability. The spatial variation of state variables is approximated by appropriate profiles to ensure accuracy and stability. The governing transport equation consists of three types of terms: the convective, the diffusive, and the source term. The convective and the diffusive terms are treated in a unified manner; the source terms are treated individually (Runchal, 1987a).

The numerical integration starts with the assumption of an integration profile for the state variable. Two different kinds of profiles are employed. These are the first- and second-order profiles. These integration profiles result, respectively, in the `upwind', and the central difference schemes. These are combined in a hybrid scheme. The central difference scheme, which provides second-order accuracy, is the preferred scheme. However, use of the central difference scheme may result in numerical instabilities if the magnitude of the local value of the grid Peclet number exceeds 2. With U, δL and Γ, respectively, as the velocity component, grid interval and diffusivity in a given direction, the grid Peclet number, Pe, is defined as:

$$Pe = U \delta L / \Gamma \qquad (4.2.1)$$

The local value of the Peclet number at each grid node is constantly monitored in each direction. If Pe > 2, then the numerical scheme automatically shifts to the 'upwind' formulation. This method of enhancing stability is known as the hybrid scheme (Runchal, 1972). The hybrid scheme has second-order accuracy if the Pe < 2; otherwise, it is only first-order accurate. Because upwinding results in an increasing amount of numerical diffusion as the angle between the velocity vector and the grid lines increases. An alternate method to obtain numerical stability with second-order accuracy is that of the CONDIF scheme (Runchal, 1987b) which is a modified central-difference scheme. It is a second-order member of the TVD family of numerical schemes (Harten, 1983) that leads to an unconditionally stable formulation. A third option which is available is that of a version of the QUICK scheme (Leonard, 1979) which has been adapted for non-orthogonal grids.

The user controls the method of evaluation of the integrals, which is equivalent to the selection of a ‘basis function’ in the finite-element technique. For most problems, the hybrid scheme is sufficient. If the grid is very coarse, then the CONDIF or the QUICK scheme should be employed.

The discretization with respect to time may be either explicit or implicit. With the explicit scheme, all terms of Equation 2.1.1 except the storage term, $\partial(\alpha F)/\partial t$, are evaluated from known values at the previous time step. The values at the current time step can then be calculated by a backward discretization of the storage term. In an implicit scheme by comparison, some or all values of the convection, diffusion and source terms appearing in Equation 2.1.1 are taken at the new time step. Each algebraic equation in this case has more than one unknown, and matrix methods are required to solve the set of equations.

An algebraic analog of the general transport equation, Equation 2.1.1, can be written in terms of appropriate expressions for the storage, convection, diffusion and source terms. This analog relates the value of a dependent variable at the central node to the value of the variable at the neighboring nodes by the equation:

$$A_P (F_P^{k+1} - F_P^k) + \sum_K A_P [\ \Theta\ (F_P^{k+1} - F_K^{k+1}) + (1-\Theta)(F_P^{k+1} - F_K^{k+1})\ ]$$
$$= \delta V\, S_{FP} - s_{FP} [\ \Theta\, F_P^{k+1} + (1-\Theta)\, F_P^k\ ] \qquad (4.2.2)$$

In Equation 4.2.2, the subscript P denotes the node point at which the discrete equation is derived, δV is the volume of the element at P and, summation is carried out over all of the neighboring nodes shown in Figure 4.1.3; i.e. the subscript K denotes the grid node values at E, W, N, S, U and D, respectively. The coefficients, the A's in the above equations, are functions of grid size, fluid velocity, and properties of the fluid. Their exact forms depend on the spatial and temporal discretization scheme that is adopted. The superscripts k and k+1 denote the value of the variable F at two successive time steps. The parameter Θ determines whether the numerical scheme is explicit (Θ=0), implicit (Θ =1), or semi-implicit (0 < Θ < 1). The choice of Θ is made internally and is based on the method of solution selected by the user.

4.3 SOLUTION METHOD

One algebraic equation is obtained for each dependent variable at each grid node. To solve the resulting set of equations, a matrix solver is required. Several options are available. Two matrix solvers are provided as standard options; these are the Alternating Direction Implicit (ADI) and the Successive Over-Relaxation (SOR) method. Other solvers are available as fully integrated modules. In addition, an open framework is used whereby an interface for any other matrix-solver package selected by the user is provided.

The ADI method solves the set of algebraic equations in three sweeps, one each along the x-, y- and z-coordinate directions. It is semi-implicit in nature. For each sweep, Equation 4.2.2 is written with only the values at the grid nodes in that direction at the advanced time step. All other values are taken from the previous time step. In this manner, each equation has only three unknowns which gives rise to a tri-diagonal system of equations that are solved by the Thomas Algorithm (Varga, 1962). The SOR method employs an explicit scheme in which there is only one unknown (F^{+1}) per equation; all other F's are known from the previous time step, previous iteration, or initial conditions. Equation 4.2.2 can then be solved for the new value at each node. For steady state solutions, the process is repeated until the calculated values do not change by more than a specified tolerance limit.

A number of other methods are available as options. These include Cholesky Decomposition, Gaussian Elimination and Conjugate Gradient methods. Any other matrix solver can easily be integrated by means of a flexible module that provides access to all matrix coefficients and the forcing function. The solution vector is the output from the matrix solver that is then returned to the parent **ACRi** Software tool.

4.4 NUMERICAL STABILITY AND ACCURACY

4.4.1 General Considerations

The selected numerical schemes and solution methods represent the best options currently available for solving a wide variety of flow and transport problems. In general, they provide accurate, economical and numerically stable solutions. However, as in all numerical methods, instability may occur when the physical process being simulated exhibits non-linearity or when strong coupling exists between various components of the model. This instability may exhibit itself either as "weak instability" or "exponential growth". Weak instability usually occurs as a step-to-step oscillation about a mean value. Exponential growth, on the other hand, occurs as an uncontrolled growth in the values of state variables.

The hybrid and CONDIF numerical schemes are unconditionally stable in a linear sense (von Neumann analysis) for any spatial discretization. The stability of the temporal discretization depends on the solution method adopted. The direct matrix solvers always provide a solution (except for round off) of the matrix of algebraic equations, provided one exists. The ADI and Conjugate gradient methods are stable in a linear sense. The SOR is conditionally stable provided that certain criteria are met. Note that no method currently exist to establish the stability of the complete, coupled, non-linear system of equations. Instabilities in such systems may arise due to the coupling and non-linear terms. The numerical accuracy of the schemes employed varies from first- to second-order with respect to spatial discretization. With respect to temporal discretization, the accuracy is second-order for the ADI and first-order for all other solution methods.

The accuracy of the numerical solution improves with decreasing grid interval and time step. The stability of a solution scheme, on the other hand, generally improves with a decrease in the time step but is adversely affected by a decrease in the grid interval. The numerical efficiency, or economy, is often adversely affected by a decrease in both the grid interval and the time step. Thus, considerations of numerical stability, accuracy and economy often impose conflicting requirements on the discretization process. The physical processes and numerical approximations inherent in **ANSWER™** imply the existence of several characteristic temporal and spatial scales. For efficient, accurate and stable numerical solution, these scales must be duly considered in selecting the grid distribution and time step. The influence of the time step is, of course, limited to transient solution process; it plays no role in steady state simulations. These issues are discussed in more detail below.

4.4.2 Time Scale of Pressure Propagation

The characteristic time scale for propagation of transient pressure effects is given by:

$$\delta t_{press} = \delta L / c \qquad (4.4.1)$$

where δL is a representative grid interval and c is the speed of sound.

This time scale is important for transient compressible flow problems; all pressure disturbances propagate across a grid interval of length δL in this time. For incompressible flows, the sonic speed is essentially infinite, and this time scale no longer influences the solution. For explicit schemes (e.g., SOR), the time step employed must not exceed the value given by Equation 4.4.1. Although no numerical stability conditions apply for the other methods, it is recommended that for accurate representation of transient effects, the time step should be within an order of magnitude of the value calculated from Equation 4.4.1

For iterative schemes, numerical instability sometimes may be encountered due to non-linear and coupling effects if the time step employed is much larger than the time scale given by Equation 4.4.1. In such cases, either the time step must be reduced or a direct solution method must be adopted.

Many problems of practical interest are distinguished by a pressure propagation time which is very small or zero. In such cases, provided the accuracy of the transient component of the solution is not of prime importance, it is more economical to use **ANSWER™** in its quasi- or fully- steady state mode. The alternative is to use an artificially low value of the sonic speed that then allows the use of large time steps. In such a case, the inverse of the sonic speed essentially acts as a relaxation parameter (see, for example, Varga, 1962) in the solution procedure. In this instance, the transient solution obtained will not be accurate.

4.4.3 Time Scale of Diffusion

An important time scale for physical phenomena that are significantly influenced by diffusion is given as:

$$\delta t_{diff} = \alpha_e \delta L^2 / (2\Gamma_e) \tag{4.4.2}$$

where α_e is a representative value of the storage coefficient, and Γ_e is a representative value of the effective diffusivity.

The general implications of this time scale are similar to those of the pressure propagation scale. The time step that is employed should be kept within an order-of-magnitude of the value given by Equation 4.4.2. For the explicit solution method, the time step employed must not, for reasons of stability, exceed the value given by this equation.

4.4.4 Time Scale of Convection

With U and δL respectively as the characteristic velocity component and grid interval, the convection time scale is defined as:

$$\delta t_{conv} = \delta L / U \tag{4.4.3}$$

The constraint based on this time scale is often stated in terms of the Courant number, Co, which is defined as:

$$Co = \frac{\delta t}{\delta t_{conv}} = U \frac{\delta t}{\delta L} \tag{4.4.4}$$

where δt is a representative time step for numerical solution

The explicit solution method requires that the Courant number must not exceed unity; the other methods are not subject to any constraint based on the Courant number, as indicated by linear stability analysis. Nonetheless, for useful simulation of a physical process that is significantly influenced by convection, the time step that is employed should be within an order of magnitude of the convection time scale for the physical process, δt_{conv}.

4.4.5 Other Pertinent Time Scales

For many problems, it is likely that other pertinent time scales may need to be considered. The need for their considerations may arise, for example, from the presence of uniform or time-varying fluid, heat or mass sources, propagating discontinuity fronts, buoyancy, chemical reaction, radioactive decay, and time-dependent boundary conditions. Each of these imposes physical and numerical time-scale constraints on the solution process. These constraints can be determined by recourse to the corresponding governing differential equation.

4.4.6 Grid Peclet Number

The grid Peclet number (Equation 4.2.1), plays an important role in the numerical stability and accuracy of a numerical scheme if both the convective and diffusive terms are not zero (ß and Γ of Table 2.2.2). The numerical schemes used are unconditionally stable for arbitrary values of the grid Peclet number. However, the numerical accuracy, especially that of the steady state component of the solution is strongly governed by the grid Peclet number (Roache, 1972; Runchal, 1977). Depending on the assumptions made for spatial variation of the state variables, it can be shown that significant numerical errors may occur if the local grid Peclet number is larger than 2 and the flow is at a significant angle to the grid lines.

The spatial profile employed to integrate the governing equations is chosen on the basis of the local value of the grid Peclet number. If the grid Peclet number is less than 2, a second-order polynomial (equivalent to a central difference-scheme) profile is used. If, however, this number exceeds 2, then **ANSWER™** provides three options. These options are the hybrid, the CONDIF and the QUICK methods, as discussed in Section 4.2. The actual choice must be based on a balance between accuracy and economy.

In practice, the second-order approximation with automatic shift to upwind differences is often adequate for most problems. However, the CONDIF or QUICK scheme may be advisable if greater accuracy is desired.

4.5 CONVERGENCE

Both iterative and direct methods are available to solve the algebraic equations that result from the discretization process. The algebraic equations in matrix from are:

$$A_{ij}\,\phi_{ij} = b_j \tag{4.5.1}$$

where A_{ij} is the matrix of coefficients, ϕ_i is the unknown solution vector and b_j is the forcing function. For direct solution methods, the matrix inversion process leads to an exact solution of the above set of equations (within the round-off error limits of the machine). For iterative methods, the matrix is rearranged as:

$$\phi_i^k = T_{ij}\,\phi_i^{k-1} + c_i \tag{4.5.2}$$

where ϕ^k is the k^{th} approximation to ϕ_i, T_{ij} is a modified coefficients matrix and c_i is a modified forcing function. This process starts with an initial guess for the solution vector and continues until the successive iterations converge to a pre-specified tolerance.

Three criteria are available to test for convergence. These are based on the matrix residue, the normalized measure of change in the successive solutions, and the absolute change in successive solutions. For any grid element, i, with up to m neighboirs and k as the counter for successive solutions, these are expressed as:

$$R_1 = \left| \sum_{j=1}^{m} A_{ij}\phi_j - b_i \right| \tag{4.5.3}$$

$$R_2 = \left| 1 - \phi_i^k/\phi_i^{k-1} \right| ; \quad \phi_i^{k-1} > \phi_{min} \tag{4.5.4}$$

$$R_3 = \left| \phi_i^k - \phi_i^{k-1} \right| \tag{4.5.5}$$

where ϕ_{min} is a minimum value of the variable. Any of these measures of change can be used to define a convergence norm based on a global average or a local maximum as given below:

$$R_{max} = \frac{1}{N}\sum_{n=1}^{N}(R_n) \le \varepsilon \ , \tag{4.5.6}$$

$$R_{max} = \max\,(R_n) \le \varepsilon, \ i = 1, 2, 3, \ldots, N, \tag{4.5.7}$$

where ε is a small quantity (typically on the order of the machine accuracy) and N is the total number of grid elements.

(This page left intentionally blank.)

CHAPTER 5

PROBLEM GEOMETRY AND GRID

The first step in solving a particular problem is to specify the problem geometry and a corresponding grid for spatial discretization. Details of this input are given in Chapters 6 and 7. This chapter provides suggestions that may be useful in selecting and specifying a geometry and grid that are suited to a specific problem. The geometry of the problem is specified in terms of six basic geometrical elements. These consist of 1) Inlet, 2) Outlet, 3) Wall, 4) Obstacle, 5) Open boundary or free surface, and 6) Axis or plane of symmetry. These divide the domain of interest into a number of subregions. This chapter outlines the manner in which the computational domain is defined and later used to accommodate the disparate components of the geometry, physics and the computational requirements of a problem.

5.1 SPECIFICATION OF PROBLEM GEOMETRY

The problem geometry is defined by a set of Cartesian (x, y, z) or cylindrical (x,r,θ) coordinates for the vertices of the elements or of the grid nodes. The domain of interest is specified in terms of the six basic elements: 1) *Inlet,* 2) *Outlet,* 3) *Wall,* 4) *Obstacle,* 5) *Open boundary or free surface, and* 6) *Axis or plane of symmetry.*

Once the physical domain of the problem has been identified in terms of these material types, the next step is to impose a discrete spatial grid over the domain of interest (Figures 4.1.1 and 4.1.2). Criteria and suggestions for the selection of this grid are discussed in Section 5.2. Intersections of the grid lines are called the grid nodes. Each node is associated with an element that is formed by the grid lines (Figure 4.1.3). The location of each grid node is identified by its physical coordinates (x, y, z or x,r, θ) and by a unique set of grid indices (I,J,K) where I, J and K refer, respectively, to the x, y (or r) and z (or θ) direction. All the state variables are defined at the grid nodes. The fluxes crossing the element boundaries are internally defined and are computed at the boundaries.

The grid must be selected in a manner that makes the physical boundaries of each zone coincident with the element boundaries. This arrangement allows the fluxes across the zone boundaries to be correctly incorporated into the solution algorithm. The fluxes at the boundaries of the outermost zones are computed from the specified boundary conditions.

5.2 SPECIFICATION OF SPATIAL AND TEMPORAL GRID

5.2.1 Basic Considerations

Considerations that influence the design of a computational grid include:

- *Computer resources*
- *Desired numerical accuracy*
- *Special geometric and physical features of the problem*
- *Stability of the numerical solution process*
- *Nature and type of boundary conditions*
- *Special output requirements*

Some of these considerations result in conflicting requirements. For example, the numerical accuracy generally increases with the number of nodes, but so does the cost. Also, it may not be possible to resolve all the special features of geometry or boundary conditions within the constraints of the available computer resources. Hence, compromises may be necessary in the design of a computational grid.

5.2.2 Computer Resources

The first step in selecting a grid size and time step is consideration of the available computer memory size and computation time. These considerations provide an upper limit on the number of nodes and time steps that can be employed for numerical simulation and lead to average spatial (δx, δy and δz, or $\delta\theta$) and temporal (δt) scales of resolution.

5.2.3 Numerical Stability and Accuracy

The criteria of stability and accuracy that were outlined in Section 4.4 should be used to determine the approximate upper and lower limits of the grid size and time step. These determinations should be based on known or expected values of the other pertinent parameters (such as velocity, viscosity, etc.) of the problem. Any adjustments of grid size or number of time steps required by these criteria should then be made accordingly.

5.2.4 Scale of Heterogeneity

The grid should adequately represent variations in fluid properties. The grid should be finer where the properties vary relatively rapidly, and coarser where the variation is more gradual. All other factors being equal, a uniformly accurate solution may be expected if the values of the dependent variables and the properties of interest vary uniformly across the domain of interest.

5.2.5 Scale of Physical Resolution

For a variety of reasons, solution may be needed at a finer scale in specific parts of the domain than in other parts. For example, interest may be focused on those areas where temperatures are high. In such areas, smaller elements should be used. On the other hand, it may prove necessary to incorporate other features at a larger scale. For example, for a problem involving injection of fluid through a small hole, the size of the hole may be too small (say, millimeters) in comparison to the spatial scale of the problem (say, on the order of meters). In such a case, the injected fluid is typically distributed over an entire grid element that may have a length dimension on the order of a few centimeters. The resulting solution will not be accurate for prediction of local effects close to the point of injection; however, it will provide acceptable approximation of the effects of injection on the rest of the flow field.

5.2.6 Sources and Sinks

The dependent variables are expected to change relatively rapidly close to sources and sinks. Unacceptable errors may occur in the solution if large elements are used in such areas. As a general principle, finer-mesh grids should be used in areas where the values of the state variables are expected to change rapidly. If this proves to be prohibitive because of cost, then it may be necessary to distribute the source (or sink) over a larger area with the knowledge that the solution in the immediate vicinity of the source may be relatively inaccurate.

5.2.7 Boundary Conditions

The physical boundaries of the problem geometry must coincide with the element boundaries. Some boundaries are natural geometric features. For instance, a wall forms a boundary at which it is appropriate to specify velocity components from no-slip constraint. For most walls, element sizes should be comparatively small in close proximity to the wall. Other boundaries, such as an open or outlet boundary, do not represent geometric features and should be located at distances sufficiently far that they do not affect the solution. Near these boundaries, coarse-mesh grids can be used.

5.2.8 Input and Output Requirements

Occasionally it may be necessary to adjust the grid size and time step to accommodate specific input or output requirements. For example, output may be required at a regular frequency during a simulation. For these instances, the time step may be adjusted such that it is an integral fraction of frequency of output. Similarly, data for comparison may be available only at specific locations or the boundary conditions may change at specified space or time intervals. These and similar factors must be accounted for in a judicious selection of grid size and time step.

(This page left intentionally blank.)

CHAPTER 6

INPUT AND OUTPUT OPTIONS

Answer™ is a flexible and user-oriented software package. The computer program is modular in nature so that selected parts can be updated without significantly changing the overall structure of the program. The software employs the freeform™ command language to provide an adaptable, simple to use and format-free user interface. The structure and syntax of the command language are fully described in appendix b. 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 ANSWER™.

6.1 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 Appendix B.

The commands that constitute the I/O interface are listed in Table 6.1.1. These commands, and the modifiers and numeric data that are associated with them, are described in alphabetical order in Chapter 7. Illustrations of the input commands are included as Appendix C. 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 ANSWER™ AND THEIR FUNCTIONS

NO.	KEYWORD	INPUT FUNCTION
1	**ADAP**TIVE	Refine the specified grid by to Adapt to the computational solution
2	**ADIA**BATIC	Specify adiabatic external walls
3	**ALLO**CATE	Allocate table space for storage of user input
4	**ARRH**ENIUS	Specify the Arrhenius reaction rate constants
5	**BANN**ER	Print user & program identification to output file
6	**BLOC**KAGE	Specify blockage or solid objects in flow field
7	**BOUN**DARY	Override built-in boundary conditions
8	**CLOS**E	Close specified output device immediately
9	**COMP**RESSIBLE	Select compressible flow option
10	**COND**UCTIVITY	Specify conduction or diffusion constants, coefficients and options
11	**CONN**ECTIVITY	To specify element and vertex connectivty for unstructured grid
12	**CONV**ERGENCE	Specify convergence criterion
13	**COOR**DINATE	Specify grid coordinates
14	**CORR**ELATION	Computation of correlations between varaibles and locations
15	**DATU**M	Specify reference datum level for hydraulic head
16	**DEBU**G	Specify debug options
17	**DECA**Y	Specify decay constants for chemical species
18	**DEFI**NE	Specify value of a symbolic variable
19	**DENS**ITY	Select fluid mass density options
20	**DIAG**NOSTIC	Diagnostic output options
21	**DIFF**USION	Specify conduction or diffusion constants, coefficients and options
22	**DISA**BLE	Disable certain built-in default options
23	**EBU**	Select Eddy-break-up limiting options for chemical reactions
24	**ELSE**	Define the “ELSE” part of the IF-ELSE-ENDIF construct
25	**END**	End of a problem
26	**ENDIF**	Terminate the IF-ELSE-ENDIF construct
27	**FILE**	Open or close selected I/O device
28	**FIX**	Fixed pressure, temperature or concentration nodes
29	**FLOW**	Specify mass flux entering at boundary
30	**FLUX**	Compute flux of fluid, heat or chemical species crossing a sub-domain
31	**FUEL**	Specify fuel composition and heat of combustion
32	**GAS**	Specify gas constant and parameters

..... Table 6.1.1 continued

TABLE 6.1.1: KEYWORDS OF ANSWER™ AND THEIR FUNCTIONS(continued)

NO.	KEYWORD	INPUT FUNCTION
33	**GEOM**ETRY	Specify or modify problem geometry
34	**GRAV**ITY	Specify constants of gravitational acceleration
35	**GRID**	Number of grid nodes in the x, y and z directions
36	**HIST**ORY	Provide time history output at selected nodes
37	**IF**	To initiate the IF-ELSE-ENDIF construct
38	**INCL**UDE	To include a file at run time in the input data stream
39	**INDE**NT	To allow command input to be indented
40	**INJE**CTION	Specification for injection of Liquid Films and Sprays
41	**INIT**IAL	Initial conditions for state variables
42	**INLE**T	To specify an inflow boundary for the domain of computation
43	**INTE**GRATION	Index for selection of integration profile
44	**LAMI**NAR	Specify laminar flow
45	**LATE**NT	Specify latent heat of fluid
46	**LIMI**T	Set upper and lower limits for dependent variables
47	**LOCA**TE	Specify location of sub-regions or boundaries
48	**MATE**RIAL	Specify material types and properties
49	**MATR**IX	Specify Options for solution of matrix of equations
50	**MOVE**	Specify moving surface within flow domain
51	**NOZZ**LE	Specify liquid fuel nozzle injection parameters
52	**OPEN**	Specify an open boundary through which fluid may enter or leave
53	**OPTI**ON	Modify built-in default options
54	**OUTL**ET	Specify an outflow boundary for the domain of computation
55	**OUTP**UT	Frequency and extent of tabular output
56	**OXID**IZER	Oxygen to nitrogen ratio in the oxidizer stream
57	**PAUS**E	Cause a temporary pause in processing
58	**PERI**ODIC	Specify periodic boundary conditions
59	**PHAS**E	Specify phase change mode and constants
60	**PRAN**DTL	Specify Prandtl numbers
61	**PRIN**T	Print flow rate, sources and statistical measures of variables
62	**PROB**LEM	Specify general nature and type of problem
63	**PROP**ERTY	Option for mode of property specification
64	**QUIT**	Terminate solution process

..... Table 6.1.1 continued

TABLE 6.1.1: KEYWORDS OF ANSWER™ AND THEIR FUNCTIONS(continued)

NO.	KEYWORD	INPUT FUNCTION
65	**RADI**ATION	Activate thermal radiation calculation
66	**REAC**TION	Specify reaction rate constants
67	**READ**	Read initial conditions from archive file
68	**REFE**RENCE	Reference values for key variables
69	**RELA**X	Relaxation factors for governing variables
70	**RENA**ME	Allows renaming of output variables listed in Table 6.8.1
71	**SAVE**	Frequency of output to archive file
72	**SCAL**E	Internal scaling of specified input
73	**SCHM**IDT	Specify Schmidt numbers
74	**SCRE**EN	Echo some of the diagnostic output to screen
75	**SELE**CT	Specify location of sub-regions or boundaries
76	**SET**	Set value of a variable as a function of space and time
77	**SOLI**D	Specify conjugate heat transfer and properties
78	**SOLV**E	Start of solution of equations
79	**SOOT**	Activate soot computations and specify constants
80	**SOUR**CE	Specify source, injection or withdrawal terms
81	**SPEC**IFIC HEAT	Select specific heat options and constants for the fluid
82	**STAC**K	To specify stack transformation operations
83	**STAT**ISTICS	Obtain output of statistical parameters of variables
84	**STOR**ATIVITY	Specify storage coefficient for a governing transport equation
85	**SWIR**L	To specify charactertistics of wwirl imposed on flow
86	**SYMM**ETRY	To identify an external boundary as a symmetry plane or axis
87	**TIME**	Set initial time for simulations
88	**TITL**E	Problem title specification
89	**TRAC**K	Compute Particle tracks and corresponding elapsed time in flow field
90	**TRAN**SFER	Transfer mass and property variables within the computational domain
91	**TURB**ULENT	Select turbulent model and specify constants
92	**USER**	User identification for input and output files
93	**VISC**OSITY	Select fluid viscosity options
94	**WALL**	Specify internal walls within domain of interest
95	**WRIT**E	Generate output of variables

6.2 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 6.2.1 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**[1] 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 6.2.1: FUNCTIONAL UNITS OF ANSWER™ COMMANDS

ORDER	FUNCTION	RELATED KEYWORD COMMAND
1	Identification	**BANNER**, **TITLE**, **USER**
2	Grid specification	**ADAPTIVE**, **CONNECTIVITY**, **COORDINATE**, **GEOMETRY**, **GRID**, **SCALE**
3	Problem definition	**BLOCKAGE**, **DATUM**, **GRAVITY**, **INCLUDE**, **INJECTION**, **INLET**, **LOCATE**, **MATERIAL**, **MOVING**, **NOZZLE**, **OPEN**, **OPTION**, **OUTLET**, **PHASE**, **PROBLEM**, **RENAME**, **SELECT**, **SYMMETRY**, **WALL**
4	Initial and boundary conditions	**ADIABATIC**, **BOUNDARY**, **FIX**, **FLOW**, **INITIAL**, **INLET**, **OPEN**, **OUTLET**, **PERIODIC**, **READ**, **SET**, **SWIRL**, **SYMMETRY**, **TIME**
5	Fluid Properties and constants	**CONDUCTION**, **DENSITY**, **DIFFUSION**, **FUEL**, **GAS**, **LATENT**, **OXIDIZER**, **PRANDTL**, **PROPERTY**, **REFERENCE**, **SCHMIDT**, **SPECIFIC**, **STORAGE**, **VISCOSITY**
6	Nature of Flow	**COMPRESSIBLE**, **LAMINAR**, **RADIATION**, **REACTION**, **SOLID**, **SOOT**, **TURBULENT**
7	Source and sink specifications	**ARRHENIUS**, **EBU**, **DECAY**, **REACTION**, **SOURCE**
8	Solution options	**DISABLE**, **INTEGRATION**, **LIMIT**, **MATRIX**, **RELAX**
9	Output control	**CLOSE**, **CORRELATION**, **DEBUG**, **DIAGNOSTIC**, **FLUX**, **HISTORY**, **PRINT**, **OUTPUT**, **SAVE**, **SCREEN**, **STACK**, **STATISTICS**, **TRACK**, **WRITE**
10	Operational control	**ALLOCATE**, **CONVERGENCE**, **DEFINE**, **ELSE**, **END**, **ENDIF**, **FILE**, **IF**, **INDENT**, **PAUSE**, **QUIT**, **SOLVE**

[1] *The keywords are denoted by bold uppercase letters.*

6.3 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, θ) coordinates. In the 2D mode, the geometry is specified in terms of (x, y) or (x, r) coordinates. Irrespective of the physical coordinates, the grid indices always vary in a uniform manner.

For the structured grids, these vary from (1,1,1) to (IMAX, JMAX, KMAX) for the 3D mode and from (1,1) to (IMAX, JMAX) for the 2D mode. IMAX, JMAX and KMAX are, respectively, the maximum number of grid lines in the x, y (or r) and z (or θ) 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, NMAX, is IMAX * JMAX in 2D and IMAX * JMAX * KMAX in 3D.

For unstructured grids, the grid element numbers vary from 1 to NELEM irrespective of the dimensionality of the problem. In addition a boundary node (linear or planar elements) is automatically inserted at each exterior boundary segment of the computational domain. These boundary nodes greatly facilitate the imposition of complex boundary conditions. In this case, the total number of nodes, NMAX, is the sum of number of elements and the number of exterior boundary segments.

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.

6.4 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 6.4.1. 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. File names for most of these files may be changed by the user. 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 prefix **"acr_"** and the file extension **"TMP".** For example, file unit automatically opened in response to the **HIST**ORY command (Mode 1) is named "**acr_HISTORY.TMP"** unless the user explicitly specifies a file name on the command.

TABLE 6.4.1: 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	xxx_LCNS.ACR	ACRi user License file. The string xxx is replaced by a set of characters that is unique to each user.
Auto	acr_DEBUG.TMP	User specific diagnostic and debug output (**DEBU**G).
Auto	acr_GRID_QUALITY.TMP	Grid quality information (**DEBU**G **GEOM**ETRY).
Auto	acr_MTRXxxx.TMP	Solution Matrix Coefficients (**DEBU**G **MATR**IX). The string xxx is repalced 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 (**FLUX**).
Auto	acr_GRID_XYZ.TMP	Internally generated ACRi format grid file (**GRID PLOT**3D).
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 (**DIAG**NOSTIC).
Auto	acr_HISTORY.TMP	Time-history data file (**HIST**ORY).
Auto	acr_HISTORY_XYZ.TMP	Time-history at specified locations(**HIST**ORY **COOR**DINATE).
Auto	acr_SOURCE.TMP	Time-history of source of variables(**HIST**ORY **SOUR**CE.
Auto	acr_STORAGE.TMP	Time-history of storage of variables(**HIST**ORY **STOR**AGE).
Auto	acr_REDBLACK.TMP	Red/Black matrix split information (**MATR**IX **REDB**LACK).
Auto	acr_NOZZLE.TMP	Nozzle spray trajectory information (**NOZZ**LE **TRAJ**ECTORY).
Auto	acr_PRINT_FORCE.TMP	Time-history of forces or moments on a surface (**PRIN**T **FORC**E).
11	xxx.SAV***	Write data file for restart, archiving and post-processing (**SAVE**).
Auto	xxx_RESTART.TMP***	ACRi restart file (**SAVE REST**ART, **READ REST**ART,).
12	xxx_TABLE.SAV***	Write archive data file in tabular form (**SAVE TABL**E).
Auto	acr_STATS.TMP	Statistics information pertaining to variables (**STAT**ISTICS).
Auto	xxx_TRACK.TMP***	Particle Track information for the flow field (**TRAC**K).
Auto	acr_XYZ_CRNR.TMP	Corner coordinates for each element (**WRIT**E **CORN**ERS).
Auto	acr_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)..**

6.5 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.

6.6 DIAGNOSTIC OUTPUT SPECIFICATION

Four keyword commands control the diagnostic output. These are the **DEBU**G, **DIAG**NOSTIC, **FLUX**, and **OUTP**UT commands. The **DEBU**G command provides a print-out of numerical error parameters which are useful in evaluating the accuracy of numerical solution. The **DIAG**NOSTIC 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 **OUTP**UT command allows the field variables to be written in a tabular format to the standard output device (unit 16).

6.7 DEPENDENT VARIABLES OF ANSWER™

ANSWER™ provides for the numerical solution of an arbitrary number of coupled transport equations. The user may elect to solve any subset of these equations. Table 6.7.1 lists the variables of the governing equations in the order in which they are solved.

TABLE 6.7.1: DEPENDENT VARIABLES OF ANSWER™

ANSWER™ Name	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 θ -direction
PP	ϕ	Density and Pressure correction variable
H	h_s	Stagnation enthalpy (See Note below)
K	k	Kinetic energy of turbulence
E	ε	Dissipation of turbulence energy (See Note below)
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_2-stripped fuel
CO	m_{CO}	Mass fraction of carbon monoxide
H2	m_{H2}	Mass fraction of hydrogen
RA	$(R_x+R_y+R_z)/3$	Net incoming thermal radiation flux.
SN	S_N	Concentration of soot nuclei
S1	S_1	Concentration of small soot particles
S2	S_2	Concentration of large soot particles
User defined	none	User-defined variables with **ALLO**CATE or **RENA**ME commands

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 **SOLV**E command that refers to the solution of the equation or the **SOUR**CE 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.

6.8 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 6.7.1 plus a number of supplementary variables. Table 6.8.1 and 6.8.2 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 6.7.1 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 corneres (IMAX-1 times JMAX-1). For an unstructured grid the vertex coordinate file (**CONN**ECTIVITY 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 **FRREFORM™** commands are provided to obtain output for these variables in different formats. The primary output of the variables can be obtained by the **OUTP**UT, **SAVE** and **WRIT**E commands.

Output for Node Based Variable: Table 6.8.1 lists 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 **OUTP**UT and **SAVE** commands.

Output for Supplimentary Variables: Table 6.8.2 lists some supplementrary 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 **WRIT**E 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 flied, the vorticity, stresses, strains, and some special variables. Please see the **OUTP**UT 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 **OUTP**UT and **SAVE** commands for a detailed description of this output.

TABLE 6.8.1: STANDARD VARIABLES OF ANSWER™

ANSWER™ Name	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 θ-direction
P	P	Fluid pressure
T	T	Fluid Temperature
K	k	Kinetic energy of turbulence
E	ε	Dissipation of turbulence energy
L	ℓ	Length scale of turbulence
H	h_s	Stagnation enthalpy
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_2-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 Nitrogens Oxides
NO2	M_{NO2}	Mass fraction of Nitrogens 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	μ	Effective viscosity of fluid
RHO	ρ	Mass density of fluid
POR	---	Effective porosity of a grid element
PHYD		Hydraulic head of the fluid
EBYK	ε/k	Turbulence time constant
CP	C_p	Specific Heat of fluid mixture
SPEED	V	Modulus of Velocity = $Sqrt(u^2+v^2+w^2)$
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 / sqrt[(γ-1) Cp T]
VOL	δV	Volume of the Element
X	x	X Coordinate or the Node Location
Y	y	Y Coordinate or the Node Location
Z	z	Z Coordinate or the Node Location
User defined	---	User-defined variables with **ALLO**CATE or **RENA**ME commands

TABLE 6.8.2: SUPPLEMENTARY VARIABLES OF ANSWER™

ANSWER™ Name	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

6.9 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 ouput from these commands is written to a file, diagnostic messages appear on the Standard Output Device that identify 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 Table 6.8.1) 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 specifed frequency.

6.10 RESTART OUTPUT

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

6.11 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 6.11.1. In this example, the output requirements for both the archive file and the tabular output are changed after 5 time units of simulation.

TABLE 6.11.1: ILLUSTRATION OF A TWO-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 -1 boundary
OUTLet at +1 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
```

6.12 DIMENSIONING PARAMETER

All ACRi Software Tools employ the FORTRAN PARAMETER statement to set the dimensions of the various arrays. The controlling parameter is named LMAX and its numerical value must be set to equal or exceed the maximum number of grid nodes in the computational domain.

(This page left intentionally blank.)

CHAPTER 7

DESCRIPTION OF KEYWORD COMMANDS

This chapter describes the keyword commands that comprise the user's interface with the ANSWER™ 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 Appendix B of this User's Manual. Suggestions on the preparation of input for ANSWER™ are given in the previous chapters. Descriptive notation for the keyword commands is explained at the beginning of this chapter.

7.1 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 Appendix B.

BOLD	Uppercase characters in bold typeface denote The **FREEFORM™ keywords**. 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.
CAPS	Upper case characters in normal or bold typeface denote **modifiers** of **FREEFORM™** that are *significant* 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 *not significant* 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.
Nn	The n^{th} numeric field in an input command denoted by a keyword.
fname	The name of a file or device; see Section 7.2.2.
subrgn	The subregion for the applicable input; see Section 7.2.3 and 7.2.4.
idsub	A unique identity for a subregion of the domain of computation; see Section 7.2.4.
func	An analytic or tabular function for input of a numerical quantity; see Section 7.2.6.
phase	Fluid phase to which the input applies; see Section 7.2.10.
V_{frq}	Frequency of output; see Section 7.2.11.

7.2 GENERAL INPUT FEATURES FOR FREEFORM™ COMMANDS

7.2.1 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 Appendix B. 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.

7.2.2 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 “**FORM**ATTED” mode. Some commands also allow an “**UNFO**RMATTED” mode of input or output. If allowed, this mode is selected by the “**UNFO**RMATTE**D**” modifier on the command.

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 6.4).

7.2.3 Subregion Specification

Some input commands may specify values for only a portion or subregion of the domain. For these commands, the desired subregion must be specified by a **LOCA**TE or **SELE**CT command. Multiple modes of this command are available to accommodate a diversity of input and output requirements. .

7.2.4 Subregion Identification

A unique identity may be assigned to a subregion by a unique character string with up to four alphanumeric characters, the first of which must be an alphabetic character. The modifier 'ID' in the command must immediately precede the identifier, which is called 'idsub' in this manual. This identifier is then used subsequently on a keyword command to selectively provide input for that subregion. *Each subregion specified as a* **STAT**ION *must be assigned a unique identity.* For other modes of subregion specification, an identity is not always required. If an input command referring to that subregion immediately follows the subregion **LOCA**TE or **SELE**CT command, then the specification of a subregion identity is not necessary. *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. However, the input can be selectively applied to a subregion selected by a modifier on the command, that was previously defined by a **LOCA**TE or **SELE**CT command. The choices available for the **'subrgn'** modifier are summarized in Table 7.2.2.

TABLE 7.2.1: VALID SUBREGION IDENTIFICATION MODIFIERS

subrgn	Interpretation
SELE	The input applies to the most recently defined subregion of the **LOCA**TE or **SELE**CT command. Same as **LOCA**TE modifier.
LOCA	The input applies to the most recently defined subregion of the **LOCA**TE or **SELE**CT command. Same as **SELE**ct modifier.
ID=idsub	The input applies to the subregion that was assigned the identity 'idsub' in a previous **LOCA**TE or **SELE**CT command.

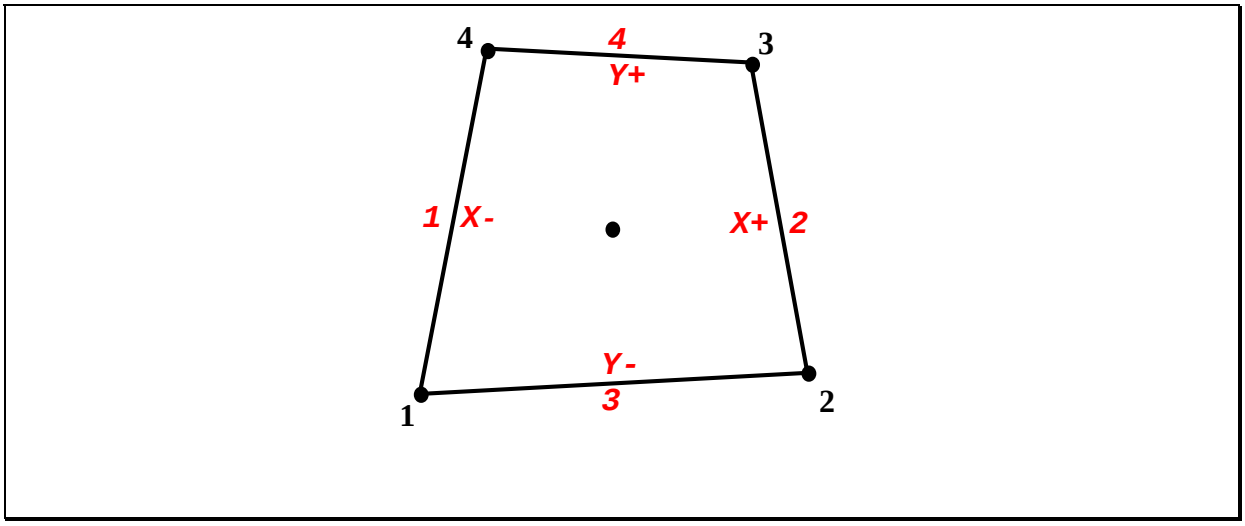
7.2.5 Boundary Identification

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 **LOCA**TE command with a **PAIR**ED list of element and surface numbers

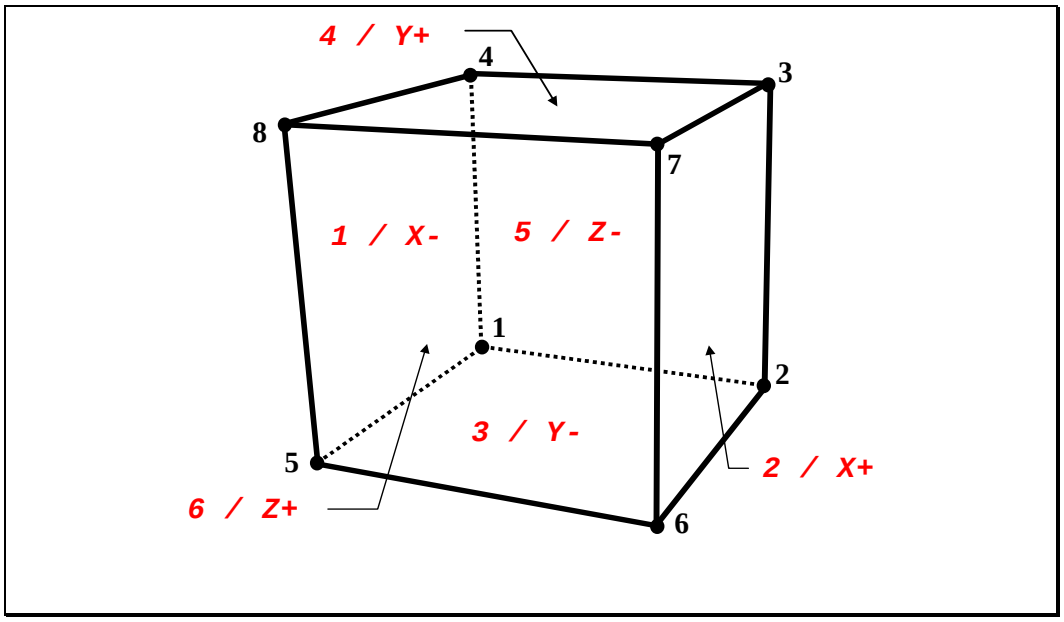
The identification of the boundary orientation by a direction modifier is illustrated in Table 7.2.2. The modifier consists of a character followed by a sign that represents the direction of the outward normal at the boundary. It should be noted that this direction is a nominal direction with respect to the local orientation of the grid element (ξ, η, ζ 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 7.2.1 for both 2D and 3D elements. The vertices are numbered in an anti-clockwise manner in the xy-plane. Briefly, the local x (ξ) vector points from vertex 1 to vertex 2, the local y (η) vector from vertex 1 to vertex 4 and, for 3D elements, the local z (ζ) 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 **LOCA**TE command with **PAIR** modifier). As illustrated in Figure 7.2.1, the surface numbers for 2D elements vary from 1 to 4 and those for 3D elements from 1 to 6.

TABLE 7.2.2: BOUNDARY IDENTIFICATION INDEX

Orientation	Interpretation
X-	The outward normal is in the negative direction of the local x or ξ coordinate. It is equivalent to the specification of surface number 1 of Figure 7.2.1.
X+	The outward normal is in the positive direction of the local x or ξ coordinate. It is equivalent to the specification of surface number 2 of Figure 7.2.1
Y-	The outward normal is in the negative direction of the local y or η coordinate. It is equivalent to the specification of surface number 3 of Figure 7.2.1
Y+	The outward normal is in the positive direction of the local y or η coordinate. It is equivalent to the specification of surface number 4 of Figure 7.2.1
Z-	The outward normal is in the negative direction of the local z or ζ coordinate. It is equivalent to the specification of surface number 5 of Figure 7.2.1.
Z+	The outward normal is in the positive direction of the local z or ζ coordinate. It is equivalent to the specification of surface number 6 of Figure 7.2.1.



Vertex and Face Number Nomenclature for a Quadrilateral Element



Vertex and Face number Nomenclature for a Hexahedral Element

FIGURE 7.2.1: BOUNDARY INDEX NOTATION

7.2.6 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:

$$\Phi = \Phi(\xi) \tag{7.2.1}$$

here Φ 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 Φ is provided. In addition, it is possible to specify user-defined functions as discussed in Section 7.2.9. 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

The most powerful and general option available for this purpose is that of a tabular function in which Φ is specified as an arbitrary function of ψ in a linear, piece-wise approximation:

$$\begin{aligned}
&\Phi = A_1; \quad \xi_1 \le \xi, \\
&\Phi = A_{n-1} + \alpha\ (A_n - A_{n-1}) \frac{\xi - \xi_{n-1}}{\xi_n - \xi_{n-1}}; \quad \xi_{n-1} < \xi < \xi_n; \quad 1 < n \le N, \\
&\Phi = A_N; \quad \xi > \xi_N,
\end{aligned} \tag{7.2.2}$$

where A_n are arbitrary constants and N is the total number of sets in the table. The factor α is an interpolation factor that is set to unity for linear interpolation and zero for step-wise interpolation. For a periodic implementation of the tabular function, the value of ξ used in the above interpolation relation is computed as:

$$\xi_{used} = \mathrm{mod}[\ \xi, (\xi_N - \xi_1)\ ], \tag{7.2.3}$$

A variation of the tabular function option is that of a table of fixed values for a subregion such that for each node of the subregion:

$$\Phi_n = A_n; \quad 1 < n \le N, \tag{7.2.4}$$

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

In addition to the piece-wise tabular options above, follwing analytic functions are currently available:

$$\Phi = A, \tag{7.2.5}$$

$$\Phi = A + B\ \xi, \tag{7.2.6}$$

$$\Phi = A + B\ \xi + C\ \xi^{2} + D\ \xi^{3} + E\ \xi^{4}, \tag{7.2.7}$$

$$\Phi = A\ (\ \xi + B\ )^{C} + D, \tag{7.2.8}$$

$$\Phi = A \sin(\ B\ \xi + C\ ) + D, \tag{7.2.9}$$

$$\Phi = A \cos(\ B\ \xi + C\ ) + D, \tag{7.2.10}$$

$$\Phi = A \tan(\ B\ \xi + C\ ) + D, \tag{7.2.11}$$

$$\Phi = A \arcsin(\ B\ \xi + C\ ) + D, \tag{7.2.12}$$

$$\Phi = A \arccos(\ B\ \xi + C\ ) + D, \tag{7.2.13}$$

$$\Phi = A \arctan(\ B\ \xi + C\ ) + D, \tag{7.2.14}$$

$$\Phi = A \exp(\ B\ \xi + C\ ) + D, \tag{7.2.15}$$

$$\Phi = A \ln(\ B\ \xi + C\ ) + D, \tag{7.2.16}$$

$$\Phi = \frac{(A + B\xi)^{E}}{(C + D\xi)^{F}}, \tag{7.2.17}$$

$$\Phi = A_o + \sum_{n=1}^{N} A_n [\xi + B_n]^{C_n} \tag{7.2.18}$$

$$\Phi = A_o + \sum_{n=1}^{N} A_n \sin[2\pi \frac{\xi + C_n}{B_n}] \tag{7.2.19}$$

$$\Phi = A_o + \sum_{n=1}^{N} A_n \cos[2\pi \frac{\xi + C_n}{B_n}] \tag{7.2.20}$$

$$\Phi = A_o + \sum_{n=1}^{N} A_n \tan[2\pi \frac{\xi + C_n}{B_n}] \tag{7.2.21}$$

$$\Phi = A_o + \sum_{n=1}^{N} A_n \exp[-\frac{\xi + C_n}{B_n}] \tag{7.2.22}$$

$$\Phi = A_o + \sum_{n=1}^{N} A_n \ln[B_n\ \xi + C_n] \tag{7.2.23}$$

In the above equations, A, B, C, D, E and F are constants.

7.2.7 Command Input of Functional Form

The functional form of input can be used with a number of commands. Currently these include the **BOUN**DARY, **COND**UCTION, **DIFF**USION, **DIST**RIBUTION, **FLOW**, **GAS**, **RETA**RDATION, **SET**, **SOUR**CE, **SPEC**IFIC heat and **STOR**AGE commands. These commands is 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 ξ, respectively. Valid input options for **func** and ξ and their corresponding interpretations are given in Tables 7.2.3 and 7.2.4. The 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 7.2.5 through 7.2.8. **By default, the value of the function at a location is computed from the value of the independent variable, ξat the same location. However, if the modifier "STAC" is present on a command then the function is evaluated with the value of ξat the most recent location specified by the STACK command.**

TABLE 7.2.3: VALID INPUT FUNCTIONAL FORMS

User Specification of Function			Dependent Variable Computed From
Function	Option	Sub-option	
TABL			Equation 7.2.2; α= 1
TABL	STEP		Equation 7.2.2; α= 0
TABL		PERI	Equation 7.2.2; α = 1; ξ from Equation 7.2.3
TABL	STEP	PERI	Equation 7.2.2; α = 0; ξ from Equation 7.2.3
NODE			Equation 7.2.4
CONS			Equation 7.2.5
LINE			Equation 7.2.6
POLY			Equation 7.2.7
POWE			Equation 7.2.8
SIN or SINE			Equation 7.2.9
COS or COSI			Equation 7.2.10
TAN or TANG			Equation 7.2.11
ASIN			Equation 7.2.12
ACOS			Equation 7.2.13
ATAN			Equation 7.2.14
EXP or EXPO			Equation 7.2.15
LN or LOG			Equation 7.2.16
RATI			Equation 7.2.17
POWE	SERIes		Equation 7.2.18
SIN or SINE	SERIes		Equation 7.2.19
COS or COSI	SERIes		Equation 7.2.20
TAN or TANG	SERIes		Equation 7.2.21
EXP or EXPO	SERIes		Equation 7.2.22
LN or LOG	SERIes		Equation 7.2.23
USER			User-defined function called for each element separately.
USER	GLOBal		User-defined function called only once for the computational domain.

TABLE 7.2.4: VALID INDEPENDENT VARIABLES

User Specification ξ	Denotes Independent Variable
TIME	t
X	Coordinate x
Y or R	Coordinate y (or r for radial geometry)
Z or THETa	Coordinate z (or θ for radial geometry)
One of the symbols in Table 6.8.1	The corresponding variable listed in Table 6.8.1

TABLE 7.2.5: NUMERICAL INPUT FOR THE TABLE SPECIFICATION

Numerical Value	Interpretation
N1	The number of sets of input in the table that are specified by Equation 7.2.2.
N2, ..., Nn	The pairs of values, ξ_n and θ_n (in that order), for Equation 7.2.2. A total of 2*N1 values (N1 sets) must be specified.

TABLE 7.2.6: NUMERICAL INPUT FOR SIMPLE ANALYTIC FUNCTIONS

Numerical Value	Interpretation
N1, ..., Nn	The constants A_n for Equation 7.2.4 or the constants A through E for the functions defined by Equations 7.2.5 through 7.2.17, as appropriate. The number of values must match those required by the function. For Equation 7.2.4, 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 7.2.7: NUMERICAL INPUT FOR THE SERIES FUNCTIONS

Numerical Values	Interpretation
N1	The number of sets of input in the series specified by Equations 7.2.18 through 7.2.23.
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 7.2.8: 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.

7.2.8 Examples of Functional Form of Input

The Tables 7.2.9 through 7.2.11 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 **SOUR**CE command would specify the source for that variable and, the **BOUN**DARY command may specify the value of the variable or that of the flux of the variable.

TABLE 7.2.9: 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 7.2.10: EXAMPLES OF ANALYTIC FUNCTION SERIES

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

TABLE 7.2.11: EXAMPLES OF TABULAR FUNCTIONS

Illustrative Commands
//// 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)
//// The flux of P at the boundary is a tabular function of time BOUNdary T at X-: FLUX is a TABle in TIME: 3 sets (0., 5), (100., 15), (200, 7.5), h=0.001
//// 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)
//// 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)
//// 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)
//// 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)
//// Table with linear interpolation but input from a file SET T for domain as function of TIME: 20 sets from file 'TIMEVALS'

7.2.9 User-Defined Input Options

Some commands permit input of user-defined functions or options. These options currently are permitted for all the commands listed in Section 7.2.7, and the **DENS**ITY, **MATR**IX, and **VISC**OSITY commands. The user must have a source-code license to exercise these options. This input may be supplied by the user in terms of FORTRAN 77 statements in a reserved module called USRFNC. 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

7.2.10 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 7.2.12. 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 7.2.12: VALID MODIFIERS FOR PHASE DESIGNATION

User Specification of phase	Interpretation For The Corresponding Command
FIRS	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.
SECO	The input is for the second phase of the fluid.
THIR	The input is for the third phase of the fluid. This modifier is available only for the **PORFLOW™** Software Tool.
GAS	The input is for the GAS Phase of the fluid. The gas phase is always the last of the active fluid phases. Thus this modifier is equivalent to the FIRSt modified for a single phase gas, the **SECO**ND modifier for a liquid-gas, and the THIRd modifier for a liquid-liquid-gas simulation.
VAPO	Same as **GAS** Modifier

7.2.11 Specification of Freqeuency 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_{frq} and may be specified in terms of the number of steps or the time interval. The step ineterval 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 modifer **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 **SOLV**E 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.

7.3 DESCRIPTION OF KEYWORD COMMANDS

COMMAND 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.

SYNTAX ADAP {Φ} {N1} [casename]

Φ	**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.

COMMENTS

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.

This command is available only under a consulting agreement with ACRi. Please contact ACRi if you are interested in this feature.

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, initial guesses will be generated in
/ files called 'Level-1.cnc', 'Level-1.blk', 'Level-1.xyz', 'Level-1.loc', 'Level-1.init' etc.
```

COMMAND ADIABATIC

PURPOSE To specify global adiabatic or fixed value boundary conditions at all wall boundaries including internal obstacle walls.

SYNTAX ADIA [Φ] [OFF]

Φ The symbol for the dependent variable for which the adiabatic wall conditions are specified. The valid symbols are listed in Table 6.7.1. **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.

COMMENTS

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 **BOUN**DARY**, INIT**IAL**, READ or SET** commands. The **ADIA**BATIC 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 **BOUN**DARY command. For the species transport variables, the **ADIA**BATIC **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

ADIAbatic conditions	! By default for temperature
ADIAbatic conditions for T at all walls	! Same as above
ADIAbatic OFF for C transport variable	! Change from default to fixed type of walls

COMMAND ALLOCATE

PURPOSE To allocate space for problem-specific input

MODE 1: Allocate Space For A New User Defined Variable

SYNTAX ALLO { name }

name The name of the user defined variable. It may consist of up to 64 alphanumeric characters of which the **1st 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 1st 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 1st 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 1st separator character. A list of separator characters is given in Appendix B.

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.

COMMENTS

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 7.2.6), 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'
```

MODE 2: **Allocate Space for User Input Tables or Material Type**

SYNTAX **ALLO { TABL | MATE | ZONE } {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.

MATE By default memory space is allocated for 100 material types. This command must be used to increase the allocated memory if the user specifies more than 100 material types. This modifier is meaningful only for the **PORFLOW™** Software Tool. It is ignored for other ACRi Software Tools.

ZONE Same as MATE modifier.

N1 The problem-specific memory space (in words) or the number of material types or zones to be allocated.

EXAMPLES

ALLOcate TABLe space for 50000 words of problem specific input
ALLOcate space for 1000 MATErial Types
ALLOcate space for 1000 ZONEs

COMMAND ARRHENIUS

PURPOSE To specify the Arrhenius reaction rate constants for the default 4-step hydrocarbon reaction.

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 6.7.1) for which the Arrhenius reaction rate constants are specified.

N1 The Arrhenius prefactor, C_A (>0), of Equation 3.6.5.

N2 The Arrhenius activation energy constant, C_E (≥0), of Equation 3.6.5.

N3, N4, N5 The exponents a, b, and c of Equation 3.6.5 for the primary, secondary and tertiary reaction species.

TABLE OF DEFAULT VALUES FOR ARRHENIUS COMMAND

symbol	N1	N2	N3	N4	N5
FU	2.0893×10^{22}	2.48×10^{4}	0.50	1.07	0.40
CH	5.0117×10^{19}	2.50×10^{4}	0.90	1.18	-0.37
CO	3.9811×10^{19}	2.00×10^{4}	1.00	0.25	0.50
H2	3.3113×10^{18}	2.05×10^{4}	0.85	1.42	-0.56

EXAMPLES

ARRHenius constants for FU: 3.9E11, 1.5E8, 1.0, 1.0, 0.0

COMMAND **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

COMMAND **BLOCK**

PURPOSE To define a solid object or blockage within the flow domain.

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

SYNTAX **BLOC {subrgn} [FLUI] [EXCL] [FLOO | DEFA]**

subrgn The subregion to be defined as a blocked or solid region. See Sections 7.2.3 and 7.2.4. 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 is 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 of exit from these boundaries. By default, the external boundaries are open for flow.

COMMENTS

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

EXAMPLES

BLOCkage for SELEcted subregion ! Currently active selected subregion
BLOCkage at subregion ID=BLK1 ! Subregion defined with ID=BKL1
BLOCkage with FLOODING option for subregion ID=BLK1
BLOCkage at subregion ID=BLK1 FLUId !Block will be removed later by a BLOCK OFF command

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

SYNTAX **BLOC {subrgn} { MOVE | MOVI } [FLUI] [EXCL]**

subrgn The subregion to be defined as a blocked or solid region. See Sections 7.2.3 and 7.2.4. 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**, **INIT**IAL 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.

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] [RATE= λ] | [FACT= β] [PLAN] [I | J | K =N_{IJK}] [AVER=N_{avg} | OFFS=N_{offset}] [ISOT] [GAMA] [WALL=N_{wall}]

subrgn The subregion to be defined as a blocked or solid region. See Sections 7.2.3 and 7.2.4. 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. The rate of dissolution of the face in touch with fluid is given by:

$$\frac{dr}{dt} = \alpha_4 \Psi^4 + \alpha_3 \Psi^3 + \alpha_2 \Psi^2 + \alpha_1 \Psi + \alpha_0 + \alpha_{-1} \Psi^{-1}$$

Where Ψ is the specific gravity (=ρ/ ρ*) of brine and the superscript * denotes the reference density. This rate is modified for inclined surfaces by a correlation which is given in the Sandia documents. For dissolution rate in units of m/s, the coefficient values are: α_4= 0.003865456, α_3 = -0.019667482, α_2 = 0.039753091, α_1 = -0.039825129, α_0 = 0.019705054, α_{-1} = -0.003827208.

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

Ψ =ρ/ ρ*	1.0	1.05	1.10	1.15	1.20
dr/dt (m/s) x10^6	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**.

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 **λ**.

λ The specified surface dissolution or melting rate if the modifier **RATE** is present.

FACT If this modifier is present, then the rate computed from the built-in correlation is multiplied by the specified factor **β**.

β The specified factor for the dissolution or melting rate if the modifier **FACT** is present.

PLAN The computation of dissolution rate (dr/dt) requires the location of specific gravity,Ψ. By default, Ψ is computed at the fluid element next to the wall between the fluid and the solid (salt). If this modifier is present, Ψ 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 **I, J, K,** and **N_{IJK}** (below). This modifier can only be used for a structured grid.

I The plane is defined by the **I** (x-direction) grid index of a structured grid.

J The plane is defined by the **J** (y-direction) grid index of a structured grid.

K The plane is defined by the **K** (z-direction) grid index of a structured grid.

N_{IJK} The I, J or K location of the plane at which the specific gravity is computed.

AVER 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 N_{IJK} and the depth of averaging specified by N_{avg}

N_{avg} The depth of averaging in the **I, J, K** direction. In the absence of the **PLAN** modifier, the averaging starts from the element located next to the salt wall. If the **PLAN** modifier is present then the averaging starts from the location on the N_{IJK} plane but in line with the wall element. If N_{avg} is positive, then the averaging is in the increasing direction of the coordinate index **I, J, K**. 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 N_{avg} is too large so that it refers to elements beyond the computational domain, then only the points lying on the transact within the computational domain are selected.

OFFS The specific gravity is computed from a location that is offset to the reference location.

N_{offset} The offset in the **I, J, K** direction. In the absence of the **PLAN** modifier, the offset refers to the location that is offset in the **I, J, K** direction from the element located next to the salt wall. If the **PLAN** modifier is present then the offset refers to the location on the N_{IJK} plane but in line with the wall element. If N_{offset} is positive, then the offset is in the increasing direction of the coordinate index **I, J, K**. 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 N_{offset} is too large so that it refers to elements beyond the computational domain, then the last element within the computational domain is selected.

ISOT 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 above correlation without any modification for the angle of inclination.

GAMA 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.

WALL 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 * N23D where N is number of elements in the blocks and N23D 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 if specified, then the amount of memory for wall facets is set to N_{WALL} below.

N_{WALL} The amount of storage allocated for the wall segments of the evolving block if the **WALL** modifier is present. A fatal error occurs if the amount of allocated memory is not sufficient.

APPLICABILITY

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

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 WALL=10000 boundary segments.
BLOCkage SALT subregion ID=BLK1 max ISOTropic and FACTor = 100 times the built-in rate
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 WALL=10000 boundary segments.

MODE 4: **Remove Previously Specified Blocks**

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

subrgn The subregion to be defined as a blocked or solid region. See Sections 7.2.3 and 7.2.4. 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

```
BLOCK     ID=BLK1 OFF                    !Previously specified Block is removed
```

COMMAND BOUNDARY

PURPOSE To specify boundary conditions for the computational domain.

MODE 1: Constant Boundary Conditions

SYNTAX BOUN [Φ] [option] [subopt] [dir] [subrgn] { N1 }, [N2]

Φ A symbol denoting the dependent variable for which the boundary conditions are specified. The valid symbols are listed in Table 6.7.1. If no symbol is specified then the boundary condition is applied to each dependent variable for which a transport equation is solved.

option The type of the boundary condition.

option	INTERPRETATION
VALU	Value of the variable at the boundary node is specified (Equation 3.11.2). This is the default option.
FLUX	Flux of the variable (Equation 3.11.3) at the boundary is specified. The flux is **positive** if it **enters** the computational domain and negative if it leaves it.
GRAD	The normal gradient of the variable (Equation 3.11.3 with Γ = 1) at the boundary is specified. The gradient is taken to be **positive** in the direction of the outward normal at the surface. A positive gradient will result in a positive flux entering the computational domain.
MIXE	A combination of the variable value and its gradient or flux (Equation 3.11.4) at the grid element interface is specified.
FLOW	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.
EXTR	The normal gradient at the boundary is extrapolated from the neighboring values immediately inside the boundary. 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.

subopt The nature of the MIXEd boundary condition.

subopt	INTERPRETATION
FLUX	Mixed boundary condition is specified in terms of the flux of the variable (Equation 3.11.4). This is the default option.
GRAD	Mixed boundary condition is specified in terms of the gradient of the variable (Equation 3.11.4 with Γ = 1).

dir The orientation index for the outward normal at the boundary. See Section 7.2.5 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 Sections 7.2.3 and 7.2.4.

N1 The boundary value of the variable (VALU option, F_o of Equation 3.11.2), the flux (FLUX option, q_F of 3.11.3), the outward normal gradient (GRAD option, $-q_F/\Gamma$ of 3.11.3), or the equilibrium value (MIXE option; F_o of Equation 3.11.4). There is no default value.

N2 The transfer coefficient (h_F of Equation 3.11.4) for the **MIXE**d option. This input is ignored for other options. There is no default value.

COMMENTS

In general, boundary conditions at any boundary segment may be specified only once. Two independent **BOUN**DARY 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 7.2.4 and **LOCA**TE command) may be replaced by a new specification provided the previously specified boundary condition is explicitly disabled by the **BOUN**DARY **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

BOUNdary for P: at X- boundary, value = 0
BOUNdary GRADient at X+ boundary for all variables is = 0.
BOUNdary for T: orientation index X+ value = 10.
BOUNdary for T at X+: value = 10 at ID=UPPEr subregion
BOUNdary for T at Y- boundary: FLUX = 10 at the selected segment
BOUNdary for P at Y- boundary: GRAD = -2.5 at the selected segment
BOUNdary for T at Y+ boundary: MIXEd type: equilibrium value=5 h=0.5
BOUNdary T: Y+; MIXEd FLUX: v=1, h=0.1 for segment with ID=MIDDle
BOUNdary T: Y+; MIXEd GRAD: v=-1, h=0.1 for SELEcted subregion

MODE 2: **Boundary Condition as a Function of Another Variable**

SYNTAX **BOUN [Φ] [func [ξ]] [option] [subopt] [dir] [subrgn] { fname | N1 ..., Nn } [Nn+1]**

Φ See Mode 1 specification.

func One of the modifiers listed in Table 7.2.3 that denotes the functional form of the input. The function specifies the value of F_0 of Equation 3.11.2 or 3.11.4 if the **VALU**e, or **MIXE**d option is selected, and it specifies the flux, q_F of Equation 3.11.3, if the **FLUX** or **GRAD** options are selected. If no function is specified then the value is assumed to be constant.

ξ One of the independent variables listed in Table 7.2.4. If no variable is specified, then the independent variable is assumed to be time.

option See Mode 1 specification.

subopt See Mode 1 specification.

dir See Mode 1 specification.

subrgn See Mode 1 specification.

fname The name of the file containing numerical values **N1** through **Nn**. This option can be used only if the selected function is a table or one of the series functions. See Section 7.2.2 for additional information.

N1, .., Nn The numerical constants and coefficients for the selected function. See Section 7.2.7 for more details. There are no default values for this input.

Nn+1 The transfer coefficient (h_F of Equation 3.11.4). In the **FLUX** mode, the input units are those of velocity, heat or mass flux divided by the units of the dependent variable, Φ. In the **GRAD**ient mode, the input units are those of Φ divided by those of distance. The default value is 0. In the presence of the **fname** modifier, the input value must be the 2nd numerical value in the command because all functional input is obtained from the file.

EXAMPLES

Generic examples for this command are given in Section 7.2.8. The command keyword (**BOUN**DARY) 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
BOUNdary value at X+ boundary LINEar function: T = 30. -0.015 * Y
BOUNdary for T: at X+ boundary FLUX given by USER function #7
BOUNdary value for P at Y-  COSIne SERIes in T:  3 sets: (0., 5, 1), (100, 10, 1 ), (200, 20, 3) datum = 10.
BOUNdary P GRADient: Y- boundary: TABLe of values: 3 sets (TIME, value)
          (0., 0.01), (100., 0.10 ), (200, -0.20)
BOUNdary P at X+:  TABle of 32 sets (TIME, value) from file='BVALUES'
////          Examples with transfer coefficient for MIXEd type
BOUNdary 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
BOUNdary 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
BOUNdary T at X-: MIXEd GRADient TABle: 3 sets:  (0., 5), (100., 15 ), (200, 7.5),   h=0.001
BOUNdary MIXEd GRAD P at Y+: SINE SERIes 24 sets  'BMIXEDV' h=0.01
```

MODE 3: **Total Pressure Boundary Condition**

SYNTAX **BOUN { TOTA } {P} [func [ξ]] [dir] [subrgn] { fname | N1 ..., Nn } [Nn+1,.. Nn+m]**

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:

$$P_{Total} = P + \frac{1}{2}\rho\left(u^2 + v^2 + w^2\right),$$ where P_{Total}, is the total pressure, P is the local pressure, ρ is the density, and u, v, 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 7.2.3 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 7.2.4. If no variable is specified, then the independent variable is assumed to be time.

dir See Mode 1 specification.

subrgn See Mode 1 specification.

fname The name of the file containing numerical values N1 through Nn. This option can be used only if the selected function is a table or one of the series functions. See Section 7.2.2 for additional information.

N1, .., Nn The numerical constants and coefficients for the selected function. See Section 7.2.7 for more details. There are no default values for this input.

Nn+1,..Nn+m 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.

APPLICABILTY

This mode of the command is only available with the **ANSWER™** Software Tool.

EXAMPLES

Generic examples for this command are given in Section 7.2.8. The command keyword (**BOUN**DARY) 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
BOUNdary TOTAL P at X+ boundary = 2.E.5
BOUNdary TOTAL P at X- bndry = 1.E6; velocity components (0.5, 0.5, 0.) ! flow at 45 deg to x-axis
BOUNdary TOTAL P X-: POLYnomial in Y: (10., 0.5, 0., -0.1, 0.)
BOUNdary 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: Gate boundary conditions for tide height

SYNTAX **BOUN {ETA} {GATE} {IN | OUT} [func [ξ]] [dir] [subrgn] { fname | N1 ..., Nn }**

ETA The boundary condition is for water elevation or tide height for **TIDAL™** Software Tool.

GATE 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 **IN** and **OUT** below).

IN The gate is assumed to open inwards. When the specified water elevation (η) exceeds the elevation just inside the boundary, the flow is allowed to go in; otherwise the normal velocity at the boundary (Vn) 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:

$\eta_S > \eta_F \rightarrow \eta_B = \eta_S ; \quad Vn_B = Vn_F$
$\eta_S \leq \eta_F \rightarrow \eta_B = \eta_F ; \quad Vn_B = 0$

OUT The gate is assumed to open outwards. When the elevation just inside the boundary exceeds the specified water elevation (η), the flow is allowed to go out; otherwise the normal velocity at the boundary (Vn) is assumed to be zero. The imposed boundary conditions are:

$\eta_F > \eta_S \rightarrow \eta_B = \eta_S ; \quad Vn_B = Vn_F$
$\eta_F \leq \eta_S \rightarrow \eta_B = \eta_F ; \quad Vn_B = 0$

func One of the modifiers listed in Table 7.2.3 that denotes the functional form of the input. The function specifies the value of F_0 of Equation 3.11.2 since this option can be used only to specify the **VALU**e of elevation,η, at the boundary. If no function is specified then the value is assumed to be constant.

ξ One of the independent variables listed in Table 7.2.4. If no variable is specified, then the independent variable is assumed to be time.

dir See Mode 1 specification.

subrgn See Mode 1 specification.

fname The name of the file containing numerical values **N1** through **Nn**. This option can be used only if the selected function is a table or one of the series functions. See Section 7.2.2 for additional information.

N1, .., Nn The numerical constants and coefficients for the selected function. See Section 7.2.7 for more details. There are no default values for this input.

APPLICABILTY

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

EXAMPLES

Generic examples for this command are given in Section 7.2.8. The command keyword (**BOUN**DARY) 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 5: **Disable Previously Specified Boundary Conditions**

SYNTAX **BOUN {Φ} {OFF} {dir} {ID=idsub}**

Φ See Mode 1 specification.

OFF Previously specified boundary conditions for Φ 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 **LOCA**TE or **SELE**CT command.

EXAMPLES

BOUNdary for T at Y+ boundary for ID=UPPEr turned OFF

COMMAND CLOSE

PURPOSE To close output devices.

SYNTAX CLOS [fname] [filetype | NUNIT]

fname The name of the file to be closed. See Section 7.2.2 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.

filetype **One** of the modifiers below that identify the file to be opened or closed.

filetype	Interpretation
DEBU	Debug output file (DEBUG) is selected for the operation.
FLUX	Flux output file (FLUX) is selected for the operation.
HIST	Time History file (HISTORY) is selected for the operation.
SAVE	Archive file (SAVE) is selected for the operation.
TABL	Tabulated archive file (SAVE TABLE) is selected for the operation.
TRAC	Particle track file (TRACK) 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.**

COMMENTS

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 particlle TRAC file immediately
CLOSe file by name 'MYOLDFILE'

COMMAND **COMPRESSIBLE**

PURPOSE To select compressible flow option.

SYNTAX **COMP [MILD | DENS | SIMP | SYMM | DIRE | COUP] [ISEN | ISOT] [ISOP] [INVI] [FIXE] [RHOU] [N1]**

MILD By default, the **COMP**RESSIBLE 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 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.

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.

N1 The constant C (>0) of the pressure perturbation equation (Equation 2.3.11). The default value is 1.4.

COMMENTS

For the compressible mode of solution, the enthalpy equation is modified to solve for stagnation enthalpy rather than enthalpy. That is the kinetic energy of fluid is added to the enthalpy defined in Equation 2.2.3.

EXAMPLES

COMPressible flow
COMPressible flow with C=1.0
COMPressible flow with MILD compressibility
COMPressible ISENthalpic flow
COMPressible flow with INVIscid and ISOPressure assumptions
COMPressible flow with INVIscid and ISOPressure assumptions; use FIXED pressure option

COMMAND **CONDUCTIVITY**

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

MODE 1: **Constant Conductivity or Diffusivity**

SYNTAX **COND {Φ} { Γ_Φ } [subrgn]**

Φ A symbol that denotes the dependent variable for which the conductivity or diffusivity is specified. Valid symbols are listed in Table 6.7.1. There is no default value; a symbol must be specified.

Γ_Φ The numerical value (>0) for tconductivity or diffusivity. There is no default value; a value must be specified.

subrgn The subregion for which the input is specified. See Sections 7.2.3 and 7.2.4. If no subregion is specified, the entire computational domain is selected.

EXAMPLES

CONDuctivity for T =1.E-6	! Thermal Condcutivity for the temperature equation
CONDuctivity for T =1.E-6 for subdomain ID=BLK1	! Thermal Condcutivity for the temperature equation
CONDuctivity for C =1.E-6	! Diffusivity for the speciies transport equation

MODE 2: Functional Form of Conductivity or Diffusivity

SYNTAX COND {Φ} {func [ξ]} [subrgn] [dir] {fname | N1 ..., Nn}

Φ A symbol that denotes the dependent variable for which the conductivity or diffusivity is specified. Valid symbols are listed in Table 6.7.1. There is no default value; a symbol must be specified.

func One of the modifiers listed in Table 7.2.3 that denotes the functional form of the input. For this input, the function specifies the value of the appropriate conduction or diffusion 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 7.2.4. If no variable is specified, then the independent variable is assumed to be time.

subrgn The subregion for which the input is specified. See Sections 7.2.3 and 7.2.4. If no subregion is specified, the entire computational domain is selected.

dir 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 diffusion 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; their specification automatically and simultaneously invokes Mode 5 of the command.** If no direction is specified, then the input is applied to all components of the tensor.

fname The name of the file containing the numerical values **N1** through **Nn**. This option can be used only if the selected function is a table or one of the series functions. See Section 7.2.2 for additional information.

N1, .., Nn The numerical constants and coefficients for the selected function. See Section 7.2.7 for more details. There are no default values for this input.

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 4) 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.

EXAMPLES

Generic examples for this command are given in Section 7.2.8. 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.

MODE 3: **Specification of Contact or Film Transfer Coefficient**

SYNTAX **COND {CONT|FILM} [REPL|ADD|TOTA] {Φ} {func [ξ]} [subrgn] [dir] {fname|N1 .. Nn}**

CONT 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.

FILM Same as **CONT**ACT Modifier.

REPL This is the default mode. Any internal computations for the conductive (diffusive) flux between two nodes across a surface are replaced by:

$$q_\Phi = \frac{A_f\left(\Phi_1 - \Phi_2\right)}{\frac{ds_1}{\Gamma_1} + \frac{ds_2}{\Gamma_2} + \frac{1}{h}}$$

where, q_Φ is flux from node "1" to node "2", A_f is the interface area, ds1 and ds2 are distances, Γ is the diffusion coefficient and, h is the specified transfer coefficient.

ADD 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:

$$q_\Phi = q_{\Phi_{internal}} + h A_f\left(\Phi_1 - \Phi_2\right)$$

TOTA The diffusive flux at the surface is computed as:

$$q_\Phi = h A_f\left(\Phi_1 - \Phi_2\right)$$

Φ A symbol that denotes the dependent variable for which the input is specified. Valid symbols are listed in Table 6.7.1. There is no default value; a symbol must be specified.

func One of the modifiers listed in Table 7.2.3 that denotes the functional form of the input. The function specifies the value of the appropriate transfer coefficient for the specified Φ variable. If no function is specified then the value is assumed to be constant.

ξ One of the independent variables listed in Table 7.2.4. If no variable is specified, then the independent variable is assumed to be time.

subrgn The subregion for which the input is specified. See Sections 7.2.3 and 7.2.4. If no subregion is specified, the entire computational domain is selected.

dir The orientation index for the element boundary associated with the transfer coefficient. See Section 7.2.5 for available choices. If no **dir** is specified, then all surfaces of the selected **subrgn** are selected.

fname The name of the file containing the numerical values **N1** through **Nn**. This option can be used only if the selected function is a table or one of the series functions. See Section 7.2.2 for additional information.

N1, .., Nn The numerical constants and coefficients for the selected function. See Section 7.2.7 for more details. There are no default values for this input.

APPLICABILITY

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

EXAMPLES

CONDuction for T with contact heat transfer coefficient of 0.01 at Y+ of ID=WALL
CONDuction for T with contact heat transfer coefficient of 0.01 at Y+ of ID=WALL TOTAl

Generic examples for this command are given in Section 7.2.8. The command keyword (COND) must replace the keyword used in these examples. The modifier FILM or CONTACT **must appear in the command.**

MODE 4: **Modification of Diffusion Coefficient by Richardson Number**

SYNTAX **COND {RICH } {dir } [UNLI | ZERO] [α, β, α_Φ , β_Φ]**

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, μ_j, and the diffusion coefficient, Γ_j, in the selected direction are computed from:

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

Where subscript "j" denotes the direction specified by the **dir** modifier, μ and Γ are, respectively, the scalar values of the viscosity and diffusion coefficients (specified by user or computed from built-in formulae), and α, β, α_Φ and β_Φ 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^{th} direction (specified by the dir modifier), ρ is the fluid density and u_j represents the velocity components in directions other than the i^{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 **UNLI**MITED 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 **LIMI**T 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_Φ , β_Φ 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

CONDuction modified by RICHardson Number
CONDuction modified by RICHardson Number with constants: 0.67, 6.67, 1., 5

MODE 5: **Treatment of Skew Diffusion Terms**

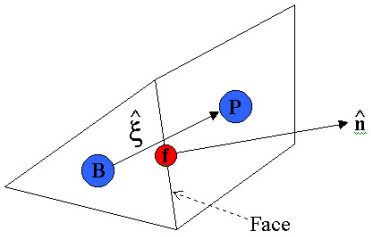
SYNTAX **COND {SKEW | SECO } [POSI] [Φ]**

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 2nd 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.

POSI The total diffusive flux for the variable Φ at the face "f" between two elements shown in the picture below is given by:

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



where summation convention is implied, and Γ_{ij}^f is the diffusion coefficient for the variable Φ at face f, A_f is the area of the face and n_i is the direction vector in the i^{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:

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

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

For highly skewed grids, above relation may be violated. If the modifier **POSI** is specified, then the diffusive flux Q_D is computed only from the values of Φ_B and Φ_P and any contribution of the 2nd order skew terms is ignored.

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 6.7.1. 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 POSItive relation between flux and node values.

MODE 6: **Tensor Diffusivity with off-Diagonal Terms**

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

TENS By default only the diagonal components of the diffusion tensor Γ_{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):

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

$$\underline{V} = (v_i v_i)^{1/2},$$

where α_L and α_T are, respectively, the longitudinal and transverse dispersivities for the porous matrix, δ_{ij} is the Kronecker delta function and, $\underline{\mathbf{V}}$ is the magnitude of the local pore or fluid particle velocity vector, v_i. The particle velocity vector is, in turn, related to the Darcy velocity vector, V_i, and the soil moisture, θ_E, by:

$$v_i = V_i / \theta_E.$$

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.

POSI See Mode 4 of the command.

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 6.7.1. 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 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 7: **Treatment of Conductivity and Diffusivity at External Boundary**

SYNTAX **COND {BOUN} [Φ]**

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 **BOUN**DARY 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.

Φ **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 6.7.1. 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 8: **Elimination of Diffusion Terms**

SYNTAX **COND {OFF} [Φ]**

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

Φ A symbol that denotes the dependent variable for which the diffusion terms are to be eliminated. The valid symbols are listed in Table 6.7.1. There is no default value; a symbol must be specified.

EXAMPLES

CONDuction OFF for the temperature variable: T

COMMAND CONNECTIVITY

PURPOSE To specify element to vertex connectivity for unstructured grid.

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 **(ξ, η)** 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 **(ξ, η, ζ)** 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 **(ξ, η, ζ)** direction for each element is defined by the order in which the vertices appear on this record. The local ξ axis is oriented from vertex 1 to vertex 2, the **η** axis from vertex 1 to vertex 4, and the ζ 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 7.2.5. 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.

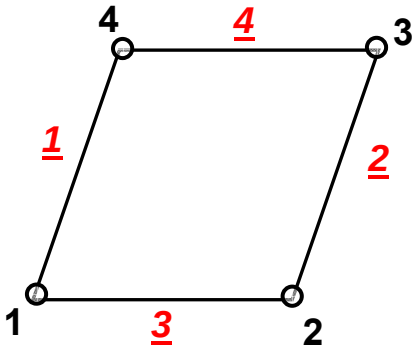
COMMENTS

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

CONNectivity information on file "VERT2ELM.CNC"
CONNectivity of VERTices to elements on file "VERT2ELM.CNC"



Quadrilateral

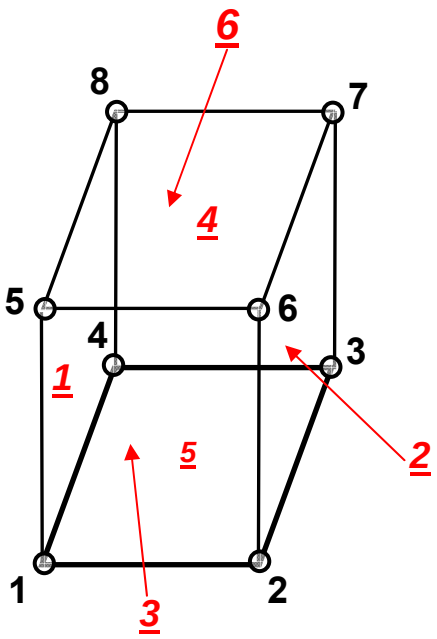
Assume that the vertex connectivity record for this element reads as follows:

Cell #	Cell Type	# of Corners,	Vertex List:
?	1	4	1, 2, 3, 4

Side Number	Vertex Numbers
1	1, 4
2	2, 3
3	1, 2
4	3, 4

The right hand rule implies that: $(\vec{r}_{12} \times \vec{r}_{13}) \bullet \vec{k} > 0$, where $\vec{r}_{12}$ is the position vector from vertex 1 to vertex 2.

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: $(\vec{r}_{12} \times \vec{r}_{14}) \bullet \vec{r}_{15} > 0$, 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)
```

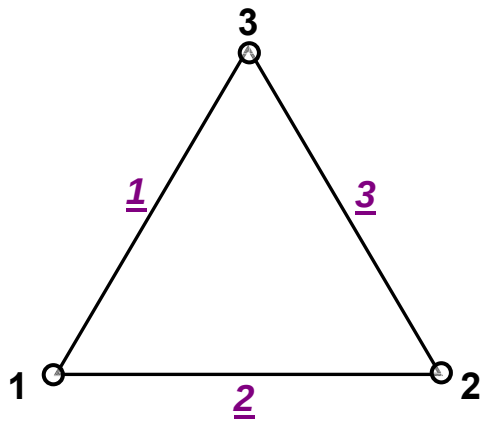
where IFILE is an internally assigned file unit number, M, is the element number, MTYPE is the element type, NVRTX is the number of vertices for the element, and NV are the vertex numbers for the element corners.

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 7.2.5).

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”



Triangle

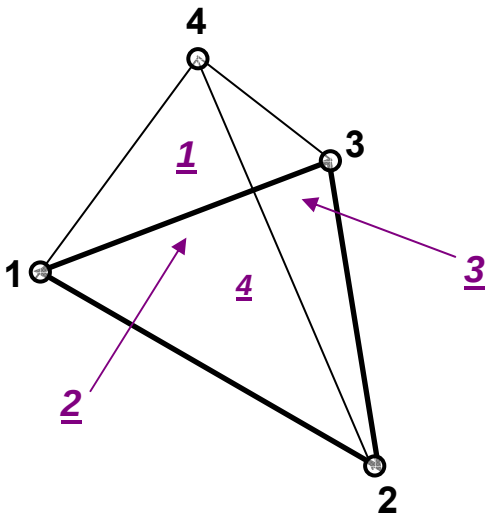
Assume that the vertex connectivity record for this element reads as follows:

Cell #,	Cell Type,	# of Corners,	Vertex List:
?	1	3	1, 2, 3

Side Number	*Vertex Numbers*
1	1, 2
2	2, 3
3	3, 1

The right hand rule implies that: $(\overrightarrow{r_{12}} \times \overrightarrow{r_{13}}) \bullet \vec{k} > 0$, where $\overrightarrow{r_{12}}$ is the position vector from vertex 1 to vertex 2.

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



Tetrahedron

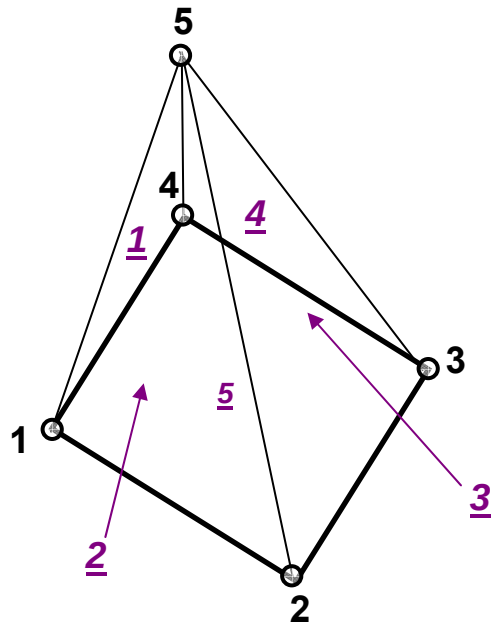
Assume that the vertex connectivity record for this element reads as follows:

Cell #,	Cell Type,	# of Corners,	Vertex List:
?	3	4	1, 2, 3, 4

Side Number	Vertex Numbers
1	1, 3, 4
2	1, 4, 2
3	4, 3, 2
4	1, 2, 3

The right hand rule implies that: $(\vec{r}_{12} \times \vec{r}_{13}) \bullet \vec{r}_{14} > 0$, where $\vec{r}_{12}$ is the position vector from vertex 1 to vertex 2.

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



Pyramid

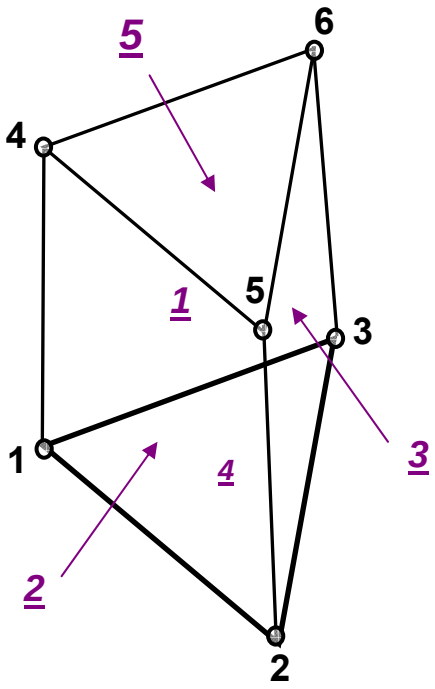
Assume that the vertex connectivity record for this element reads as follows:

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

Side Number	Vertex Numbers
1	1, 4, 5
2	1, 5, 2
3	3, 2, 5
4	4, 3, 5
5	1, 2, 3, 4

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

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



Triangular Prism

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

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

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

The right hand rule implies that: $(\vec{r}_{12} \times \vec{r}_{13}) \bullet \vec{r}_{14} > 0$, where $\vec{r}_{12}$ is the position vector from vertex 1 to vertex 2.

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.

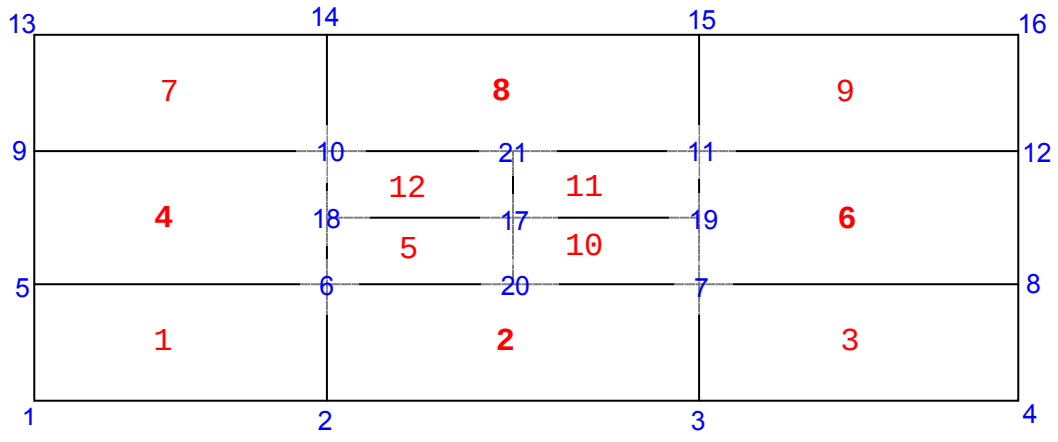
The 1st record in the file consists of a header with two numbers: the number of split elements and the total number of data items in the rest of the file. The header is followed by a number of sets of data equal to the number of split elements. The 1st record of each set consists of the element number 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)
```

COMMAND **CONVERGENCE**

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

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

SYNTAX **CONV [Φ] [phase] [REFE] [GLOB | LOCA | DIFF | NORM] [OLD | NEW] [N1, N2, N3, N4]**

Φ A symbol that denotes the dependent variable for which input is specified. The valid symbols are listed in Table 6.7.1. 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 7.2.10 for available options. By default the input pertains to the 1st phase of the fluid. This modifier is currently available only for the **PORFLOW™** Software Tool.

REFE The specified variable is used as a reference variable for monitoring the convergence of the solution process. Please see Mode 2 of this command for an expanded mode of this input.

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

$$\varepsilon = \frac{1}{N}\sum_{i=1}^{N} \left| A_{ij} X_j - B_i \right|$$

$$\varepsilon = \frac{1}{N}\sum_{j=1}^{N} \left| X_j^{new} - X_j^{old} \right|$$

$$\varepsilon = \frac{1}{N}\sum_{j=1}^{N} \left| \frac{X_j^{new} - X_j^{old}}{X_j^{old}} \right| ; \quad \left| X_j^{old} \right| \geq X_{ref}$$

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 1st 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:

$$\varepsilon = \max_i \left| A_{ij} X_j - B_i \right|$$

$$\varepsilon = \max_j \left| X_j^{new} - X_j^{old} \right|$$

$$\varepsilon = \max_j \left| \frac{X_j^{new} - X_j^{old}}{X_j^{old}} \right| ; \quad \left| X_j^{old} \right| \geq X_{ref}$$

By default the first equation is used unless the DIFF modifier is present.

DIFF Convergence is judged on the basis of the 2nd or 3rd equation above based on whether the GLOB or LOCA mode of the command is selected. In each case, the 2nd 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 3rd 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.

N1 The ε of the residual equations. The default value is 1.E-6.

N2 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.

N3 Minimum number of iterations for the specified variable. The default value is 1.

N4 The X_{ref} of the normalized residual equations above. The default value is 1.E-7.

COMMENTS

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 **N2** 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 (Φ) 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

CONVergence tolerance = 1.E-6
CONVergence criterion = 1.E-6 for SECOnd phase of fluid
CONVergence for U in LOCAl mode: value = 1.E-4
CONVergence for U in DIFFerence mode: value = 1.E-3
CONVergence for T as REFErence variable in GLOBal mode: value = 1.E-4
CONVergence 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;
CONVergence 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] [N1, N2]**

REFE The specified variable is used as a reference variable for monitoring the convergence of the solution process. The default options and values may depend on your installation. For most installations the default is set as:

(1) The MOMEntum option if the velocity and mass balance equations are solved.
(2) The 1st 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 the 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 6.7.1.

N1 The convergence tolerance ε for the mass balance equation. The default value is 1.E-6.

N2 Maximum number of iterations for convergence for the outer loop which consists of the total set of active momentum (velocity) and mass balance equations. The default value is 1.

COMMENTS

The default options and values may depend on your installation. The defaults may be configured differently based on installation-specific optimizations.

EXAMPLES

CONVergence REFErence based on FLOW with Tolerance = 1.E-6
CONVergence REFErence FLOW with Tolerance = 1.E-6 with 10 outer loop iterations
CONVergence REFErence based on MOMEntum with flow Tolerance = 1.E-5
CONVergence REFErence based on ALL equations with flow tolerance =1.E-6
CONVergence REFErence based on ALL equations with flow tolerance =1.E-6 and 5 outer iterations

MODE 3: **Coupled Convergence Test for Species Transport Equations**

SYNTAX **CONV { COUP} { SPEC} [N1]**

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 **COUP**LED modifier is present then an iterative loop is set up over the complete set of species equations. The equations are solved up to **N1** 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 **COUP**LED modifier for this mode of the command to be effective.

N1 Maximum number of iterations for convergence. If a value less than 2 is specified, then **N1** is set to 2. The default value is 2.

EXAMPLES

CONVergence for SPECies in COUPLEd mode
CONVergence for SPECies in COUPLEd mode with a maximum of 10 iterations

MODE 4: **Convergence Threshold for Termination of Solution Process**

SYNTAX **CONV { TERM } [N1]**

TERM By default, the solution process is terminated if the selected convergence index (**CONV**ERGENCE **REFE**RENCE 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 **N1**.

N1 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.

COMMAND **COORDINATE**

PURPOSE To specify the grid locations for Cartesian (x, y, z) or cylindrical (x, y, θ) coordinates.

MODE 1: **Structured Grid: Coordinate Range**

SYNTAX **COOR {RANG} {dir} [CYLI] [DEGR] [NODE] {N1} [N2]**

RANG 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.

dir One of the **X, Y, Z, R** or **THETa** modifiers that, respectively, denotes the x, y, z, r or θ coordinates for which input is specified.

CYLI By default the coordinate system is assumed to be Cartesian. If this modifier is present, then a cylindrical coordinate system is selected.

DEGR By default, the angular input for θ is assumed to be in radian. If the modifier **DEGR**ee is present, then the input values are in degrees.

NODE 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 total range is divided such that the grid nodes are placed at appropriate spacing and the element interfaces are positioned at the middle of the grid nodes.

N1 The desired range for the computational domain length in the specified direction for an orthogonal grid. There is no default value. A value must be specified.

N2 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: 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] {N1} [N2, N3]**

MINI The minimum and maximum values are specified. The coordinates are calculated internally, with a geometric ratio between the successive grid intervals.

MAXI Same as MINI modifier.

dir See Mode 1 specification.

CYLI See Mode 1 specification.

DEGR See Mode 1 specification.

NODE See Mode 1 specification.

N1 The 1st coordinate value. There is no default value; a value must be specified.

N2 The last coordinate value. There is no default value; a value must be specified.

N3 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. DEGrees for the NODEs

MODE 3: **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

COORdinate corners are: (0., 0.) (1.,0.), (0., 1.) and (1., 1.) !! Unit Square
COORdinate: (0,0) (0.707,0.707), (-0.707,0.707) and (0,1.414) !45 deg Square
COORdinate: (0,0) (1.,0.), (0.,1.) and (1,1) CYLIndrical system
COORdinates: 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
COORdinates: 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 4: **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 1st through Nth 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

COORdinate X: 0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100 !!! element interfaces
COORdinate Y: 0, 5, 15, 25, 35, 45, 55, 65, 75, 85, 95, 100 at NODES
COORdinate R: 0, 5, 15, 25, 35, 45, 55, 65, 75, 85, 95, 100 at NODES
COORdinate Y: 0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100 CYLIndrical mode
COORdinate THETa: 0, 5, 15, 25, 35, 45, 60, 75, 90, 105, 120 DEGRees

MODE 5: **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 7.2.2 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 θ 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 1st 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}, ...x_{n1}; x_{12}, x_{22}, x_{32}, ...,x_{n2}; ... 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}), ... (x_{n1}, y_{n1}); (x_{12}, y_{12}), (x_{22}, y_{22}), (x_{32}, y_{32}), ... (x_{n2}, y_{n2}); ... 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+1st record. The default value is 0.

COMMENTS

If this command is used for cylindrical geometry in unstructured mode, then the vertices for elements must be defined such that the local element (ξ, η, ζ) coordinates are identical with the global (X,R,θ) 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 7.2.5 and the **CONN**ECTIVITY command.

EXAMPLES

COORdinate X from file 'XGRID'
COORdinate X, Y and Z from file 'XYZGRID'
COORdinate X, Y and Z from file 'XYZGRID' in BLOCk mode; ignore 7 records
COORdinate X, Y and Z from file 'XYZGRID' in BLOCk IJK mode; ignore 7 records
COORdinate X, Y from file 'XRGRID' in CYLIndrical mode
COORdinate X, R, THETA (DEGRees) from file 'XRTHETA'
COORdinate X, Y and Z from file 'XYZ' in JIK mode for grid NODEs

MODE 6: **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 7.2.2 for additional information.

COMMENTS

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

COORdinates of VERTices on file 'PROBGRID'

COMMAND CPU

PURPOSE To specify the number of Central Processing Units for parallel processing.

SYNTAX CPU { N_{CPU} }

N_{CPU} The number of CPU's for parallel processing. The default value is 1.

EXAMPLES

CPU 4 CPU's for this computer system

COMMAND 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_{frq}] [OFF]

Φ A symbol that denotes the variable for which correlation is desired. Valid symbols are listed in Table 6.7.1. If Φ denotes the instant value of a variable at a given location, Φ^n its value at n^{th} time (or iterative) step, then its 1^{st} and 2^{nd} 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:

$$R_{11} = \langle \Phi^2 \rangle - \langle \Phi \rangle \langle \Phi \rangle$$

$$C_{11} = \frac{R11}{\langle \Phi \rangle \langle \Phi \rangle}$$

$$D_{11} = \Phi - \langle \Phi \rangle$$

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. The total number of open files in any simulation can not exceed 64.

TIME By default, V_{frq} 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_{frq} specification.

V_{frq} 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 **SOLV**E 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 **CORR**ELATION 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 { Φ_1, Φ_2} [subrgn] [fname] [TIME] [NOW] [V_{frq}] [OFF]**

Φ_1, Φ_2 **Two** symbols that denote the variables for which correlation is desired. Valid symbols are listed in Table 6.7.1. If Φ_k denotes the instant value of the k^{th} variable at a given location, Φ_k^n its value at n^{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:

$$R_{12} = \langle \Phi_1 \Phi_2 \rangle - \langle \Phi_1 \rangle \langle \Phi_2 \rangle$$

$$C_{12} = \frac{R12}{\sqrt{\langle \Phi_1^2 \rangle - \langle \Phi_1 \rangle^2}\ \sqrt{\langle \Phi_2^2 \rangle - \langle \Phi_2 \rangle^2}}$$

$$D_{12} = (\Phi_1 - \langle \Phi_1 \rangle)(\Phi_2 - \langle \Phi_2 \rangle)$$

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_{frq}** 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_{frq}** specification.

V_{frq} 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 **SOLV**E 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 **CORR**ELATION commands for the specified variable and subregion are disabled. New commands may be subsequently specified.

EXAMPLES

CORRelation for variable U and V for the SELEcted subregion at the end of simulations
CORRelation for variable U and T for the ID=LOC_AUTO every 25 steps
CORRelation OFF for variable U and T for the ID=LOC_AUTO

MODE 3: **Two Point Correlation for One or Two Variables**

SYNTAX **CORR { TWO } { Φ_1 } [Φ_2] [subrgn] [fname] [TIME] [NOW] [V_{frq}] [OFF]**

TWO **Two point correlation** with matched sets of pairs of elements is specified. The subregion for this option must be previously specified with a **LOCATE CORR**ELATION command.

Φ_1, Φ_2 **One or Two** symbols that denote the variables for which correlation is desired. Valid symbols are listed in Table 6.7.1. The correlation formulae are identical to the two-variable cross-correlation described in Mode 2 except for the definition **Φ_1** and **Φ_2.** The **Φ_1** is always taken to be the values of the 1st specified variable at the elements of the 1st set of paired elements. If **only one** variable is specified then **Φ_2** is taken to be the values at the 2nd set of paired elements. If **two** symbols are specified then **Φ_2** is defined to be values of the 2nd variable at the 2nd set of elements.

subrgn The subregion for computations. The subregion must be previously specified with a **LOCATE CORR**ELATION command. There is no default value.

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_{frq}** 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_{frq}** specification.

V_{frq} The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 for further details. By default the output is obtained only at the end of simulations.

OFF Any previously specified **CORR**ELATION commands for the specified variable and subregion are disabled. New commands may be subsequently specified.

EXAMPLES

CORRelation is TWO point type for variable U at ID=LOC_CORR2 at the end of simulations
CORRelation is TWO point type for variables U and V at ID=LOC_CORR2 output on file 'TWOPCORR.UV'
CORRelation is TWO point type for variables U and V at ID=LOC_CORR2 at the end of simulations
CORRelation TWO point OFF for variable U and T for the ID=L OC_CORR2

COMMAND DATUM

PURPOSE To specify datum coordinates, x, which are used to define the total hydraulic head, H of Equation 2.1.10.

SYNTAX DATU {N1, N2, N3}

N1,N2,N3 The (x,y,z) or (x,r,θ) datum elevation coordinate values, respectively, which are used to define the total hydraulic head. The default value for all of these is 0.

COMMENTS

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

COMMAND 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 named 'acr_GRID_QUALITY.TMP'.

The orthogonality index for any element is defined as the maximum of the skew ness angles for the faces of the element. For each face the skew ness 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.

$$\alpha_f = \mathrm{Cos}^{-1} \frac{\sum_{i=1}^{N23} A_i \, \delta x_i}{A_f \, \delta S}$$

The nodal distance for any element is defined as the maximum of the Eulerian distance for the nodes straddling the element faces. For any face, the Eulerian 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 {Φ} [fname] [subrgn] [V_{frq}] [TIME]**

Φ One and only one symbol for the dependent variable for debug output; valid symbols are listed in Table 6.7.1. There is no default value. A valid symbol must be specified.

fname The file name for output. The default file name is 'acr_DEBUG.TMP'. 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 the default file. The total number of open files can not exceed 64. 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 Sections 7.2.3 and 7.2.4. If no subregion is specified, the entire computational domain is selected.

V_{frq} The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 for further details. By default the output is obtained only at the end of simulations.

TIME By default, **V_{frq}** is the computation (and output) frequency in terms of number of steps. If this modifier is present, then **V_{frq}** is the time interval between successive outputs.

COMMENTS

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} [N1]**

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

$$A_{ij} X_j = B_i$$

is:

where, for the i^{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_MTRXxxxx.TMP where "xxxx" are replaced by the non-blank characters of the 1^{st} 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_l that pertains to the source or sinks in physical units), the forcing function (B_l), the diagonal component of the matrix coefficients (A_{ii}) and, the negative of the matrix coefficient for each neighbor ($-A_{ij}$, $i \neq j$).

N1 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

COMMAND 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 {Φ} [RATE | LIFE] [FIEL] [λ]

Φ A symbol that denotes the dependent variable for which the decay or reaction rate constant is specified. The valid symbols are listed in Table 6.7.1.

RATE The decay rate of Φ is specified; the units are those of inverse of time. This is the default option.

LIFE The half-life of Φ is specified; the units are those of time.

FIEL If the modifier **FIEL**D 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.

λ The decay rate of Φ 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 {Φ} [DRAG] [option] [subrgn] [C_D] [α] [N1, Nk], [Nk+1]**

Φ See Mode 1 specification.

DRAG The decay of dependent variable **Φ** is according to the drag law based on the flow speed. This adds a source, S_Φ, in the transport equation for **Φ** given by:

$S_\Phi = -0.5\ \rho\ \Phi\ C_D\ C_\Phi\ (\ U^2 + V^2 + W^2\ )^{\alpha/2}$, where C_D is the drag coefficient, C_Φ is a scaling factor, and U, V and W are components of flow velocity. C_D and C_Φ are dimensional constants such that S_Φ has appropriate units [e.g. M L T^{-2}]. For example, if N=1, and C_Φ is an area, then C_D is non-dimensional.

option Options selected for implementation of the source.

option	INTERPRETATION
VOLU	The coefficient **C_Φ** in the **S_Φ** term is set equal to the volume **(δV)** of the element.
AREA	The coefficient **C_Φ** in the **S_Φ** term is set equal to the area **(δA)** of the element face indicated by the **dir** modifier.
dir	The orientation index for the element boundary associated with the source. See Section 7.2.5 for available choices. There is no default value for this input.
NORM	The coefficient **C_Φ** in the **S_Φ** term is computed as: $C_\Phi = \sum_i A_i \bullet V_i$ where A_I is the i^{th} direction component of the element boundary area specified by **dir**. V_I are the values specified by **N3** through **Nk** (2 for 2D, and 3 for 3D).
DENS	The computed source, **S_Φ**, is further multiplied by density. The density may be specified as the last value, **Nk+1**, on the command. If this value is omitted, then the boundary value in the **dir** direction is used if the **AREA** or **NORM**AL modifiers are present, otherwise the local density for the element is used.

subrgn The subregion for which the input is specified. See Sections 7.2.3 and 7.2.4. 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.

α The power exponent, N, in the drag law (see Comments). The default value is 1.

N1, .., Nk The normalizing vector components, V_i, if the **NORM**AL modifier is present. Two values must be specified for 2D and 3 for 3D flows. There are no default values for this input.

Nk+1 The density value that multiplies the computed source, if the **DENS**ITY modifier is present. There is no default values for this input.

EXAMPLES

DECAy of U: DRAG law: cf=0.001, for previously SELEcted subregion
DECAy of T: DRAG law: cf=0.002, N = 0.80 for subregion ID=OBSTruction
DECAy for T: DRAG type: cf=0.001, N=0.5 multiply by AREA in X- direction for SELEcted subregion
DECAy for T: DRAG type: cf=0.1, N=0.5 X- dir & multiply by VOLUme. DENSity for SELEcted subregion
DECAy for T: DRAG type: cf=0.1, N=0.5 X- direction NORMalized 1. 0. DENSity for SELEcted subregion

COMMAND DEFINE

PURPOSE To define the value of a symbolic variable as a numeric or character expression.

SYNTAX **DEFI {variable} { Ψ | N1} [OFF]**

variable A symbolic variable, that at run-time, is replaced by user specified input value. **The 'variable' must be the 1st modifier following the DEFINE command. Only the first 8 characters are meaningful; any subsequent characters are ignored.** Once a variable has been defined, the character string, **Ψ**, or the numeric value, **N1**, replaces any occurrence of this variable in the user input.

Ψ The character string that replaces every occurrence of '**variable**' in the user input. 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.**

N1 The numerical value that replaces every occurrence of the '**variable**' in the user input. If both **N1** and **Ψ** modifier are specified, then **N1** input takes precedence.

OFF The previously defined variables are deactivated. New variables may be defined for input that follows.

COMMENTS

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 7.2.1. Up to 256 symbolic variables may be defined at any time.

EXAMPLES

DEFIne XXX = P in all input that follows
DEFIne PI = 3.1415
DEFInition OFF for all previous variables

COMMAND DENSITY

PURPOSE To specify the option and constants used to calculate density as a function of pressure, temperature, concentration of chemical species or other phase-space variables.

MODE 1: Constant Density

SYNTAX DENS { ρ_0 } [subrgn]

ρ_0 The numerical value (>0) for fluid density. There is no default value; a value must be specified.

subrgn The subregion for which the input is specified. See Sections 7.2.3 and 7.2.4. If no subregion is specified, the entire computational domain is selected.

EXAMPLES

DENSity of fluid =1.18 Kg/m*3
DENSity of fluid is =1000

MODE 2: **Generic Functional Form for Density**

SYNTAX **DENS {func [ξ]} [phase] [subrgn] {fname | N1 ..., Nn}**

func One of the modifiers listed in Table 7.2.3 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 7.2.4. If no variable is specified, then the independent variable is assumed to be time.

phase The phase for which the input is specified. **See Section 7.2.10 for available options**. By default the input pertains to the 1st 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 Sections 7.2.3 and 7.2.4. If no subregion is specified, the entire computational domain is selected.

fname The name of the file containing the numerical values N1 through Nn. This option can be used only if the selected function is a table or one of the series functions. See Section 7.2.2 for additional information.

N1, .., Nn The numerical constants and coefficients for the selected function. See Section 7.2.7 for more details. There are no default values for this input.

APPLICABILITY

The functional mode of the command is not available for the **TIDAL™** Software Tool; only a constant value may be specified.

COMMENTS

The density may also be set by the **SET** Command with **RHO** modifier. In this case the modifier **ALWA**YS must be specified if the density 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 **DENS**ITY command is recommended as the preferred mode of input.

EXAMPLES

DENSity of fluid is = 1.18 Kg/m^3

Generic examples for this command are given in Section 7.2.8. The command keyword (DENS) must replace the keyword used in these examples.

MODE 3: **Pre-Defined 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 7.2.10 for available options. By default the input pertains to the 1st 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 **POWE**R law or the coefficient for the **POLY**NOMIAL and **LINE**AR relations. The default value is 0.2 for the **POWE**R law eqution and 0 otherwise

β The critical temperature for the of the **POWE**R law, the coefficient for the **POLY**NOMIAL relation or the reference concentration for the **LINE**AR relation. The default value is 374.15 for **POWE**R law and 0 otherwise

γ The coefficient in the **POLY**NOMIAL and **LINE**AR relations. The default value is 0.

APPLICABILITY

This mode of the command is not available for the **TIDAL™** Software Tool.

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 4: Density from Gas Law

SYNTAX DENS {GAS} [INCO | COMP] [phase] [N1]

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_U is the universal gas constant, T is the temperature, T_a is the base to convert temperature to absolute units, m_j is the mass fraction of the j^{th} component of the gas species and Mj is the corresponding molecular weight.

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

phase The fluid phase for which the input is specified. See Section 7.2.10 for available options. By default the input pertains to the 1st phase of the fluid. This modifier is available only for the multi-phase versions of the **PORFLOW™** and **ANSWER™** Software Tools.

N1 Reference density for the fluid. If no value is specified then default value is computed from the reference pressure, temperature and gas properties

APPLICABILITY

This mode of the command is not available for the **TIDAL™** Software Tool.

EXAMPLES

DENSity from GAS law: reference value = 0.960 kg/m^3
DENSity from GAS law reference value computed from other input
DENSity GAS law in INCOmpressible mode

MODE 5: **Density of Brine as a Function of Salt Concentration**

SYNTAX **DENS {SALT} [ρ*] [α] [ρ_{max}] [ρ_{Salt}] [C_{S_Sat}]**

SALT Density varies according to the exponential law:

$$\frac{\rho}{\rho^*} = \min\{ \exp(\alpha\, C_s), \rho_{max} \} ,$$

where $\mathbf{C_S}$ is the mass concentration of salt ($0 \le \mathbf{C_S} \le 1$) in the brine mixture which is computed from a transport equation.; other symbols are defined below.

ρ* Reference density for the fluid, **ρ*.** The default value is 998.2.

α The exponent for the function; the default value is 0.6995.

ρ_{max} The maximum density of the mixture fluid; the default value is 1200.9.

ρ_{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 not available for the **TIDAL™** Software Tool.

COMMENTS

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 **ρ_{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 SALT function
DENSity SALT function; rho* = 998.2, alfa=0.6995, rho_max=1209
DENSity SALT function; rho* = 998.2, alfa=0.6995, rho_max=1209, rho_salt=2165, Csat=0.2643

COMMAND DIAGNOSTIC

PURPOSE To specify the options for diagnostic output of the values of variables, flux balances or convergence residual values.

SYNTAX **DIAG [Φ] [NOW | OFF] [ELEM] [OPEN] [HIGH] {N1, ..., Nn} [N_frq] [N_file] [fname]**
n=1 for unstructured, n = 2 for 2D and n = 3 for 3D input mode

Φ 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 6.8.1,
2) the symbols listed in Table 6.8.1 preceded by 'D',
3) the symbols listed in Table 6.7.1 preceded by a 'B'
4) the symbols listed in Table 6.7.1 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 **SCRE**EN command) by default. All 9 variables are directed to the screen if the **WIDE** modifier is present on the **SCRE**EN 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.

OFF Diagnostic output is suppressed.

NOW Diagnostic output is produced immediately.

ELEM By default for structured grids, the input of **N1, ..Nn** specifies the grid indices. 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.

OPEN If this modifier is present, and the user specifies an element for diagnostic output (see **N1, ..Nn)** that is located in a blocked region (**BLOC**K 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.

N1, ..Nn The grid index location for the element for which the output is obtained. For unstructured grids, the input is that of the element number. For structured grids the input consists of the grid indices (I,J,K) in the 3D or (I,J) in the 2D mode or the element number (if the **ELEM** modifier is present). If the specified node is located at the domain boundary, it is automatically adjusted to fall within the domain of computation.

N_frq 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.

N_file The frequency of output to the extended diagnostic file described by **fname** below. If no value is specified, then the default value is set equal to N_frq.

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.

COMMENTS

The 1st 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 which is defined in Section 4.5 and is controlled by the **CONV**ERGENCE 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 (**CONV**ERGENCE **REFE**RENCE 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

DIAGnostic node (4,8) ! 2D input mode; diagnostic printout every step
DIAGnostic node (4,8,7) ! 3D input mode; diagnostic printout every step
DIAGnostic output at element number 25 ! Unstructured mode
DIAGnostic output at element number 25 every 10 steps; also file 'Diagnostics.eqn'
DIAGnostics at (7,2,5) every 10 steps
DIAGnostic output for U, P and T at node (7,2) print every 10 steps; also 'EQUATIONS.DGN'
DIAGnostic output for TIME, U, BP and RT at node (7,2) print every 10 steps
DIAGnostic output HIGH level with TIME, U, BP and RT at node (7,2) print every 10 steps
DIAGnostic output for TIME, DTIMe, P, DP & RP at node (7,2) every 10 steps
DIAGnostic output: U, BP, RT and TIME in that order at (7,2) every 10 steps
DIAGnostic output: TIME, U, BP and RT at node (7,2) print every 10 steps
DIAGnostic output: TIME, U, BP & RT at node (7,2) print every 10 steps but every 20 steps for file output
DIAGnostic output OFF
DIAGnostic output NOW

COMMAND DIFFUSION

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

COMMENTS

This command is identical to the **COND**UCTION command.

EXAMPLES

See CONDUCTION **command**

COMMAND **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,ϕ, 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 **SOLV**E 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

DISAble FLOW
DISAble FLOW computations
DISAble ENTHalpy values read from the file for ANSWER software
DISAble FLOW
DISAble VELOcity correction for ANSWER
DISAble DENSity variations in flow equation (Boussinesq assumption)
DISAble DENSity variations in ALL equations for PORFLOW
DISAble FLOW and DENSity variation in ALL equation

MODE 2: Disable Options for Specific Variables

SYNTAX DISA {Φ} [CONV] [DIFF] [OFF]

Φ **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.

DIFF In the absence of the **OFF** modifier, the diffusion term in the transport equation for the specified variable is set to zero. If the **OFF** modifier is present, then the diffusion term is reinstated.

OFF Any previously disabled **CONV**, **DIFF** or **STOR** option is reinstated.

COMMENTS

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

DISAble CONVection term for T and C
DISAble DIFFusion and STORage term for U
DISAble CONVection and STORage terms for T
DISAble CONVection term for T is OFF (reinstate convection)
DISAble STORage term for T and C (same as steady state)
DISAble STORage term for T and C OFF (reinstate transient state)

COMMAND EBU

PURPOSE Specify the mixing limited rate and parameters for a given reaction using the eddy breakup model.

MODE 1: EBU Reaction Variables and Parameters

SYNTAX EBU {idreac} {Φ} [HARM] [C_{EBU}, 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:

$$R_{EBU} = C_{EBU} \quad \rho \frac{\varepsilon}{k} \Phi_{ebu}^{rms},$$

$$\Phi_{ebu}^{rms} = \min(\Phi_1,\ \alpha\Phi_2,\ \beta\Phi_3).$$

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:

$$R_{ACTUAL} = \min\left(R_{KIN},\ R_{EBU}\right), \text{ or}$$

$$R_{ACTUAL} = \frac{R_{KIN}\, R_{EBU}}{\gamma R_{KIN} + (1-\gamma) R_{EBU}}.$$

The first equation is used for the default mode of the EBU-limiter whereas the second equation is used if the **HARM**ONIC modifier is specified.

Φ One or more of the symbols in Table 6.8.1 which identify the EBU limiting species (Φ_1, Φ2, or Φ3). 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_{EBU} The empirical constant, C_{EBU}, in the R_{EBU} above. The default value is 3.

α, β The constants, α and β, in the EBU relation given above. The default value is unity.

γ The constant γ in the harmonic limiter as described below. It is the 2nd, 3rd or 4th 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_{EBU}] [I_{EBU}]**

C_{EBU} The global empirical constant, C_{EBU} , 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_{EBU} 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 is10.

EXAMPLES

EBU limit global rate constant = 3.2
EBU limit global rate constant is 3.2 . Start at step number 20.

COMMAND 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

COMMENTS

This command is part of the **IF-ELSE-ENDIF** construct; please see the **IF** command.

COMMAND **END**

PURPOSE To signify the end of a problem.

SYNTAX **END**

COMMENTS

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

COMMAND 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

COMMENTS

This command is part of the **IF-ELSE-ENDIF** construct; please see the **IF** command

COMMAND 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 7.2.2 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 **CLOS**E modifier is present.

attribute	Interpretation
FORM	File is opened in the FORMATTED mode. This is the default option.
UNFO	File is opened in the UNFORMATTED mode
UNKN	File status is defined as UNKNOWN. This is the default option.
NEW	File status is defined as NEW. An error will occur if the file already exists.
OLD	File status is defined as OLD. An error will occur if the file does not exists..
BOTH	File is avialable for both READ & WRITE operations. This is the default option.
READ	File is avialable only for READ operations.
WRIT	File is avialable only for WRITE operations.

filetype **One** of the modifiers below that identify the file to be opened or closed.

filetype	Interpretation
DEBU	Debug output file (**DEBU**G), Unit 17, is selected for the operation.
FLUX	Flux output file (**FLUX**), Unit 14, is selected for the operation.
HIST	Time History file (**HIST**ORY), Unit 13, is selected for the operation.
SAVE	The default Archive file (**SAVE**), Unit 11, is selected for the operation.
TRAC	Particle track file (**TRAC**K), 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.**

COMMENTS

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’
FILE CLOSe unit 11 !! same as archive SAVE file

COMMAND **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 [Φ] [subrgn]**

Φ **One or more** of symbols that denote the dependent variables for which the values are fixed for the selected subregion. Up to 20 symbols may be specified with one command. The valid symbols are listed in Table 6.7.1. If no symbol is specified, the values are fixed for all variables.

subrgn The subregion for which the input is specified. If no subregion is specified, the entire computational domain is selected. See Sections 7.2.3 and 7.2.4.

COMMENTS

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 **BOUN**DARY command). Multiple commands may be used to fix values in an arbitrary manner. The **INIT**IAL, **READ** or **SET** commands may be used to define the variable values. Once specified, these values remain unchanged unless modified by a subsequent **INIT**IAL, **READ** or **SET** command.

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 [Φ] [subrgn] { N1, ..., Nn }; n=6 for 2D and 8 for 3D input mode**

Φ A symbol that denotes the dependent variable for which input is specified. The valid symbols are listed in Table 6.7.1. 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 Sections 7.2.3 and 7.2.4.

N1, .., Nn The values of the matrix coefficients for the variable. For 2D simulation, with reference to Figure 4.1.5, 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 (Figures 4.1.3 and 4.1.4), respectively.

COMMENTS

This command provides a very powerful means to control regions of flow where special processes occur. It essentially provides a means to replace the built-in differential equations with alternative equations. A non-zero value of N1 simulates a source (>0) or sink (<0). A non-zero value of N2 controls the relative change from a previous value. The Φ's can also be updated by the **INIT**ial, **READ** and **SET** commands.

EXAMPLES

FIX T in ID=FIXD: matrix coeff: 0, 0.4, 6*0.1 !!!3D weighting factors
FIX T & P in SELEcted region: -0.01, 1. !!! decay at the rate of 0.01 delta_t
FIX matrix coeff: 0, 0, 0.5, 0, 0.5 !!! average of nodes to left & below
!!! command below gives node value as average of values at four neighbor nodes
FIX P in region ID= FX1 matrix coefficients: 0., 0., 0.25, 0.25, 0.25, 0.25

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 6.7.1. 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 Sections 7.2.3 and 7.2.4.

EXAMPLES

FIX T in ID=FIXD OFF
FIX T & P OFF for ID=FIX1
FIX all OFF in ID=FIXAll

COMMAND 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] [func (ξ)] [option] [subrgn] {dir} {N1}[fname|N2,..,Nn] [Φ=Nn+1, .., Φ=Nm] [Nm+1, Nk] [ρ_B]

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.

func One of the modifiers listed in Table 7.2.3 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 7.2.4. If no independent variable is specified, the variable is assumed to be time.

option Options selected for implementation of the source.

option	INTERPRETATION
VOLU	In the absence of the **TOTA**L modifier, the flow for each element is computed as: **Q = q δV.** Here **q** is the amount specified by the user and **δV** is the volume of the element. The **q**, in turn, is computed from **func(ξ)** and **N1** through **Nn**. If the **TOTA**L modifier is present, the amount for each element is computed as: **Q = q δV /V**, where **V** is the volume of the total **subrgn**.
AREA	In the absence of the **TOTA**L modifier, the source for each element is computed as: **Q = q δA,** where **δA** is the area of the element boundary indicated by **dir**. If the **TOTA**L modifier is present, the source for each element is computed as: **Q = q δA /A**, where **A** is the total area of the **subrgn** in the **dir** direction.
TOTAL	By default, if dir 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 **INTE**RNAL modifier is present then the wall treatment and wall diffusive flux are retained.
NORM	In the absence of the **TOTA**L modifier, the source, **Q**, is computed as: $Q = q \sum_i A_i \bullet V_i$ where A_I is the i^{th} direction component of the element boundary area specified by **dir**. V_I are the values specified by **Nn+1** through **Nk** (2 for 2D, and 3 for 3D). In the presence of the **TOTA**L modifier, **Q** is computed in a manner identical to that for the **AREA** modifier.
DENS	The computed source, **Q**, is further multiplied by density. The density may be specified as the last value, ρ_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 **dir** direction is used.

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. See Section 7.2.5 for available choices. There is no default value for this input.

fname The name of the file from which numerical values **N2** through **Nn** are read. This option can be used only if the selected function is a table or one of the series functions. See Section 7.2.2 for further information.

N1, .., Nn The numerical constants and coefficients for the selected function. See Section 7.2.7 for more details. There are no default values for this input.

Nn+1, .., Nk The normalizing vector components, V_i, if the **NOR**MAL modifier is present. Two values must be specified for 2D and 3 for 3D flows. There are no default values for this input.

ρ_B The density value that multiplies the computed source. It can only be specified if the **DENS**ITY modifier is present and then it must be the last value on the command. If no value is specified but the modifier **DENS**ITY is present, then default value is the boundary value at the node indicated by the **dir** modifier.

COMMENTS

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 **SOUR**CE **FLOW** command may be more appropriate. The injection at an exterior domain boundary is more appropriately accounted for by the **INLE**T 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 **INLE**T 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 **INTE**RNAL modifier should be used.

EXAMPLES

Generic examples for this command are given in Section 7.2.8. 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: EXPOnential 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} [func (ξ)] [TOTA] [option] [subrgn] {dir} {N1} [fname|N2,..,Nn] [Φ=Nn+1, .., Φ=Nm] [Nm+1, Nk] [ρ_B]**

MOME Velocity components of the injected flow are computed from the flow rate for the element, Q:

$$V_j = \frac{Q}{\rho_B A} n_j ,$$

where V_j are the computed velocity components of the injected flow in the j^{th} direction , ρ_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 **NORM**AL 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^{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.

TOTA See Mode 1 specification

option See Mode 1 specification. Unless the **NORM**AL modifier is present, it is assumed that the **AREA** modifier is in effect. Any specification of the **VOLU**ME modifier is ignored. All other modifiers can be used as for Mode 1 Specification.

subrgn See Mode 1 specification.

dir See Mode 1 specification. This modifier must be present for this mode of command.

fname See Mode 1 specification.

Φ See Mode 1 specification.

N1, .., Nn See Mode 1 specification.

Nn+1, .. Nm, These values represent the value of injected variable denoted by the symbol immediately preceding the value.

Nm+1, .., Nk The normalizing vector components, V_i, if the **NORM**AL modifier is present. Two values must be specified for 2D and 3 for 3D flows. There are no default values for this input.

ρ_B The density value to compute the velocity components. It can only be specified if the **DENS**ITY 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.

EXAMPLES

All the examples cited for Mode 2, except those with **VOLU**ME modifier, are applicable provided that the modifier **MOME**NTUM 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} [func (ξ)] [TOTA] [subrgn] {dir} {N1}[fname|N2,..,Nn] [Φ=Nn+1,..,Φ=Nm] [Nm+1, …, Nm+5] [DENS | SPEE] [ρ_B| V_S]**

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 ρ_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.

TOTA See Mode 1 specification

subrgn See Mode 1 specification.

dir The orientation index for the boundary associated with the source. See Section 7.2.5 for available choices. A value must be specified; there is no default value.

fname See Mode 1 specification.

Φ See Mode 1 specification.

N1, .., Nn See Mode 1 specification.

Nn+1, .. Nm, The values of injected variables denoted by the symbols immediately preceding the values.

Nm+1 The spherical angle, Θ 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.

Nm+2 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.

Nm+3, Nm+5 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_B, is specified as the last value on the command.

ρ_B See Mode 3 specification.

SPEE The injection speed, V_S, is specified as the last value on the command.

V_S The value V_S if the **SPEED** modifier is present. There is no default value for this input.

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 {Φ } {TABL} {MULT} [option] [subrgn] {N1} [fname | N2, ..,Nn]**

Φ **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.

fname See Mode 1 specification.

N1 The number of sets of data for the tabular functions.

N2, ..Nn The N1 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 Φ 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.

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] [func (ξ)] [option] [subrgn] {N1} [fname | N2, ..,Nn]**

ONLY By default the injected (or withdrawn) flow is assumed to 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.

func See Mode 1 specification.

ξ See Mode 1 specification.

option See Mode 1 specification.

subrgn See Mode 1 specification.

fname See Mode 1 specification.

N1, .., Nn The numerical constants and coefficients for the selected function. See Section 7.2.7 for more details. There are no default values for this input.

EXAMPLES

Generic examples for this command are given in Section 7.2.8. 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 {OFF} {subrgn}**

Φ 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

COMMAND 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 {Φ} [subrgn] [dir] [fname] [TIME] [V_{frq_file}, V_{frq}] [NOW]

Φ One, and only one, symbol for the dependent variable for which the flux-balance output is required. The valid symbols are listed in Table 6.7.1. 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 7.2.5 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 By default the flux output is directed to a file with the same name as the Standard Output Device but with the extension as "FLX". For example, if the Standard Output file is "PROBLEM.OUT", then the flux file is named "PROBLEM.FLX". If a file name is specified, then the flux output is directed to the named file. **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, $V_{frq\text{-}file}$ and V_{frq} are interpreted to be the frequency of output in terms of number of steps. If this modifier is present, then $V_{frq\text{-}file}$ and V_{frq} are interpreted to be the time interval between successive outputs.

$V_{frq\text{-}file}$ The frequency (step or time interval) at which the fluxes are written to the flux file specified by **'fname'**. See Section 7.2.11 for further details. The default value is 1.

V_{frq} 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 6.4) in a manner similar to $V_{frq\text{-}file}$. 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

COMMENTS

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_O and Q_N), inflow (Q_{in}), outflow (Q_{out}), and decay (Q_{decay}). The flux balance disparity, Q_d, is defined as:

$$Q_d = (Q_O - Q_N) + Q_{in} - Q_{out} - Q_{decay} ,$$

where Q_O 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 and decay, respectively.

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, Φ, the units of the cumulative flux are the units of Φ multiplied by the units of the density (ρ) 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.

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. Fluxes in Items 4 through 18 are 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. The instantaneous fluxes in Item 19 through 21 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 into the system minus the total outflow from the system
5.	Total inflow (Qin = Qc_in+Qd_in+Qsor):	Total inflow into the subregion by convective flux (See Item 11), diffusive flux (See item 12) and incoming (or positive) Sources (see 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.	Change from last time step	Change in the amount for property from the last time step
19.	Instantaneous convective flux (in-out):	The current net inflow of convective flux at all boundaries of the subregion.
20.	Instantaneous diffusive flux (in-out):	The current net inflow of diffusive flux at all boundaries of the subregion.
21.	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] [NOW] [V_{frq}] [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 6.8.1. If no symbol is specified then output is obtained for each active variables.

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
INLE	Boundaries specified by the INLET command are selected.
OUTL	Boundaries specified by the OUTLET command are selected.
OPEN	Boundaries specified by the OPEN command are selected.
IO	All boundaries specified by INLET, OUTLET or OPEN command are selected.
WALL	Walls specified by WALL or BLOCk command are selected.
EXTE	All external (or outer) boundaries of the computational domain are selected.
ALL	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 7.2.5 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. The total number of open files in any simulation can not exceed 64.

TIME By default, V_{frq} 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_{frq} specification.

V_{frq} The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 for further details. By default the output is obtained only at the end of simulations.

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 bndries
FLUX AVERage for T (temperature) at INLEt and OPEN ! INLEt & OPEN bndries
FLUX AVERage for T at INLEt and BOUNdaries ! INLEt & domain bndries
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} {Φ} {subrgn}**

OFF Any previously specified flux balance command for the specified variable and subregion is disabled. New **FLUX** commands may be subsequently specified.

Φ The symbol for the dependent variable for which the previously specified flux-balance computation is disabled. The valid symbols are listed in Table 6.7.1. 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

COMMAND FUEL

PURPOSE To specify the amount of carbon and hydrogen in the fuel, and the fuel enthalpy.

SYNTAX FUEL {N1, N2, N3}

N1 The fuel for a combustion process is assumed to have a chemical formula of the type C_nH_m. The N1 (>0) specifies the number of carbon atoms n. The default value is 10.

N2 The number of hydrogen atoms (>0), the m in the fuel formula. The default value is 19.

N3 The lower heat of combustion (>0) in energy per unit mass; the heat of combustion 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

COMMAND GAS

PURPOSE To specify composition and properties of the gas phase.

MODE 1: Gas Molecular Weights for Gas Constituents

SYNTAX GAS { [Φ_1=N1, Φ_2=N2, ..., Φ_3=Nn] }

Φ_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.

N1, .., Nn Molecular weights (> 0) for the variables in the order of the symbols. There are no default values; appropriate values must be specified. If a single 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 1st defined transport variable.

COMMENTS

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 **REAC**TION 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 Reference Pressure**

SYNTAX **GAS { PRES | P | SONI | GAMM } { N1 }**

PRES The reference datum gas pressure, p^* of Equation 3.1.4, is specified.

P Same as the modifier PRES above.

SONI The sonic speed for the gas is specified. This may be used to compute the Mach number of the flow (see the **OPTI**ON command).

GAMM γ, 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 **OPTI**ON command).

N1 The reference pressure, the sonic speed or γ for the gas. The default value for the reference datum pressure is 101325 for incompressible flow and 0 for compressible flow, 300 for the sonic speed and 1.4 for the ratio of specific heats (γ).

EXAMPLES

GAS reference PRESsure is 2.5E5
GAS reference P is 2.5E5
GAS SONIC speed is 275 m/s
GAS GAMMA is 1.2

MODE 3: **Universal Gas Constant**

SYNTAX **GAS { CONS } [func [ξ]] { N1 } [fname | N2 .., Nn]**

CONS The universal gas constant, R_u of Equation 3.1.4, is specified.

func One of the modifiers listed in Table 7.2.3 which denotes the functional form of the gas constant. If no function is specified, then the gas constant is assumed to be fixed.

ξ One of the independent variables listed in Table 7.2.4. If no variable is specified, the independent variable is assumed to be time.

fname The name of the file from which numerical values **N2** through **Nn** are read. This option can be used only if the selected function is a table or one of the series functions. See Section 7.2.2 for additional information.

N1, .., Nn The numerical constants and coefficients for the selected function. If no function is specified, then the default value is 8314. If a function is specified then there are no default values for this input See Section 7.2.7 for more details..

EXAMPLES

Generic examples for this command are given in Section 7.2.8. 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)

COMMAND GEOMETRY

PURPOSE To modify or specify the geometry for the computational domain.

MODE 1: Exchange Previously Specified Coordinates

SYNTAX GEOM { EXCH } { dir_1 } { dir_2 }

EXCH The coordinates in the two directions specified by dir_1 and dir_2 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_1, dir_2 Two of the **X**, **Y**, **Z**, **R** or **THETA** modifiers that, respectively, denote the x, y, z, r or θ coordinates to be exchanged with each other.

EXAMPLES

GEOMetry EXCHange X and Y
GEOMetry EXCHange X and R
GEOMetry EXCHange X and Z

MODE 2: **Rotate Previously Specified Coordinates**

SYNTAX **GEOM { ROTA } { dir_1 } { dir_2 } [N1, N2]**

ROTA The coordinates in the two directions specified by dir_1 and dir_2 are rotated in the plane defined by dir_1 and dir_2 by ϕ_1 and ϕ_2 degrees, respectively, according to the following equations:

$$x_{new} = x_{old} \cos \phi_1 + y_{old} \cos \phi_2$$

$$y_{new} = x_{old} \sin \phi_1 + y_{old} \sin \phi_2$$

dir_1, dir_2 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_1 and dir_2.

N1 The angle ϕ_1 (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.

N2 The angle ϕ_2 (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

GEOMetry ROTAte X and Y by 45 degrees
GEOMetry ROTAte X by 45 and Y by 135 degrees !!same as above
GEOMetry ROTAte X and Y by -45 deg.
GEOMetry ROTAte X and Y 30 and Y by 115 degrees from old x axis

MODE 3: **Scale Previously Specified Coordinates**

SYNTAX **GEOM { SCAL } { dir } { N1 } [N2]**

SCAL The coordinates in the direction(s) specified by **dir** are scaled according to the following equation:

$$x_{new} = x_{old} * N1 + N2$$

dir One or more of the **X**, **Y**, **Z**, **R** or **THETA** modifiers that, respectively, denote the x, y, z, r or θ coordinates each of which is scaled according to the equation given above.

N1 The scaling factor for the transformation defined above. There is no default value; a value must be specified.

N2 The offset for the coordinate. The default value is 0.

EXAMPLES

GEOMetry SCALe X by 0.3048
GEOMetry SCALe X by 0.3048 add 1.00
GEOMetry SCALe X and R by 0.3048 add 1.00
GEOMetry SCALe X, Y and Z by 0.3048 add 1.00

MODE 4: **Generate a Cylindrical Geometry in r-θ Plane**

SYNTAX **GEOM { CYLI | CIRC } (N1} [N2, N3]**

CYLI Generates a cylindrical grid for the computational domain in the r-θ plane.

CIRC Same as CYLI modifier.

N1 The radius of the cylinder. There is no default value; a value must be specified.

N2 The angle of the cylinder circumference in degrees. The default value is 180 degrees.

N3 The starting angle from the origin for the cylinder in degrees. The default value is 0 degrees.

COMMENTS

This command is available only if the computational domain is located in the 1st or 2nd quadrant.

EXAMPLES

GEOMetry is CYLIndrical with radius = 1.
GEOMetry is CIRCular with r = 1, theta = 90
GEOMetry is CYLIndrical with radius = 1., theta = 90
GEOMetry is CYLIndrical with r = 1., theta = 90, alpha=90

MODE 5: **Generate an Annular Geometry in r-θ Plane**

SYNTAX **GEOM { ANNU } { N1 } [N2, N3, N4, N5]**

ANNU Generates an annular geometry for the computational domain in the r-θ plane.

N1 The outer radius of the annulus. There is no default vale; a value must be specified.

N2 The inner radius of the annulus. The default vale is 0.

N3 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 vale is 0.

N4 The total angle for the outer arc of the cylinder in degrees. The default value is 180 degrees.

N5 The starting angle from the origin for the cylinder in degrees. The default value is 0 degrees.

COMMENTS

Currently this command is available only if the computational domain is located in the 1st or 2nd quadrant.

EXAMPLES

GEOMetry is ANNUlar with radius = 1.
GEOMetry is ANNUlar with rin=1, rout=0.5
GEOMetry is ANNUlar with rin=1, rout=0.5, epsilon=0.1
GEOMetry is ANNUlar with rin=1, rout=0.5, epsilon=0.1, theta = 90
GEOMetry 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_1 } { dir_2 }**

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:

$r = \text{sqrt} (x_1^2 + x_2^2)$

$\theta = \tan^{-1} (x_2 / x_1)$

dir_1, dir_2 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.

COMMENTS

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

GEOMetry convert R and THEta to CARTesian
GEOMetry convert Y and Z to CYLIndrical
GEOMetry convert X and Y to CYLIndrical !!! only with a 3D grid system

COMMAND GRAVITY

PURPOSE To specify the gravitational acceleration vector and its components.

SYNTAX GRAV { g_x, g_y, g_z }, [g] [RELA]

g_x, g_y, g_z 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_y is -9.81 for 2D flow and g_z 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_x, g_y, g_z are specified in a normalized mode.

RELA By default, for the **ANSWER™** Software Tool, the gravitational body force term, F_i in the i^{th} direction momentum equation is given by:

$$F_i = \rho g_i$$

If the RELATIVE modifier is present, then the body force term is given by:

$$F_i = (\rho - \rho^*) g_i$$

Where ρ^* is the reference density. However, the computed pressure, P, is no longer equal to the thermodynamic pressure, p, but is related to it by:

$$P = p - \rho^* \{ g_x x + g_y y + g_z z \}$$

This mode improves the stability of computations by decreasing the magnitude of the body force and hence the round-off errors.

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 gravitational force is always included for unsaturated (multi-phase, vadose or free surface) flow. For saturated flow if the GRAVITY command is specified, then the computations are assumed to be in terms of pressure head and all boundary and field values must be specified in terms of pressure head. If no GRAVITY command appears, then it is assumed that the computations are in terms of total head and all boundary and field values must be specified in terms of total head.

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_x) may be specified; the absolute value (**g**) will be set equal to this value since the other values are assumed to be zero.

COMMENTS

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

```
GRAVity constants are: 0., -9.81                ! Orientation against y-axis
GRAVity constants are: -6.937, -6.937, 0.       ! 45 deg to x & y
GRAVity constants are: 0., 0., -9.81            ! Orientation against z-axis
GRAVity constants are: 0., 0., -32.2, 32.2      ! Orientation against z-axis
GRAVity constant is 9.81
```

COMMAND 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] {N1} [N2, N3]

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 N1 through N3 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 can not be explicitly specified in this mode; they are automatically captured by a LOCATE command if the element next to the boundary is specified.

N1 The number of grid elements or grid nodes in the x direction. The number of nodes is denoted by IMAX in this document. The default value is 1 element or 3 nodes.

N2 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.

N3 The number of grid elements or nodes in the z or θ 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 N3 is not specified, then the 2Dsolution 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                      ! 1D problem: JMAX automatically set to 3
GRID is 32 by 27 by 1 NODEs     ! 2D problem: grid related input in 3D mode
GRID is 32 by 27 by 12 NODEs    ! 3D problem: grid related input in 3D mode
```

MODE 2: **Unstructured Grid Specification**

SYNTAX **GRID {UNST} [THRE] {N1} [NODE] [N2]**

UNST Grid is specified in an unstructured mode. In this mode, the user must specify a list of vertices and the connectivity of those vertices to the grid elements. The coordinates of the vertices must be specified on a file through the **COOR**DINATE command. The connectivity of the vertices to the elements is specified on a file through the **CONN**ECTIVITY command.

THRE By default the problem is assumed to be a 2D problem. This modifier must be used to specify a 3D problem.

N1 The number of grid elements for the unstructured problem. There is no default value for this input.

NODE By default the total number of nodes (one for each element plus one for each external boundary segment) are computed from the given connectivity information. If this modifier is present, then **N2** is interpreted to be the total number of nodes for the problems.

N2 The total number of grid nodes (sum of number of elements and number of external boundary segments) for the problem. This information is generated by most unstructured grid generators. If specified, this number helps in optimizing the use of the required memory arrays. By default this number is automatically generated by **ACRi** Software Tools from the connectivity information.

COMMENTS

All subregion input specification must be in terms of grid element numbers.

EXAMPLES

GRID is UNST with 100 elements ! 2D problem
GRID is UNSTructured with 1000 elements in THREE dimensional mode
GRID is UNSTructured with 1000 elements and 1256 NODES in THREE dimensional 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 **CONN**ECTIVITY command is not required. The coordinates of the vertices must be specified on a file through the **COOR**DINATE command.

VERTEX The **VERT**EX mode of the **CONN**ECTIVITY command is invoked. Please see the **CONN**ECTIVITY command for further details. This is the default option.

HYBRID The **HYBR**ID mode of the **CONN**ECTIVITY command is invoked. Please see the **CONN**ECTIVITY 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

MODE 5: **PLOT3D Format Files**

SYNTAX **GRID {PLOT} {fname} [UNFO] [XYZ]**

PLOT The grid information and grid coordinates are read from the specified file. The data on the file is assumed to be in PLOT3D format as follows: This command also generates an **ACRi** format grid file named 'acr_GRID_XYZ.TMP'.

```
READ(NUNIT) NBLOCKS
DO N = 1, NBLOCKS
    READ(NUNIT) NI. NJ. NK
    READ(NUNIT) ((( XC(I,J,K), I=1,NI), J=1,NJ), K=1,NK)
    READ(NUNIT) ((( YC(I,J,K), I=1,NI), J=1,NJ), K=1,NK)
    READ(NUNIT) ((( ZC(I,J,K), I=1,NI), J=1,NJ), K=1,NK)
    READ(NUNIT) (((IBLANK(I,J,K), I=1,NI), J=1,NJ), K=1,NK)   !May be omitted
ENDDO
```

NBLOCKS is the numbers of blocks of grid information for a multi-block grid. **Currently only single block mode is available.** Multi-block option will be available at a future date.

NI, NJ and NK are, respectively, the number of vertices in the x, y and z directions, for the current block.

XC, YC and ZC are vertex coordinates,

IBLANK is an index that denotes the nature of the vertex. A value of 0 means that the vertex is outside the domain of computational interest or is blocked by a solid; any other value means that the vertex is inside the computational domain.

fname The name of the file that contains the input coordinate values. See Section 7.2.2 for additional information. The number of values specified must be compatible with the format shown above.

UNFO By default the file is assumed to be formatted. If this modifier is present, then the file is in unformatted mode.

XYZ By default the PLOT3D file is assumed to contain IBLANK data. If this modifier is present, then it is assumed that the file contains only the grid coordinates and that the IBLANK data is not present. In this case the 4th READ statement inside the DO loop above is omitted while reading the PLOT3D file.

COMMENTS

Currently only single block mode is available. Multi-block option will be available at a future date.

EXAMPLES

GRID is PLOT3D format on file 'PLOT3D.XYZ'
GRID in PLOT3D data in UNFOrmatted data on file 'PLOT3D.XYZ'
GRID in PLOT3D XYZ data in UNFOrmatted data on file 'PLOT3D.XYZ'

COMMAND HISTORY

PURPOSE To obtain output of the time history for dependent variables at selected nodes.

MODE 1: Specification of Time History For a Set of Variables in a Subregion

SYNTAX HIST {subrgn} [Φ] {fname} [TABL] [PLOT] [NOW | OFF | ON] [TIME] [V_{frq}]

subrgn The subregion for which the time history is required. See Sections 7.2.3 and 7.2.4. For this mode to be invoked a subregion previously defined by a LOCATE command must be specified with an ID=subrgn or a SELECT/LOCATE modifier. There is no default value.

Φ One or more symbols that denote the variables for which the time history output is required. The valid symbols are listed in Table 6.8.1. 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.

fname The file name for time history output. See Section 7.2.2 for additional information. **Except for the 1st command, a file name must be specified with each command.** The default file name, for the 1st command, is "acr_HISTORY.TMP"'. To avoid confusion, it is recommended that a file name be specified with each command.

TABL The time history data are printed in a tabular form at the end of simulations.

PLOT A printer plot of time history of variables is generated at the end of simulations.

NOW One record of time history output is produced immediately.

OFF Time history output is discontinued.

ON Time history output is resumed if it was previously suppressed.

TIME By default, V_{frq} is interpreted as the frequency of output in terms of number of steps. If this modifier is present, then V_{frq} is interpreted as the 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 7.2.11 for further details. By default the time history is generated at every step.

COMMENTS

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 on file 'HISTORY.TIM' for the most recently SELECTED region
HISTory for U, V, W only ID=HIS on file "UVW.HIS"
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
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 {N1, .., Nn} [fname]**
N=m in unstructured mode, = 2m in 2D and =3m in 3D mode

N1, ..Nn 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.

COMMENTS

This mode of the command works in conjunction with Mode 3. **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

HISTory at (2,2), (2,5), (5,2), (11,17) and (17,11) ! Input in 2D mode
HISTory at elements 57, 33, 165 and 915 ! Unstructured mode
HISTory output OFF at (2,2) and (2,5)
HISTory at node (7,11) and (7,15) to be added to previous ones
HISTory at (2,2,2),(2,5,7),(5,2,7),(11,17,19) & (17,11,12) ! Input in 3D mode
HISTory OFF at (2,2,2) and (2,5,7)
HISTory 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] [TIME] [V_{frq}]**

Φ	See Mode 1 Specification.
fname	See Mode 1 Specification.
TABL	See Mode 1 Specification.
PLOT	See Mode 1 Specification.
NOW	See Mode 1 Specification.
OFF	See Mode 1 Specification.
ON	See Mode 1 Specification.
TIME	See Mode 1 Specification.
V_{frq}	See Mode 1 Specification.

COMMENTS

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

HISTory on file 'HISTORY.TIM'
HISTory for U, V, W only
HISTory every 20 steps
HISTory at TIME interval of 1.75 hours
HISTory for U, P, T, K on file 'HISTORY.OUT' every 20 steps
HISTory for U, P on file 'HISTORY.OUT' TIME=2.50 hrs; print TABLes also
HISTory for U, P, T, K at TIME interval 0.23: print TABLes also
HISTory U & T on 'HIS.NEW' at every 20 steps; print TABLes
HISTory U & T on 'HIS.NEW' at every 20 steps; print TABLes and PLOTs

MODE 4: **History at an Arbitrary Point in Space**

SYNTAX **HIST {COOR} {Φ} [fname] [LINE | SQUA] [N1,..Nn] [V_{frq}] [TIME]**

COOR History is required at an arbitrary point in space specified by its (x, y, z) coordinates.

Φ **One or more** symbols for the variables for which the history is required. No more than 10 symbols can be specified on one command. The valid symbols are listed in Table 6.7.1. There is no default value.

fname The file name for output. The default file name is 'acr_HISTORY_XYZ.TMP'. 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 the default file. The total number of open files in any simulation can not exceed 64. A summary of output is also printed to the standard output device at the end of simulations.

LINE The values at the specified (x, y, z) location are computed by linear interpolation from its nearest neighbors. This option is active by default.

SQUA The values at the specified (x, y, z) location are computed by inverse distance squared interpolation from its nearest neighbors.

N1, .., Nn The grid coordinates (x, y, z) of the point. 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.

V_{frq} The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 for further details. If present, this must be last value on the command. By default the time history is generated at every step.

TIME By default, **V_{frq}** is interpreted to be the frequency in terms of number of steps. If this modifier is present, then **V_{frq}** is interpreted to be the time interval between successive outputs.

COMMENTS

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 COORdinates (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] [OFF] [V_{frq}] [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 6.7.1. 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. The default file name is 'acr_SOURCE.TMP'. 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 the default file. The total number of open files in any simulation can not exceed 64. A summary of output is also printed to the standard output device at the end of simulations.

OFF Any previously specified command for the specified Φ and **subrgn** is deactivated.

V_{frq} The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 for further details. By default the time history is generated at every step.

TIME By default, **V_{frq}** is interpreted to be the frequency in terms of number of steps. If this modifier is present, then **V_{frq}** is interpreted to be the time interval between successive outputs.

COMMENTS

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} {Φ} [subrgn] [fname] [OFF] [V_{frq}] [TIME]**

STOR Output for the storage or accumulation term for Φ is required.

Φ 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 6.7.1. 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. The default file name is 'acr_STORAGE.TMP'. 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 the default file. The total number of open files in any simulation can not exceed 64. A summary of output is also printed to the standard output device at the end of simulations.

OFF Any previously specified command for the specified Φ and **subrgn** is deactivated.

V_{frq} The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 for further details. By default the time history is generated at every step.

TIME By default, **V_{frq}** is interpreted to be the frequency in terms of number of steps. If this modifier is present, then **V_{frq}** is interpreted to be the time interval between successive outputs.

COMMENTS

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 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

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

COMMAND 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 | N1} then

1st block of FREEFORM statements

[ELSE

2nd block of FREEFORM statements]

ENDIF

variable A symbolic variable that at run-time is compared to the value assigned to Ψ or **N1**. **The 'variable' must be the 1st 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 **N1** 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 1st block of statements is executed and the 2nd block, if present, is ignored. If the status is false then the 1st block of statements is ignored and the 2nd 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.**

N1 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 1st block of statements is executed and the 2nd block, if present, is ignored. If the status is false then the 1st block of statements is ignored and the 2nd block, if present, is executed. If both **N1** and Ψ are specified, then **N1** input takes precedence.

COMMENTS

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 2nd block is optional. However, if the 2nd 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.**

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
```

COMMAND **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 **INCL**UDE command provides for unlimited nesting in that the file being incorporated may itself contain embedded **INCL**UDE 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.

COMMENTS

In one important aspect, the INCLude keyword is treated differently than other FREEFORM™ keywords commands. This keyword does not have to start in the 1st column (character) of a command line. That is **INCL**UDE command can be indented, if desired. INCL must, of course, be the 1st four non-blank characters on the command. This is different from all other keyword commands which must start with the 1st 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
INCLude ? ! user will be prompted

COMMAND 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 1st 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 1st 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.

COMMENTS

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)
     &                                                    to (1.0,
3.0, 2.0)
INDEnt OFF
```

Here, the "&" character in the 3rd 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
```

COMMAND INITIAL

PURPOSE To specify the initial conditions for dependent variables for structured grids

SYNTAX INIT { $\Phi = \Phi_0$ } [N1 ..., Nn]

Φ The symbol that denotes the field variable for which input is provided. The valid symbols are listed in Table 6.8.1. One, and **only one**, symbol must be specified. There is no default value.

Φ_0 The initial value for the variable.

N1, .., Nn These values define the subregion to which the input is applied. In the default mode, the subregion is defined by four values for the 2D and 6 for the 3D input mode. If only one value, that is **N1**, is specified, then it is assumed that **N1** 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

INITial U is 0.1 everywhere
INITial T is 1.E-3 from (2,2) to (7,9) ! 2D input mode
INITial T is 1.E-3 from (2,2,2) to (7,9,4) ! 3D input mode
INITial T is 1.E-3 for zone 5

COMMAND INJECTION

PURPOSE To specify properties and parameters relating to injection of water films curtains or to condensation.

MODE 1 Location and Properties for Water Films

SYNTAX INJE {FILM} [subrgn], {TINJ=V_{Tinj}} {QINJ=V_{Qinj}} [TIME] [TEMP]
[STAR=V_{Start} |STAR NOW] [COOR=V_X, V_Y [, V_Z]]
[STOP=V_{Stop} | STOP NOW] [ELAP=V_{Elaps}]
[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 Sections 7.2.3 and 7.2.4 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.

NOW By default, keywords **STAR**t and **STOP** must be followed by values. If **NOW** is set with **STAR**T/**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 or Time (if modifier **TIME** is present) or Temperature (if modifier **TEMP**ERATURE is present) value.

ELAP Introduces the film duration. By default this is defined by **INJE FREQ** for all injections.

V_{Elap} Step or Time (if modifier **TIME** is present) 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.

COOR If modifier **TEMP** is present then this keyword must be added to define coordinates of the point at which film temperature monitored.

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 Name of the history file. By default the file name is **'acr_FILM.TMP'**.

EXAMPLES

INJEction **FILM** ID=FILM, **TINJ**ection=280 .Kelvin, **QINJ**=10. kg/s, **HIST**ory every 1. step, 'FILM.HIS'
INJEction **FILM** ID=FILM STARt TEMP is 340 Kelvin with COORinates (2.00,3.00,2.00)
ELAPS maximum TIME is 180. seconds

MODE 2 Location and Properties for Water Curtains

SYNTAX INJE {CURTAIN} [ID=subrgn]
{QINJ=V_{Qinj}} [DIAM=V_{Diam}] [TIME] [TEMP]
[STAR=V_{Start} |STAR NOW] [COOR=V_X, V_Y[. V_Z]]
[STOP=V_{Stop} |STOP NOW] [ELAP=V_{Elaps}]
[HIST=V_{Hist} [fname]]

subrgn The subregion for which the input is specified. **This subregion must be defined with LOCATE INJECTION command.** See Sections 7.2.3 and 7.2.4 for additional details.

QINJ Introduces the mass rate of water in the film. There is no default value; a positive value must be specified.

V_{Qinj} Mass rate of water in the film

DIAM Introduces droplet diameter of water curtain

V_{Diam} Droplet diameter. 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.

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 or Time (if modifier **TIME** is present) or Temperature (if modifier TEMP is present) value.

ELAP Introduces the film duration. By default this is defined by **INJE FREQ** for all injections

V_{Elap} Step or Time (if modifier **TIME** is present) value.

STOP Introduces 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.

COOR If modifier **TEMP** is present then this keyword must be added to define coordinates of the point at which film temperature monitored.

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, a history file is written.

V_{Hist} Step or Time (if modifier **TIME** is present) frequency of history computation.

fname Name of the history file. By default the file name is **'acr_CURTAIN.TMP'**.

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'

MODE 3 **Location and Properties for Condensation**

SYNTAX **INJE {COND} {subrgn} [dir] [option] [TIME] [HIST=V_{Hist} [fname]]**

subrgn The subregion for which the input is specified. See Sections 7.2.3 and 7.2.4..

dir The orientation index for the boundary associated with the film. See Section 7.2.5 for available choices.

option

option	INTERPRETATION
DROP	The droplet condensation model is activated. This is the default option.
FILM	The film condensation model is activated.

TIME By defaults, values of V_{Hist} (see below) is a time step frequency. If this modifier is present, then the value is in time units.

HIST By default no history file of the water film is given. 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 Name of the history file. By default the file name is **'acr_CONDens.TMP'**.

COMMENTS

If **FILM** model is activated, the water film falling direction is defined the **GRAV**ITY direction. By default this is the Z-axis in the decreasing order of Z.

EXAMPLES

INJEction CONDensation ID=NORD HISTory every 1 step, file is 'COND_NORD.his'
INJEction COND ID=NORD HISTory TIME every 0.1 second, file is 'COND_NORD.his'

MODE 4: **Injection Computations Parameters**

SYNTAX **INJE { FREQ } { N1 } [N2, N3]**

FREQ The spray computation frequency parameters are specified.

N1 The frequency of droplet computations in terms of computational steps. The default value is 10.

N2 The first computational step at which the droplet computations are performed. The default value is 1.

N3 The last computational step at which droplet computations are performed. The default frequency is set to 9999999.

COMMENTS

Injection computations are automatically triggered if a **INJE**CTION command in Mode 1 through 3 is encountered. This input is required only if the user wants to modify the default values.

This command will be modified if the **NOZZLE FREQuency** command is also active. Nozzles and water injections will be computed at the same frequency.

EXAMPLES

INJEction FREQuency of calculations: 1 start at step number 10
INJEction FREQuency of calculations: 10 start at step number 10 and stop at step 500

COMMAND INLET

PURPOSE To specify an inflow boundary for the domain of computation

SYNTAX INLE {dir} [subrgn] [DIAG] [OFF]

dir The orientation index for the inlet boundary. See Section 7.2.5 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.

COMMENTS

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 **INIT**IAL, **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

INLEt at X- boundary at left (minimum x)
INLEt at Y- boundary for the most recently SELEcted subregion
INLEt at X+ boundary for subregion ID = INFLow
INLEt OFF at X+ boundary for subregion ID = INFLow

COMMAND INTEGRATION

PURPOSE Choice of discretization scheme for integration of flow, heat and mass transport equations.

SYNTAX **INTE [Φ=N1, Φ=N2, ..., Φ=Nn] { HYBR | COND | QUIC | CENT | ACRI | UPWI }**

Φ One or more of the symbols that denote the heat or mass transport variables for which the profile specification is effective. The valid symbols are listed in Table 6.7.1. If no symbol is specified, the input is assumed to be effective for all variables.

HYBR The hybrid scheme (Runchal, 1972 is used for integration. This is the default option.

COND The CONDIF, a total variation diminishing scheme (Runchal, 1987, is used for integration. This option is currently available in a limited manner. Please check with ACRi for its use.

QUIC An 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.

CENT The 2^{nd} 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.

ACRI The stabilized 2^{nd} order central-difference scheme is used for integration. The scheme uses 2^{nd} 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.

UPWI The 1^{st} order upwind scheme is used computing convective fluxes whereas the diffusive fluxes are still computed by the 2^{nd} order central differences. The scheme is unconditionally stable in the linear sense but may increase numerical diffusion under certain conditions.

N1, .., Nn The diffusion control parameter (quantity R_{max} of Runchal, 1987) for the variable denoted by the symbol immediately preceding the value if the CONDIF scheme is selected; otherwise this input is ignored. The recommended values are between 2 and 10. Numerical diffusion decreases with increasing value of this parameter. The default value is 10.

COMMENTS

The default option should be adequate for most applications. However, if the local grid Peclet number (Equation 4.2.1) is significantly larger than 10 and the direction of flow is not (approximately) aligned with any of the coordinate directions, the CONDIF or QUICK scheme may be desirable. See Sections 4.2 and 4.4 for additional discussion.

EXAMPLES

INTEgration for U by CONDif scheme
INTEgration for T by HYBRid profile (same as default)
INTEgration for T by modified QUICk scheme
INTEgration for T=10., C=8., C2=5. by CONDif scheme
INTEgration for T=10., C=8., C2=5. by CENTral Difference scheme

COMMAND **LAMINAR**

PURPOSE To select laminar flow option.

SYNTAX **LAMI**

COMMENTS

By default the ANSWER™ Software Tool selects the laminar or turbulent mode of computations automatically. If any of the turbulence related quantities (k, ε, 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

LAMInar flow for this problem.
LAMInar flow ! by default

COMMAND LATENT

PURPOSE To specify the Latent heat of the fluid.

SYNTAX LATE { func [ξ] } [phase] [subrgn] { fname | N1 ..., Nn }

func One of the modifiers listed in Table 7.2.3 that denotes the functional form of the latent heat. If no function is specified then the value is assumed to be constant.

ξ One of the independent variables listed in Table 7.2.4. If no variable is specified, then the independent variable is assumed to be time.

phase The phase for which the input is specified. **See Section 7.2.10 for available options.** By default the input pertains to the 1st 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 Sections 7.2.3 and 7.2.4. If no subregion is specified, the entire computational domain is selected.

fname The name of the file containing the numerical values N1 through Nn. This option can be used only if the selected function is a table or one of the series functions. See Section 7.2.2 for additional information.

N1, .., Nn The numerical constants and coefficients for the selected function. See Section 7.2.7 for more details. There are no default values for this input.

COMMENTS

This command is available only under special consulting arrangement with ACRi.

EXAMPLES

LATEnt heat = 1234

Generic examples for this command are given in Section 7.2.8. The command keyword (**LATE**NT) must replace the keyword used in these examples and the dependent variable **(Φ)** must be omitted. Only the functional form, the independent variable and the numerical values need to be specified.

COMMAND LIMIT

PURPOSE To specify the limiting values for field variables

SYNTAX LIMI {Φ} {Φ_{MIN}, Φ_{MAX} | OFF}

Φ	A symbol that denotes the variable for which the limits are being set.
OFF	All previously set limits for the selected variable are deactivated.
Φ_{MIN}	The minimum permissible value for the variable. Any computed value that is less than Φ_{MIN} is set to Φ_{MIN}.
Φ_{MAX}	The maximum permissible value for the variable. Any computed value that is greater than Φ_{MAX} is set to Φ_{MAX}.

COMMENTS

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

LIMIt for T minimum = 300
LIMIt for T minimum = 0 maximum = 100
LIMIt for T OFF !Disable any previously specified or internally set limits

COMMAND 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 [ID=idsub] [FIEL] {N1, ..., Nn} , [Nn+1, Nn+2, Nn+3] [fname]

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 1st 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 1st separator character. A list of separator characters is given in Appendix B. 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 **LOCA**TE or **SELEct** on the command can make a generic reference to the specified subregion until the next **LOCA**TE or **SELE**CT command. See Sections 7.2.3 and 7.2.4 for additional details.

FIEL If the **LOCA**TE command identifies an element right next to an **exterior** boundary of the computational domain, then the node that defines the boundary value is included by default in the definition of the subregion. If the **FIEL**D modifier is present, then the subregion comprises only the interior filed nodes (or elements) and the exterior boundary nodes are not included in the subregion.

Presence of this modifier makes no difference to those commands that operate only on the interior field elements such as the **FIX** and **SOUR**CE command. Other commands, such as the **SET** and **COND**UCTIVITY, which can be used to set values of variables at both field and boundary nodes will be affected by this modifier.

N1, .., Nn The grid indices (I,J,K) for the subregion. A total of 4 numerical values for 2D and 6 for 3D input must be provided. If the values specified lie outside this range then the input values are clipped to lie in the valid range specified by the **GRID** command. There is no default value; a valid set of values must be specified. See Section 7.2.3 for additional information.

Nn+1 The x-direction node interval for subregion. Only those nodes occurring at this interval, starting with N1 specification are considered to be part of the subregion. For example, a specification of 3 results in the nodes located at I=N1+1, N1+4, N1+7, etc. to be included in the selected subregion; the intermediate nodes are excluded. The default value is 1.

Nn+2,Nn+3 The y and z direction node intervals in the manner described above. The default value is 1.

fname If a valid file name is present (see Section 7.2.2) 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 written to the file.

EXAMPLES

LOCAte subregion from (6,10) to (31,10)
LOCAte subregion (6,10) to (31,10) with ID=DMN1
LOCAte region ID=DMN2 as: (6,10,1) to (31,10,15) with interval (2,3,1)
LOCAte region (6,10) to (31,10) with interval (2,3)
LOCAte subregion (1,1) to (22,22) only FIELd nodes as ID=INNR
LOCAte 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 {COOR} [ID=idsub] [INTE] [NOT] [EXCL] [FIEL | BOUN] {N1, ..., Nn} [fname]**

COOR **One or two** "windows" are specified by their grid (x, y, z) coordinates. Each window is defined by 2 sets of coordinates. The 1st set defines the "lower-left" corner and the 2nd set the "upper-right" corner of the window. The second window is 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 **FIEL**D and **BOUNdary** modifiers.

idsub See Mode 1 Specification.

INTE If two windows are defined, then by default it is the union (elements that belong to either) which is selected. If this modifier is present, then it is the intersection (elements that belong to both) that is selected.

NOT This modifier selects the union of the two windows minus their intersection (elements that belong to one but not both). This modifier is effective only if the INTErsection modifier is also specified.

EXCL If the modifier is present, then the specified elements 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.

FIEL See Mode 1 Specification.

BOUN If the **BOUN**dary modifier is present, then only the external boundary nodes are selected. Any elements interior to the computational domain are excluded.

N1, .., Nn 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 for each window. See Section 7.2.3 for additional information. There is no default value; a valid set of values must be specified.

fname See Mode 1 Specification.

EXAMPLES

LOCAte region with COORdinates (0., -20.) to (11571.5, 80.)
LOCAte subregion ID = DMN2 for grid COORdinates (0., 0.) to (100, 1500)
LOCAte subregion ID = DMN2 for COORdinates (0., 0.) to (100, 1500) EXCLude selection
LOCAte ID = DMN2 for grid COORdinates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2)
LOCAte grid COORdinates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2) and EXCLude selected
LOCAte COORdinates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2) select INTErsection & EXCLude
LOCAte COORdinates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2) select INTErsection & EXCLude
LOCAte COORdinates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2) select NOT INTErsection & EXCLude
LOCAte COORdinates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2) select output on "DMN2.LOC"
LOCAte COORdinates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2) select only BOUNdary

MODE 3: **Subregion Specification by a Polygonal Window**

SYNTAX **LOCA {POLY} [ID=idsub] [EXCL] [FIEL | BOUN] {N1, ..., Nn} [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 **FIEL**D and **BOUN**DARY 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 **BOUN**DARY modifier is present, then only the external boundary nodes are selected. Any elements interior to the computational domain are excluded.

N1, .., Nn N1 through Nn are the grid coordinates (x, y) or (x, y, z) for the vertices of the polygon. The number of values must equal N23D * N_{vertex} where N23D is 2 for 2D and 3for 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.

EXAMPLES

LOCAte POLYgonal region (x, y) coordinates are: (0, 0) (1,0) (1,1) (0,1)!2D rectangle
LOCAte POLYgon (x, y) coordinates are: (0, 0) (1,-1) (2,0) (1,1) !Diamond shaped region
LOCAte POLYgon (x, y): (0, 0) (0.5,-0.866), (1.5,-0.866) (2,0) (1.5,0.866) (0.5,0.866) !Hexagon
LOCAte 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
LOCAte region ID=DIAMOND POLYgon (0, 0) (1,-1) (2,0) (1,1) output on “FILE.LOC”
LOCAte 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] {N1} [N2, . Nn] [IJK | ELEM] [Nn+1,..Nm], [Nm+1] [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 **FIEL**D modifier is present.

idsub See Mode 1 specification.

FIEL See Mode 1 Specification.

BOUN If the **BOUN**DARY modifier is present, then only the external boundary nodes are selected. Any elements interior to the computational domain are excluded.

N1 The diameter of the cylinder of the identified subregion.

N2, . Nn 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 **[N2, . Nn]** 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 **[N2]** specifies the element which is at the center of the cylinder. Only 1 value must be specified.

Nn+1, . Nm 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.

Nm+1 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 ≤ 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 than 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.

EXAMPLES

LOCAte CIRCle dia=0.4, center coordinates (2., 0.5) !2D
LOCAte CIRCle dia=0.4, center coordinates (2., 0.5) external BOUNdary only !2D
LOCAte CIRCle dia=0.4, center coordinates (2., 0.5, 0.5)
LOCAte CIRCle dia=0.4, center coordinates (2., 0.5, 0.5) normals (1.,1.0.) ! 45 degree in xy
LOCAte CYLInder dia=0.4, center coordinates (2., 0.5, 0.5) normals (1.,1.0.) half length=0.5
LOCAte CYLInder dia=0.4, center (2., 0.5, 0.5) normals (1.,1.0.) half length=0.5 file 'Cylinder.loc'
LOCAte CYLInder dia=0.4, center IJK at (11,5,7) normals (1.,1.0.) half length=0.5
LOCAte CYLInder dia=0.4, center ELEMent at (147) normals (1.,1.0.) half length=0.5

MODE 5: **Specification of a Spherical Subregion**

SYNTAX **LOCA {SPHE} [ID=idsub] [FIEL] {N1} [N2, . Nn] [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 **FIEL**D modifier is present.

idsub See Mode 1 specification.

FIEL See Mode 1 Specification.

N1 The radius of the sphere for the identified subregion.

N2, . Nn The (x,y) or the (x,y,z) coordinates of the center of the sphere.

fname See Mode 1 Specification.

EXAMPLES

LOCAte SPHEre radius=0.4, center coordinates (2., 0.5) !2D
LOCAte SPHEre radius=0.4, center coordinates (2., 0.5, 0.5)
LOCAte SPHEre radius =0.4, center coordinates (2., 0.5, 0.5) FIEL node only
LOCAte 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] {N1, N2} [N3, . Nn] [IJK | ELEM] [Nn+1,..Nm], [Nm+1] [fname]**

ANNU An annular subregion is specified. Only the internal elements can be specified. Associated boundary nodes are automatically included unless the **FIEL**D modifier is present.

idsub See Mode 1 specification.

FIEL See Mode 1 Specification.

N1 The outer diameter of the annulus (circular or cylindrical) of the identified subregion.

N2 The inner diameter of the annulus (circular or cylindrical) of the identified subregion.

N3, . Nn 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 **[N3, . Nn]** 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 **[N3, . Nn]** specifies the element which is at the center of the cylinder. Only 1 value must be specified.

Nn+1, . Nm See Mode 4 Specification.

Nm+1 See Mode 4 Specification.

fname See Mode 1 Specification.

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] {N1} [FIEL] [fname]**

MATE The subregion is identified by a Material type of Zone number.

ZONE Same as MATE modifier.

idsub See Mode 1 specification.

N1 The material type or zone number that previously appeared in a **MATE**RIAL (or an equivalent) command. The material type may denote a non-contiguous and non-rectangular subregion.

FIEL See Mode 1 Specification.

fname See Mode 1 Specification.

COMMENTS

This mode of the **LOCA**TE command is currently implemented in a limited manner; please consult ACRi before its use.

EXAMPLES

LOCAte MATErial 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 {COOR} {LIST} [ID=idsub] [FIEL] [fname] {N1 | N1, …, Nn }**

COOR 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 **FIEL**D modifier is present.

LIST This modifier denotes that the **COOR** 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.

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 (**N1**). 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 7.2.2) must be specified.

N1 If the file input mode with a valid **fname** is activated, then **N1** denotes the number of elements to be read from the file. If the file input mode is not activated, then **N1, …. Nn** are described below.

N1,..,Nn 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.

EXAMPLES

```
LOCAte COORdinate LIST:  (0.,0.), (2,2), (1.53,1.37), (23.1,27.2)          ! List for 4 points in 2D
LOCAte COORdinate LIST:  (0.,0,0), (2,2,2), (1.53,1.37,1), (23.1,27.2,2)   ! List for 4 points in 3D
LOCAte COORdinate LIST of  500 elements as ID=BIG from file 'XYZ.LST'      ! Read from file
LOCAte COOR LIST 500 as ID=BIG from file 'XYZ.LST' only FIELD              ! Read file; only field
LOCAte 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 } [EXTE] [ID=idsub] [FIEL] [fname] {N1 | N1, …, Nn }**

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. Unless the modifier **EXTE**RIOR is present, only the internal elements of the computational domain can be specified.

IJ Same as **IJK**.

EXTE By default only the internal elements of the domain of computations can be included in the list. If this modifier is present, then the list may include the grid indices for the corners, edges and boundaries of the domain of computations.

idsub See Mode 1 specification.

FIEL 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 (**N1**). 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 7.2.2) must be specified.

N1 If the file input mode with a valid **fname** is activated, then **N1** denotes the number of elements to be read from the file. If the file input mode is not activated, then **N1, …. Nn** are described below.

N1,..,Nn 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] [fname] {N1 | N1, …, Nn }**

LIST Explicit list of element numbers is specified. Only the internal elements can be specified. Associated boundary nodes are automatically included unless the **FIEL**D 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.

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 (**N1**). 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 7.2.2) must be specified.

N1 If the file input mode with a valid **fname** is activated, then **N1** denotes the number of elements to be read from the file. If the file input mode is not activated, then **N1, …. Nn** are described below.

N1,..,Nn The element numbers that define the subregion. .

EXAMPLES

LOCAte element numbers LIST: 1, 2, 3, 7, 17, 29 ! Structured Grid
LOCAte element numbers: 1, 2, 3, 7, 17, 29 ! Default for Unstructured Grid
LOCAte LIST of 500 elements from file 'ZONE.BIG' ! Read from file
LOCAte 5000 elements from file 'ZONE.BIG' ! Read from file for Unstructured Grid
LOCAte 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] { N1, N2 } [N3]**

SEQU A sequence of elements is specified. Only the internal elements can be specified. Associated boundary nodes are automatically included unless the **FIEL**D 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.

N1, N2, N3 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 **N3** is not specified, it is assumed to be 1.

EXAMPLES

LOCAte ID=lLISTSEQ of SEQUential elements from 1 through 51 interval=10
LOCAte SEQUence from 31 to 51 ! All elements from 31 to 51,inclusive

MODE 12: **Specification of a Given Location as a Station**

SYNTAX **LOCA {STAT} {ID=idsub} {N1, ..., Nn} [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**.

N1, .., Nn 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

LOCAte STAT ID=LOC2 at (0., -20.)
LOCAte STATion ID=LOC2 at coordinates (x=50., y=0., z=0.20)
LOCAte STATion ID=LOC2 at coordinates (x=50., y=0., z=0.20) output on file 'Station.Loc'

MODE 13: **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 1st element in the 1st subregion is paired with the 1st element of the 2nd subregion, and so on. This type of subregion can only be used with certain commands (e.g. **CORR**LELATION) where the values at the elements in the 1st set are paired with, or dependent on, the values of the elements in the 2nd set.

idsub1 The name of the 1st 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 2nd 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 8 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 **LOCA**TE or **SELE**CT on the command can make a generic reference to the specified subregion until the next **LOCA**TE or **SELE**CT command. See Sections 7.2.3 and 7.2.4 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 7.2.2) must be specified.

EXAMPLES

LOCAte CORRelated Elements from ID=REGION1 and ID=REGION2
LOCAte CORRelated Elements from ID=REGION1 and ID=REGION2 as ID=MATCHED on file='match.fil'

MODE 14: **Boundary Specification by Paired Element and Surface Numbers**

SYNTAX **LOCA { PAIR } [ID=idsub] [fname] {N1 | N1, N2, N3, … , Nn]**

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 (**N1**). 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 7.2.2) must be specified.

N1 If the file input mode is activated, then, **N1** 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 **N1, N2** below. If the file input mode is not **activated, then N1 is interpreted as given below.**

N1,N2 **N1** is the element number and **N2** is the surface number for the 1st 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 Sections 7.2.3 and 7.2.4 for further details of the manner in which these surface numbers are assigned.

N3,..,Nn The element number and a surface number for the rest of the elements which comprise current subregion in the manner **N1** and **N2**.

EXAMPLES

LOCAte (element, surface) PAIRs: (35,1), (53,3), (77,2), (13,4)
LOCAte PAIR : (35,1), (53,3), (77,2), (13,4) as boundary ID=BNDRy
LOCAte PAIR : 500 sets as ID=BNDRy from file 'BOUNDARY.BIG'
LOCAte PAIR : (35,1), (53,3), (77,2), (13,4) as boundary ID=BNDRy
LOCAte PAIR : (35,1), (53,3), (77,2), (13,4) as boundary ID=BNDRy
LOCAte PAIR : (35,1), (53,3), (77,2), (13,4), (28,5), (33,3), (35,6)
LOCAte PAIR (35,1), (53,3), (77,2), (13,4), (28,5), (33,3), (35,6) output on file "Boundary.LOC'

MODE 15: **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 (**N1**). 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 7.2.2) 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
ANY	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 **ANY** modifier is specified then a boundary element is included if any of its vertices is included in the list.
ONE	A boundary element is included if at least one of its vertices is included.
TWO	A boundary element is included if at least two of its vertices are included.
THRE	A boundary element is included if at least two of its vertices are included.
FOUR	A boundary element is included if at least two of its vertices are included

N1 If the file input mode is activated, then, **N1** denotes the number of vertices to be read from the file. If the file input mode is not activated, then **N1, …. Nn** are described below.

N1,..,Nn In the absence of the **SEQU**ENCE modifier, these numbers identify the vertices that define the boundary of interest. In the presence of the **SEQU**ENCE modifier only **N1**, **N2** and **N3** 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.

REMARKS

The complementary operation cannot be performed on regions that are defined by **MATC**HED **PAIR** or **CORR**ELATED type of element subdomains or on previous **LOCA**TE commands defined with a **FIEL**D modifier.

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 16: **Boundary Specification by Exclusion of Previously Specified Boundary**

SYNTAX **LOCA { BOUN } { UNSP } [ID=idsub]**

BOUN A boundary subregion is selected from the total external boundary of the domain by excluding the surfaces defined by all the previous LOCATE commands which identify a boundary (**LOCA**TE **PAIR**, **LOCA**TE **VERT**ex, **LOCA**TE **COOR**DINATE **BOUN**DARY and **LOCA**TE **POLY**GON **BOUN**DARY commands). **This modifier must be present to activate this mode of the command.**

UNSP Selects the previously unspecified boundary. **This modifier must be present to activate this mode of the command.**

idsub See Mode 1 specification.

EXAMPLES

LOCAte previously UNSPecified BOUNdary
LOCAte previously UNSPecified BOUNdary as ID=BND_UNSPecified

MODE 17: **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 can not 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 7.2.5 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 7.2.2) must be specified.

EXAMPLES

LOCAte COMPlimentary ID=DMN1 as ID=DMN2
LOCAte COMPlimentary of X- direction of ID=DMN1 ID=DMN2

MODE 18: **Specification of a General Second Order Surface**

SYNTAX **LOCA { SURF} { BOUN } {N1, N2, , Nn} [TOLE] {Nn+1} [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 that consists of the element nearest to the surface and the element boundary in the vicinity of the surface.

BOUN If this modifier is present then only the external boundary nodes are searched to satisfy the surface constraint. Otherwise all the element boundaries are searched.

N1,..,Nn These specify the coefficients C_0 through C_5 (for 2D) or C_0 through C_9 (for 3D). Any trailing coefficients not specified are assumed to be zero. At a minimum C_0 through C_2 (for 2D) or C_0 through C_3 (for 3D) must be specified.

TOLE 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.

Nn+1 The tolerance for judging the proximity to the surface. This numerical value must be present if the modifier **TOLE**RANCE is specified. The default value is 10^{-7}.

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 7.2.2) 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 19: **Matched Surface Subregion from Specified List**

SYNTAX **LOCA {MATC} [ID=idsub] [fname] { N1 | N1,…,N4, N5, … , Nn}**

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 (**N1**). 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 7.2.2) must be specified.

N1 If the file input mode is activated, then, **N1** 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 **N1,…,N4** below. If the file input mode is not activated, then **N1** is interpreted as defined in **N1,…,N4** below.

N1,…,N4 A set consisting of 2 element and surface number pairs which define the 1st and 2nd surfaces to be matched. N1 and N2 define the 1st surface whereas N3 and N4 define the 2nd surface. The surfaces of the element are numbered from 1 to 4 in the 2D and 1 to 6 in the 3D mode. See Sections 7.2.3 and 7.2.4 for further details of the manner in which these surface numbers are assigned.

N5,..,Nn The matched sets of element and surface number pairs in the manner of **N1,…,N4** above.

EXAMPLES

LOCAte MATCh LIST of values: (23,1) & (57,2); (25,3) & (67,4); (27,3) & (69,5) ID=MATCHED
LOCAte MATCh LIST (23,1) & (57,2); (25,3) & (67,4); (27,3) & (69,5) ID=MATCHED on file='match.fil'
LOCAte MATCh LIST 3 sets from file 'match.loc'

MODE 20: **Matched Surface Subregion from Two Previously Defined Subregions.**

SYNTAX **LOCA {MATC} {ID=idsub1} [dir1] {ID=idsub2} [dir2] [INTE] [ID=idsub] [fname]**

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.

idsub1 The name of the 1st 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 a similar surface of subregion **idsub2**. See Section 7.2.5 for available choices. This input is not required if **idsub1** was defined by a **LOCA**TE **PAIR** command.

Idsub2 The name of the 2nd subregion to be operated upon. There is no default value; a valid name must be specified. However **idsub2** may be identical to **idsub1**, if required.

dir2 The orientation index of the subregion **idsub2** that points to the surface to be matched with a similar surface of subregion **idsub1**. See Section 7.2.5 for available choices. This input is not required if **idsub2** was defined by a **LOCA**TE **PAIR** command.

INTE 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 **INTE**RIOR modifier is present, then the interior surfaces of the **idsub1** and **idsub2** are matched with each other.

idsub See Mode 1 Specification.

fname 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'

MODE 21: **Subregion by Union or Intersection of Two Previous Subregions**

SYNTAX **LOCA {UNIO | INTE} {ID=idsub1} {ID=idsub2} [NOT] [EXCL] [ID=idsub] [FIEL] [fname]**

UNIO The new subregion is created from the union (elements that belong to either) of two previously specified subregions. There is no default value. Either this or the **INTE**RSECTION modifier must be present to invoke this mode of the command.

INTE The new subregion is created from the intersection (elements that belong to both) of two previously specified.

idsub1 The identification or name of the 1st subregion to be operated upon. There is no default value; a valid name must be specified.

Idsub2 The identification or name of the 2nd subregion to be operated upon. There is no default value; a valid name must be specified.

NOT This modifier selects the union of the two subregions minus their intersection (elements that belong to one but not both). This modifier is effective only if the **INTE**RSECTION modifier is also specified.

EXCL If the modifier is present, then the specified elements 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.

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

The union and intersection operations cannot be performed on regions that are defined by MATCHED PAIR or CORRELATED type of element subdomains. If one or both the subdomains are of the surface (see, e.g., LIST PAIR option) type of subdomains, then the resulting subdomain is also of the LIST PAIR type unless the EXCLUDE modifier is present. The EXCLUDE modifier always generates a list of elements. If both subdomains are LIST PAIR type of subdomains and a particular element occurs in both, then the surface number for the second one is selected.

EXAMPLES

LOCAte UNIOn of ID=DMN1 and ID=DMN2
LOCAte UNIOn of ID=DMN1 and ID=DMN2 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=DMNM12
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 22: **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

LOCAte ID=DMN1 as new subdomain called ID=DMN2
LOCAte EXCLude ID=DMN1 and define new complimentary set as ID=COMPDMN1
LOCAte EXCLude ID=DMN1 and define new complimentary FIELd only set as ID=COMPDMN1
LOCAte EXCLude ID=DMN1 and define new complimentary FIELd only set as active subregion

MODE 23: **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 1st 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 subdomain 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.

COMMENTS

This command can operate on either the element type of subdomains (such as **LOCA**TE **LIST** command) or boundary type of subdomains (such as **LOCA**TE **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 **BOUN**DARY 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 subdomain unless the **FIEL**D modifier is also specified on the command.

EXAMPLES

LOCAte SOLId elements ID=DMN1
LOCAte FLUId elements of ID=DMN1 as ID=DMN2
LOCAte UNIQue elements ID=DMN1
LOCAte UNIQue boundary segments of ID=BOUN1 as ID=BOUN2

MODE 24: **Specification of Location of Injected Films and Curtains**

SYNTAX **LOCA {INJE} [ID=idsub1] [dir] {N1, …, Nn} [ID=idsub] [fname]**

idsub1 The identification or name of the subregion to be operated upon. See Sections 7.2.3 and 7.2.4. 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 "**CURT**AIN mode" (see **INJEC**TION **FILM** and **INJEC**TION **CURT**AIN commands).

dir The orientation index for the boundary of the subregion to be selected. See Section 7.2.5 for available choices. If no "**dir**" is specified, all available "dir" for **idsub1** are selected.

N1, .., Nn **N1** through **Nn** 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 7.2.3 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.

COMMENTS

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 **N1, …, Nn** 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 $\overrightarrow{AB} \wedge \left(\overrightarrow{AB} \wedge \vec{g} \right)$.

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)

COMMAND MATERIAL

PURPOSE Define material types and properties.

MODE 1: Material Type for a Subregion

SYNTAX MATE [TYPE] { N_{Mat} } [subrgn] [FIEL] [dir]

TYPE Identifies a material type with unique properties. If this modifier is omitted and more than 4 numerical values are specified, then the input is taken to be for material type.

N_{Mat} A number that designates the material type or zone. A distinct number should designate each different material that has its own unique properties. The default value is 1.

For **PORFLOW™,** by default, the maximum assigned material type is limited to 100. If mare than 100 material types are required, then the **ALLOCATE MATE**RIAL 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 **TIDAL™** 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.

subrgn The subregion for which the input is specified. If no subregion is specified, then the entire computational domain is selected.

FIEL Unless the **subrgn** was specified by a **LOCATE** command with the **FIEL**D modifier, the values are set at all nodes in the subregion **and** any nodes at the **exterior** boundary that are right next to the **subrgn** (see the **LOCATE** command). If this modifier is present, then only the interior field nodes are set by the command

dir By default, the input is applied to all the elements or nodes in the subregion defined by the **subrgn** 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 7.2.3 in Section 7.2.5 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.

COMMENTS

For **PORFLOW™** the material type is used by a number of properties commands (such as **HYDRAULIC**, **THERMAL**, **TRANSPORT**, etc.) that are identified with a material type. Some of these properties, especially if they are functions of space, time or other variables, may also be directly set by the **SET** command. For **TIDAL™** the material designation is used to impose land or water boundary conditions for water height and velocity components.

EXAMPLES

MATErial type 1 ! total domain
MATErial type 5 as the currently SELECTed subregion
MATErial type 3 as subregion ID=COARSE_SAND
MATErial type 5 for only the Y+ boundary of ID=GRAVEL
MATErial type 6 for only the FIELD nodes of ID=GRAVEL

MODE 2: **Material Type for a Rectangular Window**

SYNTAX **MATE [TYPE] [COOR] { N_{Mat} } [N1, ..., Nn]**

TYPE Identifies a material type with unique properties. If this modifier is omitted and more than 4 numerical values are specified, then the input is taken to be for material type.

N_{Mat} A number that designates the material type or zone. A distinct number should designate each different material that has its own unique properties. The default value is 1.

For **PORFLOW™** the maximum assigned material type is limited to 100 by default. If mare than 100 material types require to be defined, then the **ALLOCATE MATE**RIAL command must be used to specify the maximum number. Though the material numbers may be assigned in any arbitrary sequence, most efficient use of memory results if the numbers are defined sequentially.

For **TIDAL™** 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.

COOR By default, **N1** through **Nn** are assumed to specify the subregion in terms of grid index coordinates (I,J,K). However, if this modifier is present, it is assumed that the **N1** through **Nn** specify the actual grid coordinate (x, y, z) values for the window. If the **COOR** modifier is specified then an element is selected if its node (x, y, z) lies in the window. No consideration is given to the fact that significant or major part of an element may lie outside the specified window. By default, the node for an element is automatically located at the geometric center that is determined from the vertices of the element. However, this location can be overwritten by the user (see **COOR**DINATE command).

N1, ..., Nn The grid indices (I,J,K) or coordinates (x, y, z) of the material type or zone. See Section 7.2.3. If this input is omitted, then the subregion is defined to be the total domain. Generally 4 values must be specified for 2D and 6 for 3D geometry. If the modifier **TYPE** is present then a single element can be specified by 2 values in 2D and 3 in 3D mode.

APPLICABILITY

This command is currently used only by the **PORFLOW™** and **TIDAL™** Software Tools; it is not used by the **ANSWER™** Software Tool.

EXAMPLES

MATErial TYPE 1 ! total domain
MATErial TYPE 3 from (1,1) to (11,7) ! 2D input mode
MATErial TYPE 5 from (1,1,1) to (11,7,5) ! 3D input mode
MATErial TYPE 5 with COORdinates from (0., 0., 0.) to (100., 20., 2.)

MODE 3: **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 7.2.2 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

MATErial type information from 'TYPE.DAT'

MODE 4: Specification of a Fracture or Borehole

SYNTAX MATE {FRAC | BORE} [TYPE] { N_{Mat} } {N1, ..., Nn} [COOR] {Nn+1} [Nn+2]

FRAC The material type designation is for a fracture. This feature is only available with the structured grid mode of **PORFLOW™**.

BORE The material type designation is for a borehole. This feature is only available with the structured grid mode of **PORFLOW™**.

TYPE See Mode 1 specification.

N_{Mat} See Mode 1 specification.

N1, ..., Nn The grid indices (I,J,K) or coordinates (x, y, z) of the material type or zone. See Section 7.2.3. If this input is omitted, then the subregion is defined to be the total domain. A total of 4 values **must** be specified for 2D and 6 for 3D geometry.

COOR See Mode 1 specification.

Nn+1 Width of the planar feature if the modifier **FRAC** is present; outer diameter of the linear feature if the modifier **BORE** is present. There is no default value; a value must be specified.

Nn+2 Inner diameter of the linear feature if the modifier **BORE** is present. By default it is assumed that the inner diameter is zero.

APPLICABILITY

This command is currently available only for the **PORFLOW™** Software Tool.

COMMENTS

Each fracture (planar feature) or borehole (linear feature) must be specified by a different MATERIAL command. A fracture that is not oriented along one of the principal axes may be specified by its components in a stair-step manner. In the 3D mode, a fracture may be in the xy (N4=N7), yz (N2=N5), or zx plane (N3=N6). Similarly, a borehole may be in the x (N3=N6, N4=N7), y (N2=N5, N4=N7), or z (N2=N5, N3=N6) direction. In the 2D mode, only linear features may be present; they may be in the x (N3=N5) or y (N2=N4) direction. The cross-sectional area of the linear element, with D and d , respectively, as the outer and inner diameters, is calculated as:

$$\text{Area} = (\pi/4)\ (D^2 - d^2)\ ,$$

EXAMPLES

MATErial 3 from (6,10) to (31,10) BORE hole in X-dir dia = 0.2
MATErial type 4 from (6,10) to (6,19) BORE in Y: D=0.2, Inner d=0.1
MATErial type 3 from (6,10,2) to (6,10,7) BOREhole in Z dia = 0.1
MATErial type 3 from (6,10,2) to (31,10,7) FRACture in XZ width = 0.1
MATErial 5 FRACture in YZ: COOR (0.5, 0.7, 1.35) (0.9, 0.7, 1.45) W=0.05

MODE 5: **Specification of Material Density**

SYNTAX **MATE {DENS} {N1}**

DENS The material density is specified.

N1 The density of the dry solid component, ρ_s (> 0), of the porous material. The default value is 1.

APPLICABILITY

This command is currently available only for the **PORFLOW™** Software Tool.

COMMENTS

This mode of the command works in conjunction with the FOR command. It is effective for the Material Types specified by the previous, most recent, **FOR** command. If no **FOR** command was given, then the command is applied to Material Type 1. The material types can be specified by the **MATE**RIAL (Mode 1) and **ZONE** commands.

EXAMPLES

MATErial density = 1
MATErial density is 2200 kg/m**3

MODE 6: Specification of Material Porosity

SYNTAX MATE {PORO} [OVER] {N1} [N2, N3]

PORO The material porosity is specified.

OVER By default, an internal check is performed to ensure that all porosity values are between 0 and 1 (inclusive). If the values are outside of this range, the values are truncated to be within the range. However, if the modifier **OVER**RIDE is present, then no internal check is made.

N1 The effective (or flow) porosity, Θ_E. The default value is 1.

N2 The total porosity, Θ_T. If no value is specified, total porosity is set equal to the effective porosity.

N3 The diffusional porosity, Θ_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.

COMMENTS

This mode of the command works in conjunction with the FOR command. It is effective for the Material Types specified by the previous, most recent, **FOR** command. If no **FOR** command was given, then the command is applied to Material Type 1. The material types can be specified by the **MATE**RIAL (Mode 1) and **ZONE** commands.

EXAMPLES

MATErial porosity: 0.35
MATErial porosities: effective = 0.1, total 0.2
MATErial porosities: effective = 0.1, total 0.2, diffusive 0.15
MATErial porosities: 3*0.15
MATErial porosities: 0.2, 0.25, 0.21
MATErial porosities are 0.45, 1.50, 1.25 OVERride internal check

MODE 7: **Specification of Material Density and Porosity**

SYNTAX **MATE [PROP] {N1} [N2, N3, N4]**

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.

N1 The density of a dry, solid component, ρ_s (> 0). The default value is 1.

N2 The effective (or flow) porosity, Θ_E. The default value is 1.

N3 The total porosity, Θ_T. If no value is specified, total porosity is set equal to the effective porosity.

N4 The diffusional porosity, Θ_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.

COMMENTS

This mode of the command works in conjunction with the FOR command. It is effective for the Material Types specified by the previous, most recent, **FOR** command. If no **FOR** command was given, then the command is applied to Material Type 1. The material types can be specified by the **MATE**RIAL (Mode 1) and **ZONE** commands.

This mode of the **MATE**RIAL command is included primarily to provide compatibility of the data sets prepared for previous versions of PORFLOW™. The input for the density and porosity can now be provided separately by the Mode 3 and Mode 4 of the command.

EXAMPLES

MATErial density = 1; porosities: effective = 0.1, total 0.2, diffusive 0.15
MATErial density 2200, porosities: 3*0.15
MATErial density 2200, porosities: 0.2, 0.25, 0.21

MODE 8: **Specification of Material Tortuosity**

SYNTAX **MATE {TORT} [N1, N2, N3]**

TORT The input is specified for the tortuosity factor.

N1,..., N3 The x, y and z direction components, respectively, of the tortuosity factor (τ_{ij} of Equation 2.2.8 and 2.3.9) of the porous matrix. Only components in the principal directions are used. The numerical values must be between 0 and 1. The default value is 1.

APPLICABILITY

This command is currently available only for the **PORFLOW™** Software Tool.

EXAMPLES

MATE TORTuosity factors: 0.7, 0.6, 0.9

COMMAND MATRIX

PURPOSE To select the method of solution for the matrix of algebraic equations.

MODE 1: Built-In Matrix Solvers

SYNTAX MATR [Φ=N1, Φ =N2, ..., Φ =Nn] [dir] [option] [auxiliary]

Φ One or more symbols that denote the variable(s) for which the N1, N2, ..., etc. and the specified option are effective. Valid symbols are listed in Table 6.7.1.

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. **This input is effective only for the ADI method.** By default, the matrix is swept in all active coordinate directions.

option

- **ADI** Matrix is solved by an alternating direction implicit method. This is the default option.
- **SOR** Matrix is solved by implicit successive over relaxation where values from the current iteration are used where available.
- **EXPL** Matrix is solved by explicit successive over relaxation where only the values from the old iteration are used.
- **USER** Matrix is solved by an algorithm supplied by the user. See Section 7.2.9.

auxiliary

- **REDB** The matrix is analyzed for red/black split and the results are saved on a file called 'acr_REDBLACK.TMP'. This option is only activated if at least one variable is solved by the ACRi Unstructured ADI Algorithm.
- **ELEM** By default the red/black split is performed row-wise where each row is composed of a string of adjoining elements. This modifier causes the red/black split to be performed element-by-element.
- **OLD** By default a boundary-searching algorithm is used to form a row from a string of elements. This modifier causes an older element-based algorithm to be used.

N1,..., Nn Number of times the matrix is "swept" for the variable denoted by the symbol immediately preceding the value. **This input is effective only for the ADI and implicit SOR methods.** The default value may vary with each installation though it is generally set to 1 for most of the variables.

COMMENTS

Each "sweep" through the matrix consists of one pass through the matrix of equations. Each 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 an "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 (**CONV**ERGENCE 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 direction only
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
MATRix for P to be solved by SOR
MATRix for T to be solved 3 times by the ADI method with OLD row forming algorithm
MATRix REDBlack file to be generated for ELEMent based split
MATRix for T ADI; REDBlack file for OLD method
MATRix for T to be solved by EXPLicit SOR
MATRix for T USER option 5 ! Matrix Solver supplied by User

MODE 2: **Number of Inner Matrix Iterations**

SYNTAX **MATR {ITER} { N1 | Φ=N1, Φ =N2, ..., Φ =Nn }**

ITER The input pertains to the maximum number of inner iterations for the matrix.

Φ Symbols that denote the variable(s) for which the N1, N2, ..., etc. are effective. Valid symbols are listed in Table 6.7.1. If no symbol is specified, **and only one numerical value is present**, then the input is applied to all variables for which a transport equation is solved.

N1,..., Nn Number of times (≥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.

COMMENTS

Once the matrix coefficients are assembled, the matrix is solved iteratively till the specified convergence criterion (**CONV**ERGENCE 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 (see Chapter 4.5). 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

MATRix ITERations for all variables = 10
MATRix ITERations: P=8, T=2, C=2

MODE 3: Matrix Solvers from the University of Texas NSPCG Package

SYNTAX MATR {NSPC} [Φ] [precon] [accel] {ELIM [OFF]} [option]

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 6.7.1.

precon Preconditioner component of the Matrix Solver.

precon	INTERPRETATION
NEUM	Neumann matrix polynomial. **This is the default option.**
LEAS	Least squares matrix polynomial.
DEGR	Degree of NEUMann or LEASt squares polynomial (integer) default: 3
REDU	Reduced System preconditioner.
CHOL	Incomplete Cholesky Factorization.
JACO	The point Jacobi preconditioner.
SOR	The Successive Over-Relaxation preconditioner and accelerator

accel Accelerator Component of the Matrix Solver

accel	INTERPRETATION
CONJ	Conjugate Gradient accelerator.
BCGS	The Biconjugate Gradient Squared accelerator.
GMRE	GMRES accelerator. **This is the default option.**
ORTH	ORTHOMIN accelerator.
CGNR	Conjugate Gradient applied to Normal Equations.
LANC	Lanczos with ORTHOMIN accelerator.

option Modifier for the specification or matrix or solver method

option	INTERPRETATION
MODI	If CHOLESKY preconditioner is selected, then modified incomplete Cholesky decomposition is used; otherwise this input is ignored.
PERM	If CHOLESKY preconditioner is selected, then matrix is red-black permuted; otherwise this input is ignored.

EXAMPLES

MATRix for P by NSPCg (permuted CHOLesky).
MATRix for P from NSPCG with JACObi preconditioner and BCGS accelerator
MATRix for P is SYMMetric; use NSPCG with JACObi and BCGS
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 ELIMination OFF

MODE 4: Matrix Solvers from the HYPRE Library, Lawrence Livermore National Laboratory

SYNTAX MATR {HYPR} [Φ] [precon] [accel] [option] [VECT= N_{Vector}]

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.

Φ One or more symbols that denote the variable(s) for which the specified options are effective. Valid symbols are listed in Table 6.7.1.

precon Preconditioner component of the Matrix Solver.

precon	INTERPRETATION
AMG	Algebraic Multi-Grid Preconditioner. **This is the default option.**
CHOL	Incomplete Cholesky Factorization..
DIAG	Diagonal Scaling Preconditioner
SPAR	The Sparse Approximate Inverse Preconditioner.

accel Accelerator Component of the Matrix Solver

accel	INTERPRETATION
AMGA	The AMG accelerator.
BICG	The BI-Conjugate Gradient accelerator
CONJ	Conjugate Gradient accelerator.
GMRE	GMRES accelerator. **This is the default option.**

option Modifier for the specification of matrix or solver method

option	INTERPRETATION
SYMM	**By default the matrix is assumed to be non-symmetric.** If this modifier is present, then the matrix is assumed to be symmetric.
INDE	**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 **INTE**GRATION Command)
STAT	**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 **STAT**ic 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_{Vector} 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.

EXAMPLES

MATRix by HYPRe solver
MATRix for P from HYPRe solver print detailed DIAGnostics
MATRix for P from HYPRE with AMG preconditioner and CONJ accelerator and AMORtized set up
MATRix for P is SYMMetric; use HYPRe with CHOLesky with CONJugate Gradient
MATRix U V W use HYPRE: SPARse preconditioner with GMRES accelerator
MATRix U V W use HYPRE CHOLesky with GMRES and 10 VECTors

MODE 5: Global Matrix Control Parameters

SYNTAX MATR [COEF=V_{Coef}] [ZERO= V_{Zero}]

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_{Coef} The minimum threshold for any of the diagonal coffiecients 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 **NSPC**G and **HYPR**E 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_{Zero} The numerical value for the machine zero. The default values are given in the table below.

1.192×10^{-7}	IEEE real with 32 bit precision.
2.22×10^{-16}	IEEE real with 64 bit precision. **This is the default.**
7.1×10^{-15}	Cray XMP.
1.49×10^{-8}	Dec 10 (single precision)
4.768×10^{-7}	IBM 370 / 158 (single precision)

EXAMPLES

MATRix COEFficient minimum value = 1.E-30
MATRix machine ZERO =1.0E-300
! Above is useful if NSPCG thinks that the ||RHS|| < machine zero and returns prematurely.
MATRix machine ZERO set to 1.0E-300, COEFicient=1.E-30.

MODE 6: **Tolerance Threshold for Minimum Value of Matrix Elements**

SYNTAX **MATR {TOLE} { V_{Tole} | $\Phi=V_{Tole1}$, $\Phi=V_{Tole2}$, ..., $\Phi=V_{Tolen}$ }**

TOLE The tolerance or threshold for the minimum value of matrix elements. Matrix elements with numerical values below the specified tolerance are dropped. For the **AMG** preconditioner in the **HYPR**E solver, this modifier controls the connections between matrix elements in constructing a coarse level.

V_{Tole} The numerical value for the threshold. For the **SPAR** preconditioner, the typical values are between 0.01 and 0.1 and the default is set at 0.05. For the **CHOL**ESKY 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.

Φ Symbols that denote the variable(s) for which the N1, N2, ..., etc. are effective. Valid symbols are listed in Table 6.7.1. If no symbol is specified, **and only one numerical value is present**, then the input is applied to all variables for which the **CHOL**ESKY , **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).

EXAMPLES

MATRix for P from HYPRE with AMG preconditioner and CG accelerator
MATRix for U from HYPRE with SPAI preconditioner and GMREs accelerato
MATRix for V and W from HYPRE with CHOLesky preconditioner and GMREs accelerator
MATRIx TOLErance for P=0.25, for U=0.1, for V=0.001 and for W=0.01

MODE 7: **Number of levels for AMG and SPAR Preconditioners for HYPRE**

SYNTAX **MATR {LEVE} { N_{Level} | $\Phi=N_{Level1}$, $\Phi=N_{Level2}$, ..., $\Phi=N_{Leven}$ }**

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_{Level} 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_{Level}** numerical values are effective. Valid symbols are listed in Table 6.7.1. 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).

EXAMPLES

MATRix LEVELS = 10 for all relevant solvers
MATRix LEVELS U = 5, V=5, P=10 for

MODE 8: **Maximum Number of Non-zero Elements for Any Row of Modified Matrix**

SYNTAX **MATR {ELEM} { N_{Elem} | $\Phi=N_{Elem1}$, $\Phi=N_{Elem2}$, ..., $\Phi=N_{Elemn}$ }**

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 **CHOL**ESKY Preconditioner of the **HYPR**E package.

N_{Elem} The maximum number of non-zero elements that in any row of the modified matrix. Typical values are between 15 and 25 and the default is set at 15.

Φ Symbols that denote the variable(s) for which the N1, N2, ..., etc. are effective. Valid symbols are listed in Table 6.7.1. If no symbol is specified, **and only one numerical value is present**, then the input is applied to all variables for which the **CHOL**ESKY, method has been selected.

EXAMPLES

MATRix maximum non-zero elements = 10 for all solvers
MATRIx non-zero ELEMents for P=15 and for W=5

MODE 9: **Maximum Number of Non-zero Elements for Any Row of Modified Matrix**

SYNTAX **MATR {RESI} {ABSO}**

RESI For the **NSPC**G solvers, by default the specified convergence tolerance (**CONV**ERGENCE 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.

ABSO The modifier to select the absolute measure of convergence tolerance.

EXAMPLES

MATRix RESIdue in ABSOlute mode

MODE 10: **Matrix Diagnostic Output for NSPCG and HYPRE Solvers**

SYNTAX **MATR [DIAG] [SUMM] [OFF]**

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.

COMMENTS

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

MATRIx DIAGnostic output to be generated
MATRIx DIAGnostic SUMMary form output to be generated
MATRIx DIAGnostic output OFF from now on

COMMAND MOVE

PURPOSE To specify a moving coordinate system for the system geometry.

MODE 1: Grid Coordinate Frame in Uniform Translation

SYNTAX MOVE {dir} { V_{grid}} [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 along which the coordinate system is moving at a uniform speed. There is no default value; a value must be specified.

V_{grid} 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 is 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.

COMMENTS

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 seconds 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:

$$u = -\omega y; \qquad v = \omega x$$

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.

COMMENTS

The **ABSO**LUTE velocity mode is recommended for most applications. In general it depicts better numerical behavior. The **RELA**TIVE 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.

COMMAND NOZZLE

PURPOSE To specify properties and parameters relating to injection of liquid droplets through nozzle.

MODE 1: Location and Properties for Individual Droplets

SYNTAX NOZZ {DROP} [FUEL|WATE] {Q_m} [option] [[ABSO] RATE=Q_{Evap}] [STAR=t_{Start}] [STOP=t_{Stop}[[STOR=N_{Stor}] [TEMP=T_{NZL}] [FREQ=N_{Frq}] {[SETS]=N_{Sets}} {datatype [ELEM] } [UNIF] {fname | N1, ,Nn}

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_m The total mass flow rate [M T^{-1}] 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
DISA	The specified option(s) listed below is (are) deactivated
EVAP	The droplets evaporate during move. The option is active by default.
DRAG	The droplets undergo drag during move. The option is active by default.
OUTP	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_m but before the actual droplet information (**N1, ,Nn**). If **SETS** modifier is missing, then it must occur before the N_{Sets} value.

Q_{Evap} The rate of evaporation as a fraction of the droplet mass. In the presence of the **ABSO**LUTE modifier, the specified fraction of the original droplet is forced to evaporate at each droplet time step (see **NOZZ**LE 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 **ABSO**LUTE 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_m but before the actual droplet information (**N1, ,Nn**). If **SETS** modifier is missing, then it must occur before the N_{Sets} value.

T_{Start} 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_m but before the actual droplet information (**N1, ,Nn**). If **SETS** modifier is missing, then it must occur before the N_{Sets} value.

T_{Stop} 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_{Set} and N_{Frq}) 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_m but before the actual droplet information (**N1, ,Nn**). If **SETS** modifier is missing, then it must occur before the N_{Sets} value.

N_{Stor} 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_m but before the actual droplet information (**N1, ,Nn**). If **SETS** modifier is missing, then it must occur before the N_{Sets} value.

T_{NZL} 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_{Frq}) for processing the droplets data. This modifier and the corresponding numerical value, if present, may be placed anywhere after Q_m but before the actual droplet information (**N1, ,Nn**). If **SETS** modifier is missing, then it must occur before the N_{Sets} value.

N_{Frq} 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_{Sets}) 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_m but before the actual droplet information (**N1, ,Nn**)

N_{Sets} 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 be 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
X	The X-coordinate location of the droplet is specified
Y	The Y-coordinate location of the droplet is specified
Z	The Z-coordinate location of the droplet is specified
M	Element number for the location of the droplet is specified
I	Grid index I for the location of the droplet is specified
J	Grid index J for the location of the droplet is specified
K	Grid index K for the location of the droplet is specified
N	The corresponding field is not processed
Skip	The corresponding field is not processed
Dia	The diameter of the droplet, in microns, is specified
Φ	The property denoted by the symbol Φ is specified. Any of the properties listed in Table 6.8.1 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_{Set} sets is assumed to be the same. The number of droplets in the n^{th} set, N_n, is computed as:

$$N_n = \frac{Q_m}{\rho_l \sum_{n=1}^{N_{Set}} \frac{\pi}{6} d_n^3}$$

Here **Q_m** is the injected mass, **N_{Set}** is the number of sets, ρ_l is the liquid density and d_n is the diameter of the n^{th} set. If this modifier is not present, then the fuel is equally distributed in the **N_{Set}** sets and the number of droplets in the n^{th} 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_{Set}** records. Each record must have then same number of values in the same order as the **datatype** symbols on the command.

N1,.., Nn The data for the droplets if a file name is not specified. **N_{Set}** 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 ! 1st & 6th 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 ! 1st, 5th, 6th, 9th & 11th 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 ! 1st & 6th columns skipped; RHO ignored.
NOZZle DROPlet 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 ! 1st & 6th columns skipped; RHO ignored.

MODE 2: Nozzle Location and Injection Parameters (3D Geometry)

SYNTAX NOZZ {LOCA|DEGR} [FUEL|WATE] { X_o, Y_o, Z_o} {Q_m} {Rad, Φ} { n_X, n_Y, n_Z, } {Θ_o, Θ_F} {d_{SMD}} {N_{Ray}} [option] [[ABSO] RATE=Q_{Evap}] [STAR=t_{Start}] [STOP=t_{Stop}[[STOR=N_{Stor}] [TEMP=T_{NZL}] {N_{Vel} } {N1,..Nn}

LOCA The location of the nozzle discharge flow is specified. Individual droplets are internally generated.

DEGR 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.

FUEL Nozzle injects fuel droplets. This is the default option.

WATE Nozzle injects water droplets.

X_o, Y_o, Z_o The x, y, and z coordinates of the location of the nozzle. If the geometry is radial, then **Z_o** must be in radians or degrees. There is no default value.

Q_m The mass flow rate injected through the nozzle. There is no default value.

Rad The nozzle radius. There is no default value.

Φ The nozzle cone angle. There is no default value.

n_X, n_Y, n_Z Direction cosines of nozzle orientation in the X, Y, and Z-directions, respectively). There is no default value.

Θ_o, Θ_F Initial and final spray cone angles. There is no default value.

d_{SMD} Sauter mean diameter (SMD) in microns. The actual distribution of droplet diameter for each ray (see **N_{Ray}** below) is controlled by the **NOZZ**LE command with **DIST**RIBUTION modifier. There is no default value.

N_{Ray} 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 **NOZZ**LE command with **DIST**RIBUTION 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 **N_{Vel}** values of spray velocity distribution.

option Same as Mode 1.

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 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 **N_{Ray}** but before the location of **N_{Vel}**.

Q_{Evap} The rate of evaporation as a fraction of the droplet mass. In the presence of the **ABSO**LUTE 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 **ABSO**LUTE 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 N_{Ray} but before the location of N_{Vel}.

T_{Start} The starting time for the discharge of droplets from the nozzle if **STAR**T 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_{Ray} but before the location of N_{Vel}.

T_{Stop} 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 be 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_{Set} and N_{Frq}) 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_{Ray} but before the location of N_{Vel}.

N_{Stor} 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_{Stor} 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_{Ray} but before the location of N_{Vel}.

T_{NZL} 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_{Vel} The number of values ($\geq$**1**) in the spray velocity distribution. There is no default value.

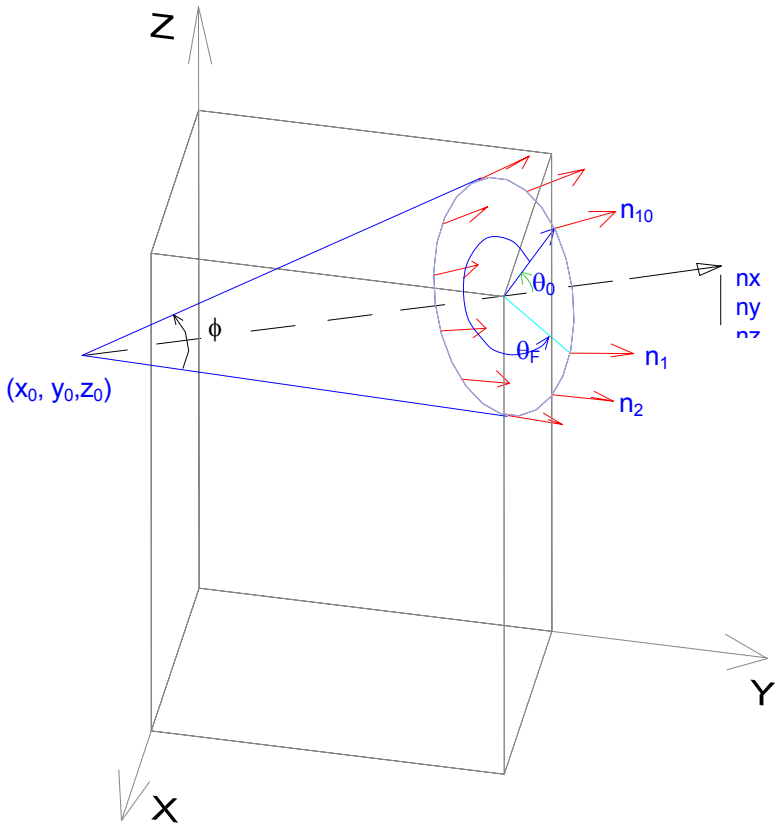
N1,.., Nn Spray velocity distribution. These values are sampled in succession for each sub-ray. The specified number of values must equal N_{Vel}; **at least one value must be specified.**

COMMENTS

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 **NOZZ**LE **DISTR**IBUTION 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 **NOZZ**LE **DISTR**IBUTION command and 10 rays (N_{Ray}) 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_{SMD}) 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 1/70th of the total injected fuel (Q_m).

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_o, Θ_F) are 45 and 315, then the 1st through the 10th 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 ray that issues is determined by the N_{Vel} values of velocity distribution (**N1,.., Nn**) given on this command. If the specified velocity distribution contains, say, 5 values, then the injection velocity for all the droplets in the 1st ray is equal to the 1st value in the velocity distribution, that of the 2nd ray is the 2nd value, and so on up to the droplets in the 5th ray. Then the cycle repeats and the 6th droplet is assigned the 1st velocity value and so on.



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_o, Y_o} {Q_m} {H_0, Φ} { n_X, n_Y,} {d_{SMD}} {N_{Ray}} [option] [[ABSO] RATE=Q_{Evap}] [STAR=t_{Start}] [STOP=t_{Stop}[[STOR=N_{Stor}] [TEMP=T_{NZL}] {N_{Vel} } {N1,..Nn}

LOCA The location of the nozzle and flow parameters are specified. Individual droplets are internally generated.

DEGR Same as Mode 2.

FUEL Nozzle injects fuel droplets. This is the default option.

WATE Nozzle injects water droplets.

X_o, Y_o The x, and y coordinates of the location of the nozzle. There is no default value.

Q_m The mass flow rate injected through the nozzle. There is no default value.

H_0 The nozzle opening height in cross-section. There is no default value.

Φ The included angle of the spray cone. There is no default value.

n_X, n_Y Direction cosines of nozzle orientation in the X and Y directions, respectively). There is no default value.

d_{SMD} Same as Mode 2.

N_{Ray} Same as Mode 2.

option Same as Mode 2.

ABSO Same as Mode 2.

RATE Same as Mode 2.

Q_{Evap} Same as Mode 2.

STAR Same as Mode 2.

T_{Start} Same as Mode 2.

STOP Same as Mode 2.

T_{Stop} Same as Mode 2..

STOR Same as Mode 2.

N_{Stor} Same as Mode 2.

TEMP Same as Mode 2..

T_{NZL} Same as Mode 2.

N_{Vel} Same as Mode 2.

N1,.., Nn Same as Mode 2.

EXAMPLES

NOZZle DEGRees: (0.1,1.0) Q=0.1, R=0.005, cone=45 deg, dir cosines (0.707, 0.707), 40, 5,1,100
NOZZle 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_m } {N_{Sets}}**

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_m The total mass flow rate [M T^{-1}] of fuel injected by the nozzle. There is no default value.

N_{Sets} The number of individual droplets injected by the nozzle. There is no default value.

EXAMPLES

NOZZle ATOMization 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 **NOZZ**LE command with **LOCA**TION or **DEGR**EE 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 **NOZZ**LE command with **LOCA**TION or **DEGR**EE 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.

COMMENTS

This command applies ONLY to the Mode 2 or 3 NOZZLE commands that follow this command. Any Mode 2 or 3 **NOZZ**LE command that precedes this command will use the previously supplied (or default) droplet distribution. Thus each Mode 2 or 3 **NOZZ**LE command may have its own droplet distribution. 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, 2.0
NOZZle DISTribution with values: 0.1, 0.4, 0.6, 0.8, 1.0, 1.2, 1.6, 1.8, 2.0, 2.5

MODE 6: Nozzle Spray Computations Parameters

SYNTAX NOZZ { FREQ } { V_{frq} } [TIME] [N_{Begin}, N_{End}]

FREQ The spray computation frequency parameters are specified.

V_{frq} The frequency (step or time interval) at which the droplet computations are performed. The default value is 10.

TIME By default, V_{frq} is the frequency of computations in terms of number of steps. If this modifier is present, then V_{frq} is interpreted to be time interval between successive computations.

N_{Begin} The first computational step at which the droplet computations are performed. The default value is 1.

N_{End} The last computational step at which droplet computations are performed. The default frequency is set to 9999999.

COMMENTS

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.

EXAMPLES

NOZZle FREQuency of calculations: 1 start at step number 10
NOZZle FREQuency of calculations: 10 start at step number 10 and stop at step 500

MODE 7: Nozzle Output Options

SYNTAX NOZZ {TRAJ}] [fname] [N_{Start}, N_{End}, N_{Frq}]

TRAJ A detailed file of trajectory and evolving particulars for each droplet is generated. The frequency of output is controlled by N_{Start}, N_{End} and N_{Frq}. 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 7.2.2 for additional information. If no file name is specified, then the output is directed to the file named 'acr_NOZZLE.TMP'. If a new command with a new file name is specified, then any file opened by a previous command is closed and subsequent output is directed to the new file.

N_{Start} 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.

N_{End} 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.

N_{Frq} The frequency of output in terms of computational steps. If both N_{End} and N_{Frq} are missing, then the default value is set to 9999999 so that the output is obtained only once as determined by N_{Start}. If on the other hand, N_{End} is specified but N_{Frq} is missing, then the default value is set to unity so that the output is obtained at every step between N_{Start} and N_{End}).

EXAMPLES

NOZZle TRAJectory output the first time droplet computations are performed
NOZZle TRAJectory output on file 'TRAJ.NZL'
NOZZle TRAJectory output on file 'TRAJ.NZL' at step number 50
NOZZle TRAJectory 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=α_{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, **α_{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.

α_{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

NOZZle TIME 2.0 ! Multiply internally computed value by 2
NOZZle TIME STEP is = 1.E-6 seconds ! Force this time step for droplet Movement
NOZZle TIME multiplier is 2 and INCREASE factor is = 1.
NOZZle ITERations=200
NOZZle TIME STEP = 1.E-6, ITERations=200
NOZZle ITERations=200, TIME STEP = 1.E-6, INCRease factor 1.0

MODE 9: Droplet Temperature and Evaporation Factors

SYNTAX NOZZ [MINI = T_{min}] [MAXI= δT_{max}] [EVAP= δq_{max}] [RANZ] [FUEL] [WATE] [GRAV] [STOC] [AUTO]

MINI 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.

T_{min} The minimum droplet temperature in absolute units. The default value is 273.15.

MAXI 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.

δT_{max} The maximum change in temperature of the droplet as a fraction of current liquid temperature. The default value is 0.1.

EVAP This modifier sets the maximum fraction of the mass of the droplet that can evaporate in any single time step.

δq_{max} 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.

RANZ 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 **FUEL** nor **WATEr** modifier is present, then the correlation is used for all nozzles.

FUEL The Ranz-Marshall correlation is applied to the nozzles with fuel injection.

WATE The Ranz-Marshall correlation is applied to the nozzles with water injection.

GRAV 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.

STOC 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.

AUTO A fixed evaporation rate for droplets may be explicitly set by the Mode 1 of the command with **RATE** modifier. This **AUTO** 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

NOZZle MINImum temperature=300 degrees
NOZZle MINImum temperature=300 degrees; MAXImum temperature change=0.01
NOZZle MINImum temperature=300 degrees; Evaporated fraction in one step=0.01
NOZZle MINImum temp=300; AUTOmatic mode
NOZZles with GRAVitational and STOChastic forces and RANZ marshall correlation
NOZZles of WATER type with RANZ Marshall correlation
NOZZle switch back to AUTOmatic mode

COMMAND OPEN

PURPOSE To specify an open boundary through which fluid may enter or leave based upon prevailing flow conditions.

SYNTAX **OPEN {dir} [subrgn] [DIAG] [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.

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 open boundary for the identified subregion is deactivated. A new specification for this subregion may follow.

COMMENTS

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 **INIT**IAL, **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

COMMAND OPTION

PURPOSE To select or modify built-in default options.

SYNTAX OPTI [LINE | GRAD | FIXE | FIX] [P | PRES]
[ENTH | TEMP] [DIFF] [ARIT | SOLI | FLUI | OFF]
[MACH] [SONI | GAMM | GAS]
[OUTL] [OFF | LOCA | VELO]
[NEW | OLD] [QUAD | VOLU] [INTE] [ROBU]

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 2nd 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.

ENTH For **ANSWER™,** the thermal diffusion term is computed from the product of the effective thermal conductivity divided by the specific heat and the enthalpy gradient. When the specific heat is constant and the Lewis number is unity, this is equivalent to Fourier's law of Conduction. This leads to a very stable numerical formulation of the enthalpy equation and is accurate as long as the gradients of specific heat are much smaller than those of temperature. This is the default option if no conjugate heat transfer in the solid blocks (**SOLI**D command) occurs.

TEMP 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, it may be numerically less stable than that described by the **ENTHaply** modifier above if the gradients of specific heat are large or if enthalpy is not a linear function of temperature. This is the default option if conjugate heat transfer in the solid blocks (**SOLI**D command) is specified. This option is available only for ANSWER™ Software Tool.

DIFF In **ANSWER™** Software Tool, (when the **TEMP** option is active), the thermal conductivity is computed 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.

MACH This modifier selects the method used to compute the Mach number for the flow.

SONI The Mach number is computed from the relation::

$$Ma = \frac{V}{c}$$

Here V is the fluid speed and c is the speed of sound which can be specified with the **GAS SONI**C command. This is the default option unless **COMP**RESSIBLE or **REAC**TION command is specified. This modifier is effective only in the presence of the **MACH** modifier.

GAMM The Mach number is computed from the relation:

$$Ma = \frac{V}{\sqrt{(\gamma - 1) C_p T}}$$

Here V is the fluid speed, γ is the ratio of specific heat at constant pressure to that at constant volume, C_p is the specific heat at constant pressure and, T is the temperature in absolute units. The γ can be specified by the **GAS GAMM**A command. This modifier is effective only in the presence of the **MACH** modifier.

GAS The Mach number is computed from the relation:

$$Ma = \frac{V}{\sqrt{\frac{R}{C_p - R} C_p T}}$$

Here V is the fluid speed, R is the gas constant for the gas mixture, C_p is the specific heat at constant pressure and, T is the temperature in absolute units. The R is computed from the mass fraction and molecular weights of the gas components. This is the default option if **COMP**RESSIBLE or **REAC**TION command is specified. This modifier is effective only in the presence of the **MACH** modifier.

OUTL For all incompressible flow simulations and for steady state compressible flow simulations, the flow at **OUTL**ET 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 **OUTL**ET 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 **OUTL**ET modifier for the **ANSWER™** Software Tool.

LOCA The mass balance at each of the **OUTL**ET elements is corrected individually to ensure that the local outflow from the element equals the inflow to the element. This modifier is effective only in the presence of the **OUTL**ET modifier for the **ANSWER™** Software Tool. **The use of this modifier is not generally recommended.**

VELO By default the corrections relating to the **OUTL**ET 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 **OUTL**ET modifier for the **ANSWER™** Software Tool. **The use of this modifier is not generally recommended.**

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 in the presence of the **NEW** modifier.

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 in the presence of the **NEW** modifier.

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

OPTIon GRAD for pressure ! For extrapolation of pressure for this highly skewed grid
OPTIon use ENTHalpy ! Formulation for conjugate heat transfer
OPTIon use TEMPerature formulation for heat transfer even though there are no solid blocks.
OPTIon thermal DIFFusion with specific heat of SOLId at the interface
OPTIon NEW geometry computations with QUADratic interpolation

COMMAND OUTLET

PURPOSE To specify an outflow boundary for the domain of computation

SYNTAX OUTL { [subrgn], [dir] } [DIAG] [OFF] [AREA | N1]

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 7.2.5 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 COORdinate or LOCATE LIST command). Volume type of subregions specified with a LOCATE command with FIELd, STATion or CORRelation modifiers can not 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. There is no default value; a value must be specified.

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 outlet boundary for the identified subregion is deactivated. A new specification for this subregion may follow.

AREA For steady state or incompressible flow, the flow rate at each outlet port is adjusted so that the total outflow equals the total inflow **(See Comments below).**. If this modifier is present, then the outflow rate at each port is distributed in proportion to the ratio of its area to the total outflow area. If this modifier as well as the N1 specification are omitted, then the outflow is distributed in proportion to the computed outflow rate. **The default practice is equivalent to treating all the outlets as a single outlet.** This modifier is effective only if multiple OUTLET commands are specified.

N1 The flow rate or flow fraction that exits through this outlet as a fraction of the total outflow through all outlet ports **(See Comments below)**. This modifier is effective only if multiple OUTLET commands are specified.

COMMENTS

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. If multiple OUTLET commands are specified, then the flow rate through the i^{th} port, Q_I, is adjusted as:

$$Q_i = Q \frac{F_i}{\sum_i F_i}$$

Where Q is the total outflow and F_i is the fraction through the I^{th} port. If the **AREA** modifier is present, then F_i is set equal to the area of the port, if **N1** is specified, then it is set equal to **N1**. If neither **AREA** nor **N1** is

specified, then F_i is set equal to the computed flow (before any adjustment). **Though it is possible to specify, a mix of these options, it is recommended that a consistent practice be followed**. That the **AREA** modifier or **N1** be either specified for all or omitted for all **OUTL**ET commands.

EXAMPLES

OUTLet at X+ boundary at right (maximum x)
OUTLet at Y+ boundary for most recently SELEcted subregion
OUTLet port at Y+ for SELEcted subregion; outflow fraction = 0.5
OUTLet port at Y+ for SELEcted subregion; outflow fraction in proportion to AREA
OUTLet at X-: FIXEd pressure boundary for subregion ID = OUTFlow
OUTLet OFF at X- boundary for subregion ID = OUTFlow

COMMAND 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 [Φ] [plane|TABL] [ADD] [STAN|fname] [subrgn] [STAT|NOST] [NARR|WIDE] [V_{frq}] [TIME] [IMME|NOW|ONLY|OFF]

Φ **One or more** of the symbols that represent the variables for which output is desired. The valid symbols are listed in Table 6.8.1. 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 1st **SOLV**E 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 in a liner columnar table of values ordered by element numbers. This the default mode for unstructured grids.

ADD One **OUTP**UT command is activated by default. If the **ADD** modifier is present, then the specified command is added to the list of active **OUTP**UT 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 **OUTP**UT command with an **ADD** modifier, then this file is uniquely attached to a unit for output from that command and any subsequent **OUTP**UT 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 7.2.2 for additional information.

Output from **all OUTP**UT 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 **STAN**DARD modifier is subsequently specified, then the previous file is closed and all subsequent output from **all OUTP**UT 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 **LOCA**TE command). **If any other type of subregion is specified, then the TABL**E **mode (MODE 2) of the command is automatically invoked.** For **TABL**E type of output, any type of subregion can be specified. If no subregion is specified, the entire computational domain is selected. See Sections 7.2.3 and 7.2.4.

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.

NARR The output tables are produced in an 80-column (narrow) format.

WIDE The output tables are produced in a 132-column (wide) format. This is the default mode.

V_{frq} The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 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.

TIME 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.

IMME The output is produced immediately. This modifier is necessary only before the 1st **SOLV**E command is given. It then enables the output of the current value of specified variable(s) immediately. After the 1st **SOLV**E 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 1st **SOLV**E command.

NOW If this modifier is specified before the 1st **SOLV**E command, then the output is produced just before the 1st 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 1st **SOLV**E 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 in response to a **NOW** or **IMME**DIATE modifier or at the frequency determined by **Vfrq**.

OFF Output for any previous command(s) for the same **subrgn** and type (**plane** or **TABL**E) is subsequently suppressed.

COMMENTS

If no **OUTP**UT 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 a 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

MODE 2: **Output of Properties and Coefficients for the Differential Equations**

SYNTAX **OUTP {Φ} { [COND| DIFF] [STOR] [SOUR] [MATR] [RESI] } [plane|TABL] [ADD] [STAN | fname] [subrgn] [STAT|NOST] [NARR|WIDE] [V_{frq}] [TIME] [IMME|NOW|ONLY|OFF]**

Φ **One, and only one,** of the symbols that denotes the dependent variable for which a differential equation is solved. The valid symbols are listed in Table 6.7.1. There is no default value; a valid symbol must be specified.

COND 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.

DIFF Same as **COND**.

STOR The storage coefficient for the variable is printed to the output file.

SOUR The net source (algebraic sum of source and sink) for the variable is printed to the output file.

MATR 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.

RESI 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.

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 The name of the file to which the output is directed. See Section 7.2.2 for additional information. Since the **ADD** modifier is assumed by default, the filename, if specified, is considered unique for the command and any subsequent **OUTPUT** command that specifies the **same file name**.

plane See Mode 1 specification.

subrgn See Mode 1 specification.

NARR See Mode 1 specification.

WIDE See Mode 1 specification.

V_{frq} See Mode 1 specification.

TIME See Mode 1 specification.

IMME The output is produced at the start of of computations. In this mode of input, this modifier is equivalent to the **NOW** modifier because the coefficients and components of the transport equation are not available till the1st **SOLVE** command is given and the computation process has started.

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 **TABL**E) and same combination of **COND, DIFF, STOR, SOUR, MATR,** and **RESI** modifiers is subsequently suppressed.

COMMENTS

This mode of the **OUTP**UT 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.

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

MODE 3: **Output of Convective, Diffusive and Total Flux for each Element**

SYNTAX **OUTP {Φ} {FLUX} [CONV] [DIFF] [TOTA] [AREA] [plane|TABL] [ADD] [STAN | fname] [subrgn] [STAT|NOST] [NARR|WIDE] [V_{frq}] [TIME] [IMME|NOW|ONLY|OFF]**

Φ **One, and only one,** of the symbols that denotes the dependent variable for which a differential equation is solved. The valid symbols are listed in Table 6.7.1. There is no default value; a valid symbol must be specified.

FLUX 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.

CONV The convective flux for each face of the elements of the computational domain is printed.

DIFF The diffusive flux for each face of the elements of the computational domain is printed.

TOTA The sum total of the convective and diffusive fluxes for each face of the elements of the computational domain is printed. This modifier is assumed by default if the **CONV** or **DIFF** modifiers are not present.

AREA The computed flux is divided by the projected area of the face for output.

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.

V_{frq} See Mode 1 specification.

TIME See Mode 1 specification.

IMME See Mode 2 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 **TABL**E) and same combination of **FLUX, CONV, DIFF** and **TOTA**L modifiers is subsequently suppressed.

COMMENTS

This mode of the **OUTP**UT 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.

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] [NARR|WIDE] [V_{frq}] [TIME] [IMME|NOW|ONLY|OFF]**

GRAD The tensor components of the gradients of velocity, φ_{ij} are output, where:

$$\varphi_{ij} = \frac{\partial u_i}{\partial x_j}$$

Here u_i is the i^{th} component of velocity and x_j is the j^{th} coordinate. The gradient consists of 4 components for 2D and 9 for 3D.

VORT Output of the components of vorticity vector, ω, 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, Ω, is obtained. It is defined as:

$$\Omega = -\frac{\partial u_1}{\partial x_2}\frac{\partial u_2}{\partial x_1} \quad \text{for 2D}$$

$$= -\frac{\partial u_1}{\partial x_2}\frac{\partial u_2}{\partial x_1} - \frac{\partial u_2}{\partial x_3}\frac{\partial u_3}{\partial x_2} - \frac{\partial u_3}{\partial x_1}\frac{\partial u_1}{\partial x_3} \quad \text{for 3D}$$

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, τ_{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 μ 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 μ is viscosity and ρ 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.
NARR	See Mode 1 specification.
WIDE	See Mode 1 specification.
V_{frq}	See Mode 1 specification.
TIME	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 **TABL**E) and same combination of **GRAD, VORT, STRUCTURE, STRAIN, STRESS** and **LIGHTHILL** 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

COMMAND **OXIDIZER**

PURPOSE To specify the proportion of inert substance to active substance in the oxidizer for reactive flows.

SYNTAX **OXID {N1}**

N1 The molar ratio of inert to active substance in the oxidizer. The default value is 3.76.

COMMENTS

The default oxidizer is "air" which consists of oxygen and nitrogen in a molar ratio of 3.76 moles of N_2 per mole of O_2. Any other proportion of diluent nitrogen, including zero, may be set by this command.

EXAMPLES

OXIDizer is pure oxygen
OXIDizer with N2:O2 = 2.5 by molar ratio

COMMAND **PAUSE**

PURPOSE To cause a temporary halt in the calculations.

SYNTAX **PAUS**

COMMENTS

Operator intervention is required to restart the calculation process. This command allows for operator action during interactive execution.

EXAMPLES

PAUSe and await operator action

COMMAND **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 **BOUN**DARY 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 **LOCA**TE command with **PAIR** modifier and Sections 7.2.3 and 7.2.4 for further details of the manner in which these surface numbers are assigned.

option Options selected for implementation of the **PERI**ODIC command.

option	INTERPRETATION
LINE	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.
CYCL	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**
Φ	**One or more (≤ 54)** symbols that denote the variables which are periodic (or not periodic if **OFF** modifier is specified) at the θ_1 and θ_2 boundaries. By default all variables are assumed to be periodic.
OFF	The periodic boundary conditions are not applied to the variables specified by the **Φ** modifier. All variable not so identified are assumed to be periodic.
PRES	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

PERIodic boundaries in X
PERIodic boundaries in X, Y and Z
PERIodic boundary in X for U,V,W,T,P only
PERIodic boundary at X option OFF for P and FU variable
PERIodic boundary surface data on file: 'PERIODIC.DAT'
PERIodic boundary in Z direction with LINEAr solver
PERIodic 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] [N1, N2, N3]**

X | Y | Z The boundary conditions for the corresponding x, y or z direction boundaries are determined automatically from the requirement of periodicity. A **BOUN**DARY 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 (θ) direction of a cylindrical coordinate system. At times a problem which is periodic in cylindrical (x,r,θ) 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.

N1 The angle of periodicity, Ψ, in degrees. There is no default value, if the modifier THET is present, then a value must be specified.

N2 A number **(1 ≥ N2 ≥ 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.

N3 A number **(1 ≥ N3 ≥ 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

PERIodic boundaries in Z with THETa = 45 degrees
PERIodic boundaries in Y with THETa = 45 degrees v = 1, w = 2
PERIodic boundary in Z with THETa = 18 degrees for U,V,W,T,P only
PERIodic boundary in Z with THETa = 18 degrees but option OFF for P and FU variable
PERIodic 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] [N1, …, N9]**

X | Y | Z The boundary conditions for the corresponding x, y or z direction boundaries are determined automatically from the requirement of periodicity. A **BOUN**DARY 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:

$$\begin{bmatrix} U2 \\ V2 \\ W2 \end{bmatrix} = \begin{bmatrix} n1 & n2 & n3 \\ n4 & n5 & n6 \\ n7 & n8 & n9 \end{bmatrix} \bullet \begin{bmatrix} U1 \\ V1 \\ W1 \end{bmatrix}$$

STAC The velocity vector at the identified periodic boundaries is transformed according to the transformation specified in the most recent **STAC**K **TRAN**SFORMATION command.

option See Mode 1 specification.

N1,..N9 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

PERIodic boundaries in Z Velocity TRANsformed by:

```
1    0    0
0    0   -1
0    1    0
```

PERIodic boundaries in Z Velocity TRANsformed by:

```
1.0          0.          0.0
        0.0          0.995004     -0.0998334
        0.0          0.0998334     0.995004
```

PERIodic boundaries in Z Velocity by STACk TRANsformation
PERIodic boundary in X by STACk TRANsformation for U,V,W,T,P only
PERIodic boundary TRANsformation as follows with surface data on file: 'PERIODIC.DAT'

```
1.0          0.          0.0
        0.0          0.995004     -0.0998334
        0.0          0.0998334     0.995004
```

PERIodic boundary by STACk TRANsformation surface data on file: 'PERIODIC.DAT'

COMMAND **PHASE**

PURPOSE To specify phase change option for multi-phase option.

SYNTAX **PHAS {CHAN | CLAY | VAPO } [TEMP | T] [N1] [N2, N3, N4]**

CHAN Phase change occurs at a fixed temperature.

CLAY Phase change occurs according to Clayperon's vapor-pressure relation.

$$P_s = P_o \, Exp\left(a_1 - \frac{a_2}{T_s - T_0}\right) ,$$

where, P_S is the vapor pressure, T_S is the phase change temperature, and P_0, T_0, a_1 and a_2 are empirical constants.

VAPO Same as **CLAY**.

TEMP The temperature at which phase change occurs if the **CHAN**ge modifier is specified; otherwise this input is ignored.

T Same as **TEMP** modifier.

N1 The temperature at which the phase change occurs if **CHAN**ge modifier is specified; otherwise it is the base temperature T_0 of the vapor pressure equation given above. The default value is 45.

N2 The pressure P_0 of the vapor pressure equation. The default value is 131.57894.

N3 The constant a_1 of the vapor pressure equation. The default value is 18.3443.

N4 The constant a_2 of the vapor pressure equation. The default value is 3841.1954.

COMMENTS

The phase change option is an advanced option which is currently available only under a consulting arrangement with ACRi.

EXAMPLES

PHASe CHANge at 100 Degrees Centigrade
PHASe CHANge according to CLAYperon's equation: T0=45, p0=131.57894, a1=18.3443, a2=3841.1954

COMMAND **PRANDTL**

PURPOSE To specify Prandtl number for the fluid.

SYNTAX **PRAN [EFFE] { σ }**

EFFE If the modifier **EFFE**CTIVE is present on the command line then the input is assumed to be for the effective Prandtl number, σ_e, otherwise it is assumed to be the molecular Prandtl number, σ, for the fluid (Equations 3.3.3 and 3.3.4).

σ The effective or molecular Prandtl number **(>0)** for the fluid. The default value is 0.7.

EXAMPLES

PRANdtl number for fluid = 1.
PRANdtl number EFFEctive value = 0.5

COMMAND **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 6.8.1. 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 Sections 7.2.3 and 7.2.4.

EXAMPLES

PRINt FLOW immediately
PRIN STATistics for U, V, T now
PRIN FLOW and STATistics for U, V, T now

MODE 2: **Integrated Averages for Selected Variables at Uniformly Spaced Locations**

SYNTAX **PRIN {Φ} {AVER | PROF} {N1} {coordinate} [subrgn] [dir] [TOLE=V_{tol}] [NORM] [BASE=V_{basel}] [MASS | AREA | VOLU] [OUTP | fname]] [V_{frq}] [TIME]**

Φ One or more of the symbols that represent the variables for which output is desired. The valid symbols are listed in Tables 6.8.1 and 6.8.2. 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 the averages are to be computed only at the boundary. See Section 7.2.5 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: $S_i - V_{tol} \le S < S_i + V_{tol}$ 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 Φ_{base} and Φ_{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 The base value (Φ_{base}) for non-dimensional output. This modifier will automatically select the normalized (**NORM**) form of the output. If this modifier exists, then **V_{base}** must be specified and, in this case, only one variable (**Φ**) must be specified on the command.

V_{base} The user specified value for Φ_{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 Φ_{norm} is taken to be arithmetic mean of the values for the sub-domain. If this modifier is present, then Φ_{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 Φ_{norm} is the arithmetic mean of the values for the sub-domain. If this modifier is present, then Φ_{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. The total number of open files in any simulation can not exceed 64.

V_{frq} The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 for further details. **If V_{frq} 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_{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.

COMMENTS

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 Φ should be the symbol T (for Temperature) and the base value, **V_{base}**, should be specified as the average inlet temperature (commonly called T3). The Φ_{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 {Φ} {AVER | PROF} {N1} {N2,…,Nn} {coordinate} [subrgn] [dir] [TOLE=V_{tol}] [NORM] [BASE=V_{basel}] [MASS | AREA | VOLU] [OUTP | fname]] [V_{frq}] [TIME]**

Φ — One or more of the symbols that represent the variables for which output is desired. The valid symbols are listed in Tables 6.8.1 and 6.8.2. 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.

N2,..Nn — The coordinates of the locations at which averages are computed. There must be a total of **N1** values and these must precede any other numerical input except **N1**. **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 **coordinate** for the **subrgn**. There is no default value.

coordinate — Same as Mode 2 of the command.

subrgn — Same as Mode 2 of the command.

dir — Same as Mode 2 of the command.

TOLE — The coordinate tolerance, **V_{tol}**., 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 0.01.

V_{tol} — Same as Mode 2 of the command.

STAT — Same as Mode 2 of the command.

NORM — Same as Mode 2 of the command.

BASE — Same as Mode 2 of the command.

V_{base} — Same as Mode 2 of the command.

MASS — Same as Mode 2 of the command.

AREA — Same as Mode 2 of the command.

VOLU — Same as the **AREA** modifier.

fname — Same as Mode 2 of the command.

V_{frq} — Same as Mode 2 of the command.

TIME — 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+ every 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 [Φ] {FORC | MOME } [subrgn] [dir] [X_i] [Φ_o] [V_{freq}] [TIME] [NOW] [OFF] [fname]**

Φ The symbol for the variable for which the "force" or "moment" is to be computed. The valid symbols are listed in Tables 6.8.1. 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:

$$F_i = \int (\Phi + \Phi_o)\, n_i \, dA$$

$$M_i = F_i \otimes (X_i - Y_i)$$

F_i is the force vector for the **Φ** variable,
Φ_o is a reference datum for the **Φ** 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 **Φ** 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 **FORC**E 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 7.2.5 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.

Φ_o Datum for the selected variable **Φ**. If specified this must be the 3rd numerical value for 2D and 4th numerical value for 3D geometry. The default value is zero.

V_{frq} The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 for further details. If specified this must be the 4th numerical value for 2D and 5th numerical value for 3D geometry. By default the output is obtained only at the end of simulations.

TIME By default, **V_{frq}** 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 **Φ** and **subrgn** is deactivated.

fname The file name for output. The default file name is 'acr_FORCE.TMP'. 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 the default file. The total number of open files can not exceed 64. 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_{frq}] [TIME] [NOW] [OFF]**

FLUX The convective and diffusive fluxes for a 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 6.7.1. If no symbol is specified then output is obtained for each active variables for which a transport equations 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
INLE	Boundaries specified by the **INLE**T command are selected.
OUTL	Boundaries specified by the **OUTL**ET command are selected.
OPEN	Boundaries specified by the **OPEN** command are selected.
IO	All boundaries specified by **INLE**T, **OUTL**ET or **OPEN** command are selected.
WALL	Walls specified by **WALL** or **BLOC**K command are selected.
EXTE	All external (or outer) boundaries of the computational domain are selected.
ALL	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 7.2.5 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 can not exceed 64.

V_{frq} The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 for further details. By default the output is obtained only at the end of simulations.

TIME By default, **V_{frq}** 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_{frq}** 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_{frq}] [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
INLE	Boundaries specified by the INLET command are selected.
OUTL	Boundaries specified by the OUTLET command are selected.
OPEN	Boundaries specified by the OPEN command are selected.
IO	All boundaries specified by INLET, OUTLET or OPEN command are selected.
WALL	Walls specified by WALL or BLOCK command are selected.
EXTE	All external (or outer) boundaries of the computational domain are selected.
ALL	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 7.2.5 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 can not exceed 64.

V_{frq} The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 for further details. By default the output is obtained only at the end of simulations.

TIME By default, **V_{frq}** 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_{frq}** 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

COMMAND **PROBLEM**

PURPOSE To specify the general nature and type of problem to be solved.

MODE 1: **Atmospheric Pressure Computation Mode**

SYNTAX **PROB { ATMO }**

ATMO The "atmospheric" mode of **ANSWER™** is invoked. By default the body force in the momentum equations is computed as (ρ g_j) where 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 (ρ - ρ^*) g_j. This effectively adds hydrostatic pressure (ρ^* g_j x_j) to the computed pressure. In this case, the thermodynamic pressure, p, is related to the computed pressure by:

$$p = p_{computed} + \rho^* \sum_j (x_j^* - x_j)\, g_j$$

where x^*_j is the coordinate datum in the j^{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, $p_{computed}$; that is the pressure boundary values must be in terms of **the thermodynamic pressure minus the hydrostatic pressure.**

COMMENTS

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_o }**

BOUS The "Boussinesq" approximation mode of **ANSWER™** is invoked. In this mode, a source terms for body force is added to momentum equations. The source term, S_i, for the i^{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^{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, ξ is the symbol T (for temperature). The valid symbols are listed in Table 6.8.1.

β The numerical value for the coefficient for the body force defined as:

$$\beta = \frac{\partial \rho}{\partial \xi}$$

where ρ is the density of the fluid. If the working fluid is a gas and the perfect gas law applies, then β can be estimated from:

$$\beta = -\frac{\rho_o}{T_o}$$

where T_o is a reference temperature and ρ_o is a reference density. For air at atmospheric pressure and 300K, β ≈ -0.004 [Kg/(m^{3}K)]. For water at 300K, β ≈ -0.2 [Kg/(m^{3}K)].

ξ_o 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 ξ in the computational domain.

COMMENTS

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 **ALLO**CATE 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 **ALLO**CATE command can be used to specify a composite variable ξ 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, ξ, in turn can be defined by an appropriate SET command with the **ALWA**YS modifier so that it is continuously updated as T and C change.

EXAMPLES

PROBlem BOUSSINESQ approximation: function of T with beta=0.004, T0=300

MODE 3: Other Geometry or Client Specific Options

SYNTAX **PROB {RADI | option } [attributes] [N1, …, N4]**

RADI 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.

N1,.., N4 Numerical input relevant to the implemented option.

COMMENTS

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

COMMAND **PROPERTY**

PURPOSE To specify the mode of interpretation and calculation of fluid properties.

SYNTAX **PROP [Φ] [option]**

Φ One or more symbols that represent corresponding variable(s) for which the input is effective. The valid symbols are listed in Table 6.7.1. If no symbol is specified, the input is assumed to be effective for all variables.

option The diffusion coefficient is computed at a element interface as described in the table below.

option	INTERPRETATION
ARIT	Arithmetic mean of the two nearest grid node values. This is the default option for the ANSWER™ Software Tool
GEOM	Geometric mean of the two nearest grid node values.
HARM	Harmonic mean of the two nearest grid node values. This is the default option for the PORFLOW™ and TIDAL™ Software Tools.
UPWI	Value at the node located upstream of the interface.
MINI	The smaller of the two nearest grid node values.
MAXI	The larger of the two nearest grid node values.

EXAMPLES

PROPerty at interface for all variables by arithmetic average
PROPerty for U use GEOMetric mean
PROPerty averages for T and C by UPWInd mean
PROPerty for C by MAXImum method

COMMAND QUIT

PURPOSE To signify the termination of a sequence of problems.

SYNTAX QUIT

COMMENTS

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

COMMAND RADIATION

PURPOSE To activate thermal radiation calculations and specify radiation constants.

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_2 and H_2O) 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 practically purposes this is equivalent to a non-participating fluid medium.

COMMENTS

The modifiers of this mode can be combined with any of the other modes.

EXAMPLES

RADIation by VIEW factor approach.
RADIation computations in PASSive mode
RADIation computations by VIEW factor method in PASSive mode

MODE 2: Radiation Computation Frequency

SYNTAX RADI {FREQ} [N1]

FREQ The frequency of computation of radiation fluxes.

N1 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

RADIation FREQency of computations is 20 steps.

MODE 3: Specification of Gas Absorptivity

SYNTAX RADI {ABSO} {func [ξ]} [subrgn] {fname | N1 ..., Nn}

ABSO The absorptivity for radiation is specified. By default gas absorptivity is computed from the built-in formulae, which take account of gas and soot constituents.

func One of the modifiers listed in Table 7.2.3 that denotes the functional form of gas absorptivity. If no function is specified then the value is assumed to be constant.

ξ One of the independent variables listed in Table 7.2.4. If no variable is specified, then the independent variable is assumed to be time.

subrgn The sub region for which the input is specified. See Sections 7.2.3 and 7.2.4. If no sub region is specified, the entire computational domain is selected.

fname The name of the file containing the numerical values N1 through Nn. This option can be used only if the selected function is a table or one of the series functions. See Section 7.2.2 for additional information.

N1, .., Nn The numerical constants and coefficients for the selected function. See Section 7.2.7 for more details. There are no default values for this input.

COMMENTS

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 7.2.8. The command keyword (RADI) with modifier ABSO must replace the keyword used in these examples.

MODE 4: Specification of Surface Emissivity

SYNTAX RADI {EMIS} {func [ξ]} [subrgn] [dir] [WALL] {fname | N1 ..., Nn}

EMIS 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.

func One of the modifiers listed in Table 7.2.3 that denotes the functional form of surface emissivity. If no function is specified then the value is assumed to be constant.

ξ One of the independent variables listed in Table 7.2.4. If no variable is specified, then the independent variable is assumed to be time.

subrgn The sub region for which the input is specified. See Sections 7.2.3 and 7.2.4. If no sub region is specified, the entire computational domain is selected.

dir The orientation index for the boundary associated with the emissivity. See Section 7.2.5 for available choices. If no **dir** is specified, then all surfaces of the selected **subrgn** are selected.

WALL If this modifier is present then any specification of **subrgn** and **dir** is ignored and the input is applied to all walls of the computational domain.

fname The name of the file containing the numerical values **N1** through **Nn**. This option can be used only if the selected function is a table or one of the series functions. See Section 7.2.2 for additional information.

N1, .., Nn The numerical constants and coefficients for the selected function. See Section 7.2.7 for more details. There are no default values for this input.

EXAMPLES

Generic examples for this command are given in Section 7.2.8. 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 =σ]

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 ≥ 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

σ The numerical value for the Stefan-Boltzmann constant. It must be > 0.

EXAMPLES

RADIation SCATtering coefficient is 0.02
RADIation STEFan-Boltzmann constant is 4.76E-13 in fps units.
RADIation SCATtering coefficient= 0.02, STEFan= 4.76E-13 in fps units.

COMMAND **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:

$$C_nH_m \rightarrow C_nH_{m-2} + H_2$$
$$C_nH_{m-2} + \tfrac{1}{2}\, n\, O_2 \rightarrow n\, CO + \tfrac{1}{2}\,(m-2)\, H_2$$
$$CO + \tfrac{1}{2}\, O_2 \rightarrow CO_2$$
$$H_2 + \tfrac{1}{2}\, O_2 \rightarrow H_2O$$

If this modifier is present, then the reaction mechanism is assumed to be:

$$C_nH_m + \tfrac{1}{2}\, n\, O_2 \rightarrow n\, CO + \tfrac{1}{2}\, m\, H_2$$
$$CO + \tfrac{1}{2}\, O_2 \rightarrow CO_2$$

See Section 3.6 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. **See Section 3.6 and comments below for further details.**

PROD This modifier is used if the user selects the explicit reaction mode (Mode 2) but still wants to employ the algebraic relations given in Section 3.6.2 to determine the mass concentrations of O_2, H_2O, CO_2, and N_2.from the solved species and conserved variables.

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.

COMMENTS

The default reaction system consists of the 4 steps and 8 species described in Section 3.6. The total gas phase consists of 8 chemical species: C_nH_m, C_nH_{m-2}, CO, H_2, O_2, H_2O, CO_2, and N_2. The default fuel is $C_{10}H_{19}$. This can be changed with the **FUEL** commands. The default Arrhenius reaction constants are given in Table 3.6.1; these can be changed by the **ARRH**ENIUS 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] {N1, .., N5] [Nk]

idreac A unique identifier for the reaction rate. The reaction rate is computed as:

$$\text{Rate} = Z_0 \exp\left(-\frac{E}{R_u T}\right) \Phi_1^{\alpha} \Phi_2^{\beta} \Phi_3^{\gamma} T^{\vartheta}$$

where Z_0 is the pre-exponential constant, E is the activation energy, R_u is the universal gas constant, (E/R_u 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_0 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_o 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_u) 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 Sections 7.2.3 and 7.2.4.

option The modifier for interpretation of numerical input

option	INTERPRETATION
MASS	By default it is assumed that the computed Φ's are in mass units and that **N1** is given in molar units. Therefore **N1** is internally converted to appropriate mass units. If this modifier is present, then no unit conversion is performed. It is assumed that **N1** is specified in appropriate units for the computed Φ's.
CGS	By default, **N1and N2** are assumed to be in specified SI (kg-mole - m^3 - J - K) units. If this modifier is present, then **N1** and **N2** are assumed to be in the cgs (gm-mole – cm^3 - 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.
LIFE	**N1** is in units of time [t] representing the half -life of the reaction. In this case **N2** must be the activation temperature [T].
DECA	**N1** is in units of inverse of time [t^{-1}] representing the decay rate of the reaction. In this case **N2** must be the activation temperature [T].

N1 The pre-exponential constant, Z_0. The specified value may be internally converted to appropriate units based on the **option** modifier. For example, in the CGS system, units are $cm^{3n-3}/moles^{n-1}$-s, where n = **N3 + N4 + N5**.

N2 Activation energy, E, (J/kg-mole or cal/gm-mole) or activation temperature, E/R_u, (Kelvin).

N3,N4,N5 Stoichiometric coefficients or exponents, α, β and γ, for the species specified by Φ's. There is no default value; one value must be specified for each identified reactant.

Nk Exponent of the temperature, ϑ, in reaction rate equation; it is the 4^{th}, 5^{th} or 6^{th} numerical value depending on the order of the reaction (1, 2 ,or 3 reactants). The default value is 0.

EXAMPLES

REACtion R1 preexp=1.960E+18, activ_e=2.48E+04 FU ^ 0. and O2 ^ 0.5
REACtion R1 FU, O2 1.960E21 2.062E8 0.0 0.5 1.07 CGS EBU TEMP SELEcted subregion
REACtion R2 CH, O2, FU: 5.395E16, 2.50E04 0.0 0.9 1.18 -0.57 TEMP for ID=RGN1 active
REACtion R1 MASS based preexp=1.50, activ_T= 600 FU ^ 1. and O2 ^ 1
REACtion R1 MASS FU, O2 1.50 0. 1 0.5 1.07 EBU ! No effect of temperature
REACtion R2 DECAy frequency FU: 2.00, 1000 1.0 for SELEcted region
REACtion 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} {N1, N2} {fname | N3, …., Nn}**

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 **N1** times **N2**.

N1 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.

N2 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

N3,…,N5 These represent the numerical values in the table. If the file name is specified, then these values are read from the file; otherwise N1 times N2 values must be specified with the command. There is no default value.

EXAMPLES

REACtion FAST chemistry; lookup table has 1000 rows and 5 columns on file ‘FAST_CHEM’
REACtion FAST chemistry 1000 rows & 5 columns values !!!! 5000 values must be specified)

MODE 4: **Empirical Constants for Fast-Chemistry Reaction**

SYNTAX **REAC {FAST} {CONS} {N1, N2}**

FAST The two-parameter PDF fast-chemistry reaction model is used.

CONS The empirical constants required for the fast-chemistry reaction are specified.

N1 The constant C_{g1} **(>0)** for fast-chemistry reaction (see Jones and Whitelaw, 1982). The default value is 2.

N2 The constant C_{g2} **(>0)** for fast-chemistry reaction (see Jones and Whitelaw, 1982). The default value is 2.

EXAMPLES

REACtion FAST chemistry CONStant 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

REACtion R1 for subdomain ID=R1DMn switched OFF
REACtion R2 for SELEcted subdomain OFF

COMMAND 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 = Φ] [STAR] [N1]

fname The name of the file from which the input is obtained. See Sections 6.4 and 7.2.2 for more information. There is no default file name; a file name must be specified.

fmt The modifier "**FORM**ATTED" or "**UNFO**RMATTED" 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 6.10).

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.

Φ **One or more** of the character strings listed in Table 6.8.1 that represent the corresponding variables on the file that are 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.

N1 The data-set number **(≥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 6.9, a data set in this context consists of several records. The default value is 1.

COMMENTS

If no simulation time has been explicitly set (such as by the **TIME** command) and a file is read at the start of the computations, then the starting simulation time is set to be the value read from the file. This ensures a continuity of time from the previous record read from the file. In this case, for transient simulations, the time specified on **SOLV**E command must account for the that the starting time is that read from the file.

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 **REST**ART 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, OUTP**UT and **SOLV**E 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 6.4 and 7.2.2 for more information. The default file name is the output file name (see **OUTP**UT 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".

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

COMMAND **REFERENCE**

PURPOSE To specify the reference values of selected constants and variables.

MODE 1: **Specify Reference Value for a Variable**

SYNTAX **REFE {Φ |TEMP| DENS } [ABSO | FREEZ | BOIL] [N1]**

Φ The symbol for the variable to which the specified input applies. The valid symbols are listed in Table 6.8.1.

TEMP The reference value for temperature, T*, is specified. The default value is 0.

DENS The reference value for density, ρ^*, is specified. The default value is 1.

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.

BOIL The boiling 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 100.

N1 The reference value of the corresponding variable specified by Φ. or another modifier.

EXAMPLES

REFErence P is 2.5E5 !!! N/m^2
REFErence T = 30. deg C
REFErence T = 60. F
REFErence T ABSOlute base = 459.67 R
REFErence ABSOlute TEMPerature base = 459.67 R
REFErence FREEZing TEMPerature is = 32 R
REFErence BOILing TEMPerature is = 212 R

MODE 2: **Specify Absolute Pressure mode for PORFLOW™**

SYNTAX **REFE { P } { ABSO }**

ABSO In the PORFLOW™ Software Tool, by default the governing equation for P^n, pressure for the n^{th} fluid phase, is solved for the normalized form of pressure;

$$P^n = \frac{p - p^*}{\rho^* g}.$$

The pressure is therefore in units of length (also called Head). If this modifier is present then the governing equation for pressure is solved with the normalizing denominator set to unity. Thus pressure is now in absolute thermodynamic units (such as N/m^2) with reference to the datum pressure, p*. In this form of the equations, all initial and boundary conditions must be specified with reference to the datum pressure in absolute thermodynamic units. The sources and fluxes are then in mass units and input for hydraulic conductivity (**HYDR**AULIC command) must be replaced by that for hydraulic permeability divided by viscosity.

EXAMPLES

REFErence P in ABSOlute mode

MODE 3: **Specify Constants and Options for Pressure Related Choices**

SYNTAX **REFE { P } [VALU | NORM] { N1 }**

P The specified input pertains to the pressure equation.

VALU 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 is the default option. This input can also be provided by the **GAS** P or **GAS PRES** command.

NORM The normalizing pressure is specified (see **REFE P ABSO** command). By default the normalizing pressure is unity except for PORFLOW™ it is the pressure head ($\rho^* g$) where g is the gravitational constant.

N1, Nn The datum or normalizing pressure value.

EXAMPLES

REFErence P = 10000 for this case.
REFErence NORMalizing P = 1
REFErence NORMalizing P = 10

MODE 3: **Specify Absolute Pressure mode for PORFLOW™**

SYNTAX **REFE { P } { ABSO }**

ABSO In the PORFLOW™ Software Tool, by default the governing equation for P^n, pressure for the nth fluid phase, is solved for the normalized form of pressure;

$$P^n = \frac{p - p^*}{\rho^* g}.$$

The pressure is therefore in units of length (also called Head). If this modifier is present then the governing equation for pressure is solved with the normalizing denominator set to unity. Thus pressure is now in absolute thermodynamic units (such as N/m^2) with reference to the datum pressure, p*. In this form of the equations, all initial and boundary conditions must be specified with reference to the datum pressure in absolute thermodynamic units. The sources and fluxes are then in mass units and input for hydraulic conductivity (**HYDR**AULIC command) must be replaced by that for hydraulic permeability divided by viscosity.

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, ρ 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**).

NORM The normalizing pressure is specified. By default the normalizing pressure is unity except for PORFLOW™ it is the pressure head (ρ^* g) where g is the gravitational constant.

N1, Nn The location of the reference pressure if the **NODE** or **COEF** modifiers are present or the normalizing pressure value if the **NORM** modifier is present. In the latter case, only **N1** is processed.

EXAMPLES

REFErence NODE for P at (I=5, J=10, K=3)
REFErence location for P COEFficient is (I=1, J=5, K=2)
REFErence NORMalizing P = 1

MODE 4: **Location of Reference Pressure**

SYNTAX **REFE { 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, ρ 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)

COMMAND RELAX

PURPOSE To specify relaxation factors for iterative solution of the matrix of equations.

SYNTAX RELA { [Φ=N1, Φ=N2, ..., Φ=Nn] | [OFF] }

Φ One or more of the symbols for which the relaxation parameters are specified. The valid symbols are those listed in Table 6.7.1 plus **RHO**, **VIS** and **PP**.

OFF All relaxation parameters are set to unity. Any other input in the command is ignored. In essence, the relaxation feature is disabled.

N1, .., Nn Relaxation factor (Θ of Equation 4.2.2) for the variable denoted by the symbol immediately preceding the value. The numerical value must be larger than 0 and less than 2.

COMMENTS

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 **CONV**ERGENCE 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 OFF

COMMAND RENAME

PURPOSE To rename the output variables listed in Table 6.8.1.

SYNTAX RENA {Φ = name}

Φ 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, Φ. 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"
```

COMMAND 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 [Φ] [BLOC|TABL] [ADD] [fname] [fmt] [GEOM|COMP] [DATA] [REPL|SEQU] [subrgn] [V_{frq}] [TIME] [IMME|NOW|ONLY|OFF]

Φ **One or more** of the symbols that represent the variables for which output is desired. The valid symbols are listed in Table 6.8.1. 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 1st **SOLV**E 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 1st element. This is the default mode.

TABL The file is written as a columnar table of values with one record per element. Each record contains the element number, the element material type, and its node coordinates (x and y for 2D and x, y and z for 3D) followed by the specified variables. If there are more than 7 total values (including node coordinates but excluding element number and material type) then the record wraps to the next line of output with 7 variable values per line.

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 7.2.2 for additional information. The default file name is the Standard Output Unit name (see **OUTP**UT Command) but with the extension changed to "SAV"' for the **BLOC**K and "_TABLE.SAV" for the **TABL**E 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 7.2.2 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 "**FORM**ATTED" or "**UNFO**RMATTED", 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 Φ above). If **GEOM**ETRY modifier is present, then **only** the problem geometry and grid connectivity information is written to the file; the Φ 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 Φ variables are written. If both modifiers are specified, then **COMP**ACT modifier takes precedence over the **GEOM**ETRY 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_{frq} 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_{frq} 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_{frq} 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 9999th file.

subrgn This modifier is effective only for the **TABL**E 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 Sections 7.2.3 and 7.2.4.

V_{frq} The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 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.

TIME 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.

IMME The output is produced immediately. This modifier is necessary only before the 1st **SOLV**E command is given. It then enables the output of the current value of specified variable(s) immediately. After the 1st **SOLV**E 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 1st **SOLV**E command.

NOW If this modifier is specified before the 1st **SOLV**E command, then the output is produced just before the 1st 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 1st **SOLV**E 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 in response to a **NOW** or **IMME**DIATE modifier or at the frequency determined by **Vfrq**.

OFF Output for any previous command(s) for the same **subrgn** and type (**BLOC**K or **TABL**E) is subsequently suppressed.

COMMENTS

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 LOCAte 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'

MODE 2: **Output of Properties and Coefficients for the Differential Equations**

SYNTAX **SAVE {Φ} { [COND| DIFF] [STOR] [SOUR] [MATR] [RESI] } [BLOC|TABL] [ADD] [fname] [fmt] [GEOM|COMP] [DATA] [REPL|SEQU] [subrgn] [V_{frq}] [TIME] [IMME|NOW|ONLY|OFF]**

Φ **One, and only one,** of the symbols that denotes the dependent variable for which a differential equation is solved. The valid symbols are listed in Table 6.7.1. There is no default value; a valid symbol must be specified.

COND 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.

DIFF Same as **COND**.

STOR The storage coefficient for the variable is printed to the output file.

SOUR The net source (algebraic sum of source and sink) for the variable is printed to the output file.

MATR 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.

RESI 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.

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 7.2.2 for additional information.

fmt See Mode 1 specification.

GEOM See Mode 1 specification.

COMP See Mode 1 specification.

DATA See Mode 1 specification.

REPL See Mode 1 specification.

SEQU See Mode 1 specification.

subrgn See Mode 1 specification.

V_{frq} See Mode 1 specification.

TIME 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 (**BLOCK** or **TABLE**) and same combination of **COND, DIFF, STOR, SOUR, MATR,** and **RESI** modifiers is subsequently suppressed.

COMMENTS

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.

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

MODE 3: **Output of Convective, Diffusive and Total Flux for each Element**

SYNTAX **SAVE {Φ} {FLUX} [CONV] [DIFF] [TOTA] [AREA] [BLOC|TABL] [ADD] [fname] [fmt] [GEOM|COMP] [DATA] [REPL|SEQU] [subrgn] [V_{frq}] [TIME] [IMME|NOW|ONLY|OFF]**

Φ **One, and only one,** of the symbols that denotes the dependent variable for which a differential equation is solved. The valid symbols are listed in Table 6.7.1. There is no default value; a valid symbol must be specified.

FLUX 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.

CONV The convective flux for each face of the elements of the computational domain is printed.

DIFF The diffusive flux for each face of the elements of the computational domain is printed.

TOTA The sum total of the convective and diffusive fluxes for each face of the elements of the computational domain is printed. This modifier is assumed by default if the **CONV** or **DIFF** modifiers are not present.

AREA The computed flux is divided by the projected area of the face for output.

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 7.2.2 for additional information.

fmt See Mode 1 specification.

GEOM See Mode 1 specification.

COMP See Mode 1 specification.

DATA See Mode 1 specification.

REPL See Mode 1 specification.

SEQU See Mode 1 specification.

subrgn See Mode 1 specification.

V_{frq} See Mode 1 specification.

TIME 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 (**BLOC**K or **TABL**E) and same combination of **FLUX, CONV, DIFF** and **TOTA**L modifiers is subsequently suppressed.

COMMENTS

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.

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] [V_{frq}] [TIME] [IMME|NOW|ONLY|OFF]**

GRAD The tensor components of the gradients of velocity, φ_{ij} are output, where:

$$\varphi_{ij} = \frac{\partial u_i}{\partial x_j}$$

Here u_i is the i^{th} component of velocity and x_j is the j^{th} coordinate. The gradient consists of 4 components for 2D and 9 for 3D.

VORT Output of the components of vorticity vector, ω, 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, Ω, is obtained. It is defined as:

$$\Omega = -\frac{\partial u_1}{\partial x_2}\frac{\partial u_2}{\partial x_1} \quad \text{for 2D}$$

$$= -\frac{\partial u_1}{\partial x_2}\frac{\partial u_2}{\partial x_1} - \frac{\partial u_2}{\partial x_3}\frac{\partial u_3}{\partial x_2} - \frac{\partial u_3}{\partial x_1}\frac{\partial u_1}{\partial x_3} \quad \text{for 3D}$$

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, τ_{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 μ 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 μ is viscosity and ρ 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 7.2.2 for additional information.
fmt	See Mode 1 specification.
GEOM	See Mode 1 specification.
COMP	See Mode 1 specification.
DATA	See Mode 1 specification.
REPL	See Mode 1 specification.
SEQU	See Mode 1 specification.
subrgn	See Mode 1 specification.
V_{frq}	See Mode 1 specification.
TIME	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 **TABL**E) and same combination of **GRAD, VORT, STRUCTURE, STRAIN, STRESS** and **LIGHTHILL** 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: **Restart File for Later Continuation of Simulations**

SYNTAX **SAVE { REST } [fname] [V_{frq}] [TIME] [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 **REST**ART modifier to restart the computations for the same identical problem. Only the **SOLV**E command may be modified for the new simulations; all other input data must be identical to the run that generated the file. **REST**ART 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 7.2.2 for additional information. The default file name is the output file name (see **OUTP**UT 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".

V_{frq} The frequency (step or time interval) at which the restart file specified by **'fname'** is written. See Section 7.2.11 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.

TIME By default, the **V_{frq}** is interpreted to be the frequency in terms of number of steps. If this modifier is present, then **V_{frq}** is interpreted to be time interval between successive outputs.

NOW A restart file is immediately written. If **V_{frq}** modifier is not present, then the file is written only once, otherwise the file is replaced at the specified frequency.

EXAMPLES

SAVE REST
SAVE RESTart file ! acr_RESTART.TMP file will be generated at end of simulations
SAVE RESTart file as 'STAGE_1.FILE' ! File STAGE_1.FILE will be generated at end of simulations
SAVE RESTart NOW ! acr_RESTART.TMP file will be immediately generated.
SAVE RESTart 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 RESTart 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.

COMMAND 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.

COMMENTS

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 the **COOR**DINATE, **SET** and **SOUR**CE commands.

EXAMPLES

SCALe multiply by 3.3 and add 10.
SCALe multiply by 0.3048

COMMAND **SCHMIDT NUMBER**

PURPOSE To specify the Schmidt number for the fluid.

SYNTAX **SCHM [EFFE] [Φ=N1, Φ=N2, ..., Φ=Nn]**

EFFE If the modifier **EFFE**CTIVE 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 (Equations 3.3.3 and 3.3.4)

Φ One or more symbols to denote the variables for which the Schmidt numbers are specified. The valid symbols are those listed in Table 6.7.1 with the exception of the U, V, W and P. If no symbol is specified, then the input is assumed to be for the first mass transfer variable in the system of equations.

N1, .., Nn Schmidt numbers for the dependent variables denoted by the symbol immediately preceding the value. The numerical value must be greater than 0. The default value is 0.7.

EXAMPLES

SCHMidt number for fluid = 1.
SCHMidt number for FF = 0.5, FU = 0.7, CH = 0.7
SCHMidt number EFFEctive value for FF = 0.5

COMMAND **SCREEN**

PURPOSE To control the echo of diagnostic output obtained from the **DIAG**NOSTIC 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 (**DIAG**NOSTIC command) are echoed to the CRT device. If this modifier is present, then the maximum is set to 9.

COMMENTS

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

SCREen echo for diagnostic output to be turned OFF
SCREen ON ! This is the default option
SCREen in WIDE format
SCREen ON in WIDE format ! Turn on previously off screen output

COMMAND 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.

COMMENTS

This command is identical in all respects to the LOCATE command.

EXAMPLES

See LOCATE command

COMMAND 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 {Φ} [func [ξ]] [option] [mod] [ALWA] [subrgn] [FIEL] [dir] {N1} [fname | N2 .., Nn] [STAC]

Φ **One, and only one**, symbol that denotes the variable for which the values are specified. The valid symbols include those listed in Table 6.8.1 plus the user defined variables (**ALLO**CATE 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 7.2.3, which denotes the functional form of the dependent variable. If no function is specified, the value is assumed to be constant.

ξ One of the independent variables listed in Table 7.2.4. If no variable is specified, the independent variable is assumed to be time.

option Option selected for the operation to be performed.

option	INTERPRETATION
REPL	The computed value replaces the existing value. This is the default option.
ADD	The computed value is added to the existing value.
SUBT	The computed value is subtracted from the existing value.
MULT	The computed value multiplies the existing value.
DIVI	The computed value divides the existing value. A value of 1.E-30 is added to the divisor if it is zero.

mod The modifier for function evaluation

mod	INTERPRETATION
ABS or **ABSO**	The absolute value for the computed function is taken.
POSI	The negative values for the computed function are set of zero
NEGA	The positive values for the computed function are set of zero

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.

FIEL Unless the **subrgn** was specified by a **LOCA**TE command with the **FIEL**D modifier, the values are set at all nodes in the subregion **and** any nodes at the **exterior** boundary that are right next to the **subrgn** (see the **LOCA**TE command). If this modifier is present, then only the interior field nodes are set by the command. This modifier is ignored if the **LOCA**TE command was specified for a structured grid with the grid index coordinates.

dir By default, the input is applied to all the elements or nodes in the subregion defined by the **subrgn** 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 Section 7.2.5 for available choices. The **SET** command may thus be effectively used as an alternative to the **BOUN**DARY command.

fname The name of the file (see Section 7.2.2) from which **N2** through **Nn** are read. This option can be used only if the selected function is a table or one of the series functions.

N1, .., Nn The numerical constants and coefficients for the selected function. See Section 7.2.7 for more details. There are no default values for this input.

STAC By default Φ is computed from the value of ξ at the same location. If this modifier is present, then Φ is computed from ξ at a location previously specified by the **STAC**K **LOCATE** command.

EXAMPLES

Generic examples for this command are given in Section 7.2.8. 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 {Φ} { NODE | STRU | UNST | fname } [V_1,..V_m] [subrgn] [FIEL] [dir]**

Φ A symbol to denote the variable for which the values are specified. The valid symbols include those listed in Table 6.8.1 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 if is assumed that the input values are given in the **STRU**CTURED mode if the **subrgn** was defined with a **LOCA**TE command in terms of grid indices or in the **UNST**UCTURED 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.

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 Φ, and **MLO and MIHI** 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 **LOCA**TE command which can be examined by including a file name on the **LOCA**TE command.

fname The name of the file for numerical values unless the values are directly specified by . **V_1….V_m** below. See Section 7.2.2 for additional information.

V_1….V_m 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.

FIEL See Mode 1 specification.

dir See Mode 1 specification.

COMMENTS

It is highly recommended that the user examine the input data (for example by running the data set with **SOLV**E **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 Multiple Variables on a Node-by-Node Basis from a Table**

SYNTAX **SET {Φ_1, …, Φ_n} { NODE | STRU | UNST | fname } {V_1,..V_{nm} } [subrgn] [FIEL] [dir]**

Φ_1, …, Φ_n Symbols to denote the variables that are to be set by the command. The symbols include all those listed in Table 6.8.1 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 Φ_n 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 if is assumed that the input values are given in the **STRU**CTURED mode if the **subrgn** was defined with a **LOCA**TE command in terms of grid indices or in the **UNST**UCTURED 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 **Φ'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 **Φ's**, **n** denotes the number of variables, and **MLO and MIHI** 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 **LOCA**TE command which can be examined by including a file name on the **LOCA**TE command.

fname The name of the file (see Section 7.2.2) from which **V_1** through **V_{nm}** are read.

V_1….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 4: **Specification of a Variable as a Linear Sum of other Variables**

SYNTAX **SET** **{Φ} {SUM | LINE } [MASS] {ξ_1, …, ξ_n } [a_1, .., a_n] [a_0] [option] [mod] [ALWA] [subrgn] [FIEL]**

Φ See Mode 1 specification.

SUM The variable Φ is computed from:

$$\Phi = \sum_n \xi_n$$

LINE The variable Φ is computed from:

$$\Phi = \sum_n a_n \xi_n + a_0$$

MASS By default, the ξ'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 ξ's are replaced by the total mass of the corresponding ξ variable. The total mass is equal to the quantity that appears in the accumulation term of the governing differential equation for ξ (see Chapter 2) multiplied by the volume of the element. In the presence of this modifier only those ξ's can appear on the right hand side for which differential equations are solved.

ξ_1, …, ξ_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 6.8.1 plus the variables defined by the user (**ALLOC**ATE 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, …, a_n The numerical constants and coefficients for the linear function. These values must be specified if the **LINE**AR modifier is present. With the **SUM** modifier these values must not be specified. There is not default value; appropriate number of values must be specified.

a_0 The datum for the linear function if **LINE**AR 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 5: Specification of a Variable as a Square or Square Root Sum of other Variables

SYNTAX **SET** **{Φ} {SQUA | ROOT} { ξ_1, …, ξ_n }[a_1, .., a_n] [a_0] [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_1, …, ξ_n Symbols to denote the variables that contribute to the linear sum as in the above equation. The valid symbols include those listed in Table 6.8.1 plus the variables defined by the user (**ALLO**CATE 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, …, a_n 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_0 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 6: **Specification of Variables as Distance-based Interpolated Function from a Table**

SYNTAX **SET** **{DIST} [SQUA | LINE] {Φ_1, …, Φ_n } {Nsets} {V_1,..V_m | fname} [option] [mod] [ALWA] [subrgn] [FIEL]**

DIST The Φ_n variables are computed from a inverse distance based interpolation function (here ξ_i is i^{th} value in the specified table for the corresponding variable and $x_{k,i}$ are its coordinates):

$$\Phi = \sum_i \frac{\xi_i / r_i}{1/r_i} \quad \text{or} \quad \Phi = \sum_i \frac{\xi_i / r_i^2}{1/r_i^2} \; ; \quad r_i^2 = \sum_k \left(x_k - x_{k,i}\right)^2$$

SQUA The inverse distance squared interpolation is used. This is the default option.

LINE The inverse distance based interpolation is used.

Φ_1, …, Φ_n 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. See also below. There is no default value.

V_1….V_m 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_1,** through **Φ_n**. The number of rows of the table must equal **Nsets**.

fname The name of the file (see Section 7.2.2) from which **V_1** through **V_n** are read.

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 ID=MIDDLE 3 sets of U V T file 'SPATIAL' ALWAYS ADD POSITIVE to existing

MODE 7: Specification of Values from Total Inventory

SYNTAX SET {Φ} {INVE} [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 6.7.1, 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.

INVE The total inventory or amount of the property, Φ, is specified. The inventory, Q, for a subregion is defined as:

$$Q = \sum_{subrgn} \alpha_i \, \Phi_i \, \delta V_i$$

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:

$$\Phi_i = \theta \, \Phi_i^o + \frac{1}{\alpha_i} \frac{Q}{\sum_{subrgn} \delta V_i}$$

Here Φ^o 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:

$$\Phi_i = \theta \, \Phi_i^o + \frac{Q}{\sum_{subrgn} \alpha_i \delta V_i}$$

SCAL The inventory is distributed proportional to the existing value of Φ::

$$\Phi_i = \theta \, \Phi_i^o + \frac{Q \, \Phi_i^o}{\sum_{subrgn} \alpha_i \Phi_i^o \delta V_i}$$

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.

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.

COMMENTS

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 8: Transfer of Inventory From One Subregion to Another

SYNTAX SET {Φ} {INVE} {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 6.7.1, except the pressure variables, plus the user defined variables (**ALLO**CATE command) for which a differential equation is solved. There is no default value; a symbol must be specified.

INVE 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.

β A constant (**β** ≥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 **β** specifies the fraction of the current inventory in the donor region that is transferred. Limits are imposed so that (0 ≤ **β** ≤ 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 **β**.

TIME **β** specifies the fraction per unit time of the current inventory that is transferred. The actual fraction transferred is equal to **β** δt where δt is the time step. Limits are imposed dynamically so that (0 ≤ **β** δt ≤ 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**ys modifier for the same **Φ**, and the same donor and receptor subregions is disabled.

COMMENTS

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 INVEntory of C from ID=DONOR to ID=RECEPTOR                    ! Total amount transferred
SET INVEntory of C from ID=DONOR to ID=RECEPTOR FRACtion=0.5              ! 50% transferred
SET INVEntory of C from ID=DONOR to ID=RECEPTOR   0.1 TIME ALWAYS   ! 0.1*dt every time step
SET INVEntory of C from ID=DONOR to ID=RECEPTOR   OFF               ! previous command off
```

MODE 9: 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 **ALLO**CATE command.

Φ Symbol for one of the previously defined field variables listed in Table 7.2.4. 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
ABS or **ABSO**	The absolute value for the computed gradient is taken.
POSI	The negative values for the computed gradient are set of zero
NEGA	The positive values for the computed gradient are set of zero
SQUA	The function is computed as the square of the gradient
ROOT	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 **BLOC**K 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 10: **Specification of Multiple Variables from Table of Values at Vertices**

SYNTAX **SET { VERT } { Φ_1, …, Φ_n } { V_1,..$V_{(n+1)m}$ | fname} [subrgn] [FIEL] { dir | BOUN }**

VERT The Φ_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.

Φ_1, …, Φ_n Symbols to denote the variables that are to be set by the command. The symbols include all those listed in Table 6.8.1 plus the user defined variables (**ALLO**CATE 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….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 7.2.2) 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 7.2.5 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 11: **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

COMMAND SOLID

PURPOSE To specify material and thermal properties of the solid for conjugate heat transfer.

SYNTAX SOLI [N1, N2, N3]

N1 The density of the solid material. The default value is 1.

N2 The specific heat of the solid material. The default value is 1.

N3 The thermal conductivity of the solid material. The default value is 0.

COMMENTS

By default temperature variations in the solid (constituting the internal obstacles in the flow field) are ignored in heat transfer calculations. This command activates the option of a solid with finite rate of heat transfer.

EXAMPLES

SOLId: stainless steel: properties: rho = 7810, cp = 0.46, k=0.016

COMMAND 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 [Φ] [MANU] [N1, N2, N3, N4, N5]

Φ By default, an appropriate set of governing equations is solved based on the user-specified input. However, the user may override this default by explicitly specifying one or more symbols to select the corresponding dependent variable for which the governing equations are solved. The valid symbols are listed in Table 6.7.1.

MANU The time step is determined according to the user-specified input of **N2** through **N4**. This is the default mode.

N1 The incremental time period **(> 0)** for solution of the governing equations which is added to any previously specified value. The default value is 0.

N2 The time step **(> 0)** at the start of the current segment of calculations. The time step may be changed during the calculations or by a subsequent **SOLV**E command. The default value is **N1**/1000.

N3 The geometric ratio multiplier **(> 0)** for the time step. Each successive time step is multiplied by this value until a maximum value specified by **N4** is reached. The default value is 1.

N4 The maximum permissible time step. The default value is 10^{30}.

N5 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, max=1000, min=1, dfac=1.1, 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 [Φ] {AUTO} [PASS] [N1, N2, N3, N4, N5, N6, N7]**

Φ See Mode 1.

AUTO The time step is automatically determined so that the specified primary convergence constraint is achieved for each time step (See **CONV**ERGENCE 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 **CONV**ERGENCE command with **FLOW** modifier and decreased if it is larger than $4/5^{th}$ 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.

N1 See Mode 1.

N2 See Mode 1.

N3 The geometric ratio multiplier **(> 0)** for the time step. Each successive time step is multiplied by this value until a maximum value specified by **N4** is reached. The default value is 1.01

N4 The maximum permissible time step. The default value is 10^{30}.

N5 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}.

N6 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 **CONV**ERGENCE command. The default value is 2.

N7 The maximum number of time steps. The default value is 99 999 999.

COMMENTS

The **AUTO**matic 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 have 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 mode
SOLVe AUTO 10 days DT=0.2, fac=1.3, mx=1, mn=1.E-5, df=1.1, Nmax=1000
SOLVe 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 [Φ] {CFL} [N1, N2, N3, N4, N5, N6, N7, N8]**

Φ 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:

$$R = \max\left[\frac{\max\ |A_{ij}|_{i\neq j}}{A_{jj}}\right]_{\text{over all } i} \leq R_{max}$$

where A_{ij} are the coefficients of the solution matrix: $A_{ij}\ X_j = B_i$

This criteria 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.

N1 See Mode 1 of command.

N2 See Mode 1 of command.

N3 See Mode 1 of command.

N4 The maximum permissible time step. The default value is 10^{30}.

N5 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^{-7}.

N6 The cut-off threshold value, R_{max}, for the CFL parameter. If the computed value of R exceeds the threshold value then new time step is computed from:

$$dt_{new} = \frac{dt_{old}}{dt_{fac}}; \quad dt_{fac} = \max\left(\frac{R}{N6}, N7\right)$$

where N7 is defined below. The default value is 0.8

N7 The minimum geometric ratio divisor (>0) for the time step. The default value is 1.1.

N8 The maximum number of time steps. The default value is 99 999 999.

COMMENTS

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, factor=1.0, max=1, min=1.E-5, Rmax=0.5 DTfac_min=1.5 Nmax=1000
SOLVe for T only for 10 seconds in CFL automatic mode with DT=0.2

MODE 4: **Steady State Solution Mode**

SYNTAX **SOLV {STEA} [Φ] [N1, N2]**

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 By default, an appropriate set of governing equations is solved based on the user-specified input. However, the user may override this default by explicitly specifying one or more symbols to select the corresponding dependent variable for which the governing equations are solved. The valid symbols are listed in Table 6.7.1.

N1 The maximum number of iterative steps for solution of the matrix. The default value in this instance is 99999999.

N2 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 5: **Frequency of Computation for Selected Variables**

SYNTAX **SOLV {FREQ} [Φ] [N1]**

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.

Φ **One or more** symbols to specify the variables for which the solution frequency is explicitly specified. Valid symbols are listed in Table 6.7.1. There is no default value.

N1 The frequency, in terms of number of steps of the solution process, for the solution of the variables identified on the command.

EXAMPLES

SOLVe FREQency for C1 is every 10 steps
SOLVe FREQency for FU, CO and C2 every 20 steps

MODE 6: **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.

COMMENTS

The Mode 1 and 2 of the **SOLV**E 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 **SOLV**E command specified for each segment. See Section 6.10 for more information.

EXAMPLES

SOLVe OFF
SOLVe OFF now to obtain initial output for problem verification.

COMMAND SOOT

PURPOSE To activate soot computation and specify soot constants and parameters.

MODE 1: Soot Particle Constants

SYNTAX **SOOT [N1, N2, N3, N4]**

N1 Soot particle density. The default value is 2000 (kg/m^3).

N2 Diameter of the smaller soot particles in microns. The default value is 1/40 micron.

N3 Diameter of the larger soot particles in microns. If this value is ≥ 0 or is omitted, then the computations are performed with only one soot particle size specified by **N2**. The default value is 1 micron.

N4 Percentage of the smaller soot particles in the total. This input is ignored if **N3** is zero or is omitted. The default is 90 percent.

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} [N1, N2, ..., N6]**

NUCL Soot nuclei constants are specified.

N1 The Arrhenius prefactor **(>0)** for the soot nuclei equation. The default value is 1.16×10^{31}.

N2 The Arrhenius activation energy constant divided by the universal gas constant **(>0)** for the soot nuclei equation. The default value is 9.00×10^{4}.

N3 The 1st exponent **(≥0)** of the soot nuclei equation. The default value is 1,000.

N4 The 2nd exponent of the soot nuclei equation. The default value is 8.00×10^{-16}.

N5 The 3rd exponent of the soot nuclei equation. The default value is 1.00×10^{15}.

N6 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} [N1, N2, , N7]**

FORM Soot formation constants are specified.

N1 The Arrhenius prefactor **(>0)** for the soot formation equation. The default value is 1.5626×10^{13}.

N2 The Arrhenius activation energy constant divided by the universal gas constant **(>0)** for the soot nuclei equation. The default value is 1.60×10^{4}.

N3 The 1st exponent **(≥0)** of the soot formation equation. The default value is 1.81.

N4 The 2nd exponent of the soot formation equation. The default value is –0.50.

N5 The 3rd exponent of the soot formation equation. The default value is –1.94.

N6 The incipient carbon-to-oxygen ratio at which the soot formation starts. The default value is 0.1.

N7 The incipient temperature (in absolute units) at which soot formation starts. The default value is 400.

EXAMPLES

SOOT FORMation constants are 1.90E13, 1.50E4, 2.0, -0.5, -2.0, 0.1, 500.
SOOT FORMation constants: 1.90E13, 1.50E4, 2.0, -0.5, -2.0, 0.1, 500.

MODE 4: **Soot Computation Frequency**

SYNTAX **SOOT {FREQ} [N1]**

FREQ Frequency of soot computations is specified.

N1 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 FREQency by default.

COMMAND 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]] [option] [subrgn] [dir] {N1} [fname | N2, .,Nn] [Nn+1, Nk], [ρ_B]

Φ The symbol for the dependent variable for which the source is specified. Valid symbols are listed in Table 6.7.1. 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 **VOLU**ME or **AREA** modifiers are present.

func One of the modifiers listed in Table 7.2.3 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 7.2.4. If no independent variable is specified, the variable is assumed to be time.

option Options selected for implementation of the source.

option	INTERPRETATION
VOLU	In the absence of the **TOTA**L modifier, the source for each element is computed as: **Q = q δV.** Here **q** is the amount specified by the user and **δV** is the volume of the element. The **q**, in turn, is computed from **func(ξ)** and **N1** through **Nn**. If the **TOTA**L modifier is present, the amount for each element is computed as: **Q = q δV /V**, where **V** is the volume of the total **subrgn**.
AREA	In the absence of the **TOTA**L modifier, the source for each element is computed as: **Q = q δA,** where **δA** is the area of the element boundary indicated by **dir**. If the **TOTA**L modifier is present, the source for each element is computed as: **Q = q δA /A**, where **A** is the total area of the **subrgn** in the **dir** direction.
INTE	By default, if **dir** 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 **INTE**RNAL modifier is present then the wall treatment and wall diffusive flux are retained.
NORM	In the absence of the **TOTA**L modifier, the source, **Q**, is computed as: $Q = q \sum_i A_i \bullet V_i$ where A_I is the i^{th} direction component of the element boundary area specified by **dir**. V_I are the values specified by **Nn+1** through **Nk** (2 for 2D, and 3 for 3D). In the presence of the **TOTA**L modifier, **Q** is computed in a manner identical to that for the **AREA** modifier.
DENS	The computed source, **Q**, is further multiplied by density. The density may be specified as the last value, ρ_B, on the command. If this value is omitted, then the boundary value at the node indicated by the **dir** direction is used if the **AREA** or **NORM**AL modifier is present, otherwise the local density for the element is used. If the **dir** 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 **NORM**AL modifier is present. See Section 7.2.5 for available choices. There is no default value for this input.

fname The name of the file from which numerical values **N2** through **Nn** are read. This option can be used only if the selected function is a table or one of the series functions. See Section 7.2.2 for further information.

N1, .., Nn The numerical constants and coefficients for the selected function. See Section 7.2.7 for more details. There are no default values for this input.

Nn+1, .., Nk The normalizing vector components, V_i, if the **NORM**AL modifier is present. Two values must be specified for 2D and 3 for 3D flows. There are no default values for this input.

ρ_B The density value that multiplies the computed source. It can only be specified if the **DENS**ITY modifier is present and then it must be the last value on the command. If no value is specified but the modifier **DENS**ITY is present, then default value is the boundary value at the node indicated by the **dir** direction if the **AREA** or **NORM**AL modifier is present, otherwise the default value is the local density for the element.

EXAMPLES

Generic examples for this command are given in Section 7.2.8. The command keyword (**SOUR**CE) 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]] [option] [subrgn] [dir] {N1} [fname | N2, .,Nn] [Φ=Nn+1, .., Φ=Nm] [Nm+1, Nk] [ρ_B]**

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.

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.

N1, .., Nn See Mode 1 specification.

Nn+1, .. Nm Each of these represents the value of injected variable denoted by the symbol immediately preceding the value.

Nm+1, .., Nk The normalizing vector components, V_i, if the **NORM**AL modifier is present. Two values must be specified for 2D and 3 for 3D flows. There are no default values for this input.

ρ_B See Mode 1 specification.

COMMENTS

The **SOUR**CE **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 **INLE**T 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 **INLE**T 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 **INTE**RNAL 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.

SOURce with FLOW injection: amount = 0.001 with T=50, C=1.
SOURce with FLOW per unit AREA of X- face: = 0.001 with T=50, C=1.
SOURce with FLOW withdrawal: amount =-0.001 per second
SOURce 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
SOURce for FLOW: EXPOnential series with TIME 7 sets from 'SOURCE' T=100, C=0.
SOURce FLOW q=10 X- direction T=100, U=20. NORMalized velocity 0., 1.5, 2.5 ID=VSOURce;
SOURce FLOW q= -10 in X- dir with NORMalized vel 1., 1.5, 2.5 and DENSity for ID=VSOURce
SOURce 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:
SOURce 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 3: **Flow Injection with Fixed Variable Values and Computed Momentum Components**

SYNTAX **SOUR {MOME} [func [ξ]] [TOTA] [option] [subrgn] {dir} {N1} [fname | N2, .,Nn] [Φ=Nn+1, .., Φ=Nm] [Nm+1, Nk] [ρ_B]**

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^{th} direction , ρ_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 **NORM**al 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^{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.

TOTA See Mode 1 specification

option See Mode 1 specification. Unless the **NORM**al modifier is present, it is assumed that the **AREA** modifier is in effect. Any specification of the **VOLU**me modifier is ignored. All other modifiers can be used as for Mode 1 Specification.

subrgn See Mode 1 specification.

dir See Mode 1 specification. This modifier must be present for this mode of command.

fname See Mode 1 specification.

Φ See Mode 1 specification.

N1, .., Nn See Mode 1 specification.

Nn+1, .. Nm These values represent the value of injected variable denoted by the symbol immediately preceding the value.

Nm+1, .., Nk The normalizing vector components, V_i, if the **NORM**al modifier is present. Two values must be specified for 2D and 3 for 3D flows. There are no default values for this input.

ρ_B The density value to compute the velocity components. It can only be specified if the **DENS**ity 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.

COMMENTS

The **SOUR**CE **MOME**TUM command is similar to the **SOUR**CE **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 **SOUR**CE **FLOW** command.

EXAMPLES

All the examples cited for Mode 2, except those with **VOLU**me modifier, are applicable provided that the modifier **MOME**ntum 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[ξ]] [TOTA] [subrgn] {dir} {N1}[fname|N2,..,Nn] [Φ=Nn+1,..,Φ=Nm] [Nm+1, …, Nm+5] [DENS | SPEE] [ρ_B| V_S]**

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 ρ_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.

TOTA See Mode 1 specification

subrgn See Mode 1 specification.

dir See Mode 1 specification. This modifier must be present for this mode of command.

fname See Mode 1 specification.

Φ See Mode 1 specification.

N1, .., Nn See Mode 1 specification.

Nn+1, .. Nm The values of injected variables denoted by the symbols immediately preceding the values.

Nm+1 The spherical angle, Θ 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.

Nm+2 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.

Nm+3, Nm+5 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_B, is specified as the last value on the command.

ρ_B See Mode 3 specification.

SPEE The injection speed, **V_S**, is specified as the last value on the command.

V_S The value **V_S** if the **SPEE**D modifier is present. There is no default value for this input.

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.

SOURce q=10., X- direction with ID=VSOURce; T=100, K=0.05, L=0.02, Theta=45, phi=20, axis=1., 0, 0
SOURce q=10., X- ID=VSOURce; T=100, K=0.05, L=0.02, Theta=45, phi=20, axis=1., 0. 0, SPEEd=120
SOURce 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} {Φ} {TABL} {MULT} [option] [subrgn] [fname] [N1, .., Nn]**

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.

Φ **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.

fname See Mode 1 specification.

N1 The number of sets of data for the tabular functions.

N2, ..Nn The N1 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 Φ 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.

EXAMPLES

SOURce 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

SOURce with FLOW 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 6: **Solubility-Limited Source for a Chemical Species**

SYNTAX **SOUR {Φ} {SOLU] [func [ξ]] [subrgn] [fname] {N1,..., Nn} {Nn+1} [Nn+2}**

Φ 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 Equation 3.10.3.

func The function that specifies the solubility limit, C_s, for the species. See Mode 1 specification.

ξ See Mode 1 specification.

subrgn See Mode 1 specification.

fname See Mode 1 specification.

N1, .., Nn See Mode 1 specification.

Nn+1 Total amount of material to be dissolved. The default value is 0.

Nn+2 Time at which release begins. The default value is 0.

EXAMPLES

All examples cited for Mode 1 are applicable provided that the modifier "SOLU" is added and the values for the variables (Nn+1 and Nn+2) 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 {Φ} {RADI} [COEF] [VARI] {H, Φ_{eq}} [POSI | NEGA] [option] [subrgn]

Φ 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 Φ in appropriate units, H is a transfer coefficient and Φ_{eq} is an equilibrium value of Φ. The H and Φ_{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.

Φ_{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.

POSI The source is implemented only if Q < 0.

option See Mode 1 specification. **NORM**al and **DENS**ity options are not available with this type of source

subrgn See Mode 1 specification.

EXAMPLES

SOURce for T: RADIation type coefficient 0.001, equilibrium value = 15 degrees.
SOURce for T: RADIation .hcoef = 0.001 value from variable EQVAlu !EQVA is a symbol
SOURce for T: RADIation VARI COEF as function HVALu , value=15 !HVAL is a symbol
SOURce for T: RADI VARI COEF HCOEf & HVALu !HCOE & EQVA are symbols

MODE 8: **Source Term as a Linear Decay or Half-Life.**

SYNTAX **SOUR {Φ} {DECA} [HALF] [subrgn] {N1}**

Φ 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 Φ as:

$$Q = -\lambda \rho \Phi,$$

In this equation, Q is the source rate for **Φ**, λ is the decay rate and Ψ is the fluid density if the variable Φ is defined in terms of mass units and unity if it is defined in terms of volumetric units.

LIFE Half-life of the property **Φ** is specified. The decay rate is computed from the half life, σ:

$$\lambda = -\frac{\log(0.5)}{\sigma}$$

subrgn See Mode 1 specification.

N1 The decay rate, λ, or half-life, λ, for the variable **Φ**.

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 {Φ} {REAC} {id_1, . . ., id_n} [subrgn] {N1, .., Nn}**

Φ 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 chemical species according to the equation:

$$Q = \sum_{n} C_n R_n \ ,$$

In this equation, Q is the source for **Φ**, C_n are the scaling constants and R_n are the previously specified reactions.

Id_1,.., id_n Identifiers for the reactions, R_n , described by Mode 2 of **REAC**TION command, which comprise the source. Up to 9 reactions may be specified.

subrgn See Mode 1 specification.

N1, .., Nn The scaling constants, C_n , 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 **Φ** per second). The number of constants must match the number of reactions specified.

EXAMPLES

SOURce REACtion type FU LINEar -1. * R1
SOURce REACtion type CH LINEar -1. * R2 + 0.9586 * R1 in SELEcted subdomain.
SOURce REACtion type CH LINEar -1. * R2 + 0.9586 * R1 for ID=RRGN

MODE 10: **Flux Transfer Between Adjacent Elements**

SYNTAX **SOUR {FLUX} {Φ} {VALU|DIFF|GRAD} [AREA | NORM|VOLU] {dir} [subrgn] [func [ξ]] {N1} [fname | N2, .,Nn] [Nn+1, Nk] [DENS] [ρ_B]**

Φ 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 (ξ)** and **N1** through **Nn**). The factors F_ϕ, F_V and F_ρ 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 $F_\phi = \phi_D$, where ϕ_D is the value of ϕ 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 ϕ_B and ϕ_F are the values of ϕ in the "boundary" and "field" cells, respectively. The field cell is that defined by the **subrgn** 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_V 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_V is set to unity.

NORM $F_V = \sum A_i V_i$, where A_i is the projected area of the interface in the direction of the i^{th} coordinate denoted by **dir** modifier, and V_i are the user inputs, **Nn+1** through **Nk**. The summation is over 2 terms for 2D, and 3 for 3D simulations.

VOLU The factor F_V is set equal to the volume of the receptor cell.

subrgn See Mode 1 specification.

func See Mode 1 specification.

ξ See Mode 1 specification.

fname See Mode 1 specification.

N1, .., Nn See Mode 1 specification.

Nn+1, .., Nk See Mode 1 specification.

DENS F_ρ is equal to **ρ_B**. In the absence of this modifier, F_ρ is equal to unity.

ρ_B The density value that multiplies the computed source. It can only be specified if the **DENS**ITY modifier is present and then it must be the last value on the command. If no value is specified but the modifier **DENS**ITY is present, then default value is the boundary value at the node indicated by the **dir** direction if the **AREA** or **NORM**AL 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 **SOUR**CE commands for the identified subregion are deactivated for the specified variable. A new specification may follow.

subrgn See Mode 1 specification.

EXAMPLES

SOURce OFF for T for most recentlu SELEcted region
SOURce OFF for T for ID=MIDDle

COMMAND SPECIFIC

PURPOSE To specify the specific heat of the fluid.

MODE 1: Generic Functional Form for Specific Heat

SYNTAX SPEC { func [ξ] } [phase] [subrgn] { fname | N1 ..., Nn }

func One of the modifiers listed in Table 7.2.3 that denotes the functional form of the fluid specific heat. If no function is specified then the value is assumed to be constant.

ξ One of the independent variables listed in Table 7.2.4. If no variable is specified, then the independent variable is assumed to be time.

phase The phase for which the input is specified. **See Section 7.2.10 for available options.** By default the input pertains to the 1st 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 Sections 7.2.3 and 7.2.4. If no subregion is specified, the entire computational domain is selected.

fname The name of the file containing the numerical values N1 through Nn. This option can be used only if the selected function is a table or one of the series functions. See Section 7.2.2 for additional information.

N1, .., Nn The numerical constants and coefficients for the selected function. See Section 7.2.7 for more details. There are no default values for this input.

COMMENTS

The specific heat may also be set by the **SET** Command with **CP** modifier. In this case the modifier **ALWA**YS must be specified if the specific heat 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. In general, the **SPEC**IFIC command is recommended as the preferred mode for input.

EXAMPLES

SPECific heat = 1234

Generic examples for this command are given in Section 7.2.8. The command keyword (**SPEC**IFIC) must replace the keyword used in these examples and the dependent variable (Φ) must be omitted. Only the functional form, the independent variable and the numerical values need to be specified.

MODE 2: **Pre-Defined Specific Heat Function**

SYNTAX **SPEC {GORD } [fname]**

GORD The specific heat is computed from the 4th 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).

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

SPECific heat from GORDon-McBride formulae with default library values
SPECific heat by GORDon-McBride on file called 'CPHTDATA.JP4'

COMMAND **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_1, .,Nn}**

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.

N1, .., Nn 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 alter 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

MODE 2: **Store A Local Grid Transformation for Later Use**

SYNTAX **STAC {TRAN} [MATR | PLAN | POIN] {N1, .,Nn} {Nn+1, .,Nm} [N7, .,N9]**

TRAN 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 **FREEFORM™** commands that allow the use of the STACk modifier.

MATR The 9-component transformation matrix (**T**) between the local (**x**) and the global (**X**) 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 **j** varies the fastest. Each row of the transformation matrix is interpreted as a unit vector directed along the $\mathbf{x}_i$ direction. This is the default option.

PLAN 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 $\mathbf{x}_1$ to be the unit normal to the plane, $\mathbf{x}_2$ is computed by intersecting the plane with the bounding box of the domain, and $\mathbf{x}_3$ is computed as the cross product of $\mathbf{x}_1$ and $\mathbf{x}_2$. The user has no control over the directions of $\mathbf{x}_1$ and $\mathbf{x}_2$. However, they are guaranteed to be in the specified plane and mutually perpendicular.

POIN The plane of transformation is specified by the coordinates of 2 points. The plane is assumed to pass through the 2nd point and the normal points from the 1st to the 2nd point. The transformation is computed in the same way as the previous option.

N1, .., Nn The 1st through 3rd components of the transformation matrix if the **MATR**IX 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 **PLAN**E modifier is present and the coordinates of the 1st point if the **POIN**T modifier is present. There is no default value.

Nn+1, .., Nm The 4th through 6th components of the transformation matrix if the **MATR**IX modifier is present, a vector (or direction cosines) to specify the normal to the plane of transformation if the **PLAN**E modifier is present and the coordinates of the 2nd point (on the plane of transformation) if the **POIN**T modifier is present. There is no default value.

N7, .., N9 The 7th through 9th components of the transformation matrix if the **MATR**IX 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 **WRIT**E command for the specified variable is scaled by a and b according to:

$$\Phi_{out} = a\,\Phi + b$$

Φ A symbol that denotes the variable for which the output is required. The valid symbols include those listed in Table 6.8.1 plus the user defined variables (**ALLO**CATE 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 **STAC**K **WRIT**E 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.

COMMENTS

Up to 10 **STAC**K 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.

COMMAND STATISTICS

PURPOSE To compute and obtain output of the statistics for a dependent variable for a selected subregion.

SYNTAX **STAT {Φ} [subrgn] [OFF] [fname] [NO] [TABL] [V_{frq}] [TIME]**

Φ A symbol for the dependent variable for which the statistics are required. The computed statistics consist of the minimum, maximum, mean and standard deviation, and the location of the minimum and the maximum. The valid symbols are listed in Table 6.8.1. One and only one character string must be specified for each command.

subrgn The subregion for computation of statistics. If no subregion is specified, the entire domain is selected.

OFF Computation of statistics for the specified variable and subregion is deactivated.

fname The file name for output. The default file name is 'acr_STATS.TMP'. 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 the default file. The total number of open files can not exceed 64. A summary of output is also printed to the standard output device at the end of simulations.

NO This modifier is effective only in the presence of the **TABL** modifier. It disables some default features of the command.

TABL A tabulated summary of the computed statistics appears on the standard output file at the end of simulations. If the **NO** and the **TABL** modifiers are present, then this output is suppressed.

V_{frq} The frequency (step or time interval) of output. See Section 7.2.11 for further details. The default value is set so that output is obtained at every step of simulations.

TIME By default, **V_{frq}** is interpreted to be the frequency of computations in terms of number of steps. If this modifier is present, then **V_{frq}** is interpreted to be the time interval between successive computations.

COMMENTS

The user should perform these computations only at the required frequency since frequent computations will results in increased computation time.

EXAMPLES

STATistics for U for the entire domain
STATistics for T for SELEcted region every 20 steps
STATistics for T for SELEcted region every 20 steps with NO TABLes of output
STATistics for T for subregion defined by ID=VAULt every TIME=0.50 units
STATistics for T for ID=VAULt OFF !!! Switch off previously specified STAT command
STATistics for U to 'FLUX.OUT' for SELEcted region at TIME interval of 0.4

COMMAND STORAGE COEFFICIENT

PURPOSE To specify the STORage coefficient for the governing differential equations.

MODE 1: STORage Coefficient as a General Function

SYNTAX STOR {Φ} {func (ξ} [subrgn] {fname | N1 ..., Nn}

Φ A symbol for the dependent variable for which the storage coefficient is specified. Valid symbols are listed in Table 6.7.1. 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 α is the storage coefficient, ρ is the fluid density and Φ is dependent variable.

subrgn The subregion for which the input is specified. See Sections 7.2.3 and 7.2.4. If no subregion is specified, the entire computational domain is selected.

func One of the modifiers listed in Table 7.2.3 that denotes the functional form of the input. For this input, the function specifies the value of the appropriate storage 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 7.2.4. If no variable is specified, then the independent variable is assumed to be time.

fname The name of the file containing the numerical values **N2** through **Nn**. This option can be used only if the selected function is a table or one of the series functions. See Section 7.2.2 for additional information.

N1,.., Nn The numerical constants and coefficients for the selected function. See Section 7.2.6 for more details. There are no default values for this input.

COMMENTS

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 (see Chapter 2). **This mode of the command is currently implemented only for the PORFLOW™ Software Tool and then only for the pressure and species equations.**

EXAMPLES

Generic examples for this command are given in Section 7.2.8. 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 6.7.1. There is no default value; at least one symbol must be specified.

DYNA A typical storage term, S_Φ, 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 α, ρ and Φ or the product (ρ α Φ) can be stored at the end of n^{th} time step and re-used at the $n+1^{th}$ step. When α and ρ are constants, the two procedures lead to identical results. However, if α and ρ are functions of other variables, then depending on the method and sequence of solution of equations, α and ρ may be revised after Φ^n was computed. That is, the prevailing values of α and ρ may have been used to compute Φ^n at the n^{th} time step and then modified by subsequent operations. For strict compliance with balance equations the product of (α ρ Φ) used at the n^{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 $n+1^{th}$ step. This remvoes the need for storage arrays to store the previous product and also accounts for dynamic changes in α and ρ. However, the strict mass balance of the Φ variables may not be achieved if α and ρ 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 { Φ } { OFF | ON }**

Φ One or more symbols for the dependent variables for which the input is specified. Valid symbols are listed in Table 6.7.1. 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.

COMMENTS

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 α and ρ vary with time, then product (α ρ Φ) must be stored at the end of n^{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 $(\rho\ \alpha\ \Phi)^n$ is recomputed from the most recent current values of the variables at the start of the $n+1^{th}$ step.

MIN Same as **MINI**.

EXAMPLES

STOR term allocate MINImum storage

COMMAND SWIRL

PURPOSE To specify the a transformation of the velocity components from an induced swirl.

MODE 1: Swirl induced by Vanes at Fixed Angles or by Solid Body Rotation

SYNTAX SWIR { ANGL | SOLI } [LOCA | FLUX | SOUR] [subrgn] [dir] {N1} {N2, .,Nn} [Nn+1]

ANGL Swirl is induced in the manner of the flow passing vanes at a fixed angle. With α 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 ω 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 ρ 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 ρ 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 **SOUR**CE 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 **LOCA**TE **MATC**H command, then the transformation is applied only to the 2nd 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 7.2.5 for available choices. There is no default value for this input. A value must be supplied unless the **subrgn** was defined by a **LOCA**TE **PAIR** or **LOCA**TE **MATC**H command.

N1 The vane angle, α, in degrees if the **ANGL**E modifier is specified or the angular velocity of rotation, ω, if the **SOLI**D modifier is specified. The numerical value of α must be less than 89.9. There is no default value for this input.

N2, Nn 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.

Nn+1 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^{30}.

COMMENTS

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] {N1} [fname| N2, .,Nn] [Nn+1,..,Nm] [Nm+1]**

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.

fname The name of the file from which numerical values **N2** through **Nn** are read. This option can be used only if the selected function is a table or one of the series functions. See Section 7.2.2 for further information.

N1, .., Nn The numerical constants and coefficients for the selected function. See Section 7.2.7 for more details. There are no default values for this input.

Nn+1, ..,Nm 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.

Nm+1 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^{30}.

EXAMPLES

Generic examples for this command are given in Section 7.2.8. The command keyword (**SWIR**L) must replace the keyword and the dependent variable (**Φ**) 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

MODE 3: **Swirl Profile or Mapping from a data file**

SYNTAX **SWIR [PROFile] [RADIal] {subrgn} {dir} {N1,N2,N3,N4} {fname} [SCALe [NONE] N5,N6] [AXIS N7,N8,N9] [ALWAys] [FUEL = N10] [DIAGnostics] ($\Phi_1..\Phi_N$)**

PROF Implies that a profile of the velocity components and other specified variables be read in from a data file.

RADI **Implies a radial swirler. The default is an axial swirler.**

For an axial swirler, the axial velocity component is scaled to match the mass flow. 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. *The velocity profile data is assumed to be a function of the radius*.

For a radial swirler, the radial velocity component is scaled to match the mass flow. 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 *The velocity profile data is assumed to be a function of the axial distance*.

subrgn 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.

dir The orientation index for the element boundary associated with the input. See Section 7.2.5 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**

fname **The name of the file from which profile data are read in**. This file is mandatory. The file consists of an **optional** header block ending with a "**End Header**" string starting from the first column, followed by columns of data, one per variable. The first four data columns are the *independent variable* (axial distance for radial swirler / radial distance for axial swirler), *axial velocity*, *radial velocity* and the *swirl velocity*. 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.

N1 **The mass flow through the subregion**. This subregion should be defined as an INLET or an OPEN or DIRICHLET boundary. There are no default values for this input.

N2, N3, N4 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.

SCALE 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. *In the absence of this modifier, the min value is set to zero and the max value to the actual maximum radial extent of the sub-region for an axial swirler, and to the actual min and max values of the axial extent of the sub-region for a radial swirler*. The independent variable is thus scaled to lie between 0 and 1.

SCALE NONE Turns off all scaling. The independent variable is not scaled prior to doing the interpolation.

N5, N6 The min and max scale factors for the transformed coordinate for the **SCALE** modifier. The coordinate is transformed as ***r = (R –N5) /(N6 –N5)***. **Must be specified if the SCALE modifier is present without the NONE modifier**. In the absence of the **SCALE** modifier, these values are computed internally as described above.

AXIS The axis of the transformation between the global 3D Cartesian system and the local cylindrical system applicable to the defined subregion**. For an axial swirler, this modifier and the associated data may be omitted**, in which case it is computed as the normal pointing into the mesh from a suitable boundary node**. For a radial swirler this input and the associated data is mandatory**.

N7, N8, N9 The three components of direction vector of the axis of the local cylindrical coordinate system. **There are no default values for a radial swirler**. For an axial swirler, in the absence of the **AXIS** modifier, these are computed as the normal pointing into the mesh from a suitable boundary node.

FUEL 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.

N10 The fuel mass flow rate if the **FUEL** modifier is specified. There is no default.

$\Phi_1..\Phi_N$ Optional list of scalar variables also to be interpolated from the profile data. (except pressure)

ALWA 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 **SOLV**E command. This behavior is analogous to the **SET** vs. **SET ALWA**ys 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.

DIAG This modifier causes detailed diagnostics to be output. In the absence of the **DIAG**nostics modifier, only a brief summary is output for each swirler.

COMMENTS

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 PROFline 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+ ALWAys 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
End Header							
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 ALWAys detailed DIAGnostics on.

The contents of file 'swirl3-r.dat':

Swirler mapping file for radial swirler:

X	U	Ur	Ut	T	K	L	FU
End Header							
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

COMMAND 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 7.2.5 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.

COMMENTS

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

SYMMetry at Y- boundary
SYMMetry at Y- for ID=B_SYMMETRY
SYMMetry OFF for Y- and ID=B_SYMMETRY

COMMAND TIME

PURPOSE To set the initial simulation time for a problem.

SYNTAX TIME { N1 }

N1 The starting time (≥0) for simulations. The default value is 0.

EXAMPLES

TIME = 50 years at start of simulations

COMMAND TITLE

PURPOSE To specify the problem title.

SYNTAX TITL followed by character information

COMMENTS

The specification must be restricted to one 80 character record. Title specification is included in all output files generated by the ACRi Software.

EXAMPLES

TITLe ILLUSTRATIVE PROBLEM - DEFAULT SET UP - 07/01/93:ACRi/akr

COMMAND 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] {N1, ..Nn} [Nn+1, Nn+2, Nn+3]

fname The file name for output. The default file name is 'acr_TRACK.TMP'. 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 the default file. The total number of open files can not exceed 64. 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
X	Particle tracking stops if its x-coordinate exceeds the Nn+1th numerical value.
Y or **R**	Particle tracking stops if its y-coordinate exceeds the Nn+1th numerical value.
Z or **THETA**	Particle tracking stops if its z-coordinate exceeds the Nn+1th numerical value.
DIST	Particle tracking stops when its distance from the point of release exceeds the Nn+1th numerical value. This is default option if **STOP** modifier is specified.
TIME	Particle tracking stops when the time exceeds the Nn+1th numerical value.
ELAP	Particle tracking stops when the elapsed time from its moment of release exceeds the Nn+1th numerical value.
FREE	Particle tracking stops if it reaches a free surface or zone of saturation.

N1, .., Nn The coordinates of the starting location of the particle. Two values are required for the 2D and 3 for 3D input modes.

Nn+1 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.

Nn+2 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.

Nn+3 The frequency index for providing the output in the particle track file. The output is obtained every **Nn+3** (≥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

TRACks for particle start at: (35.0, 5.27)
TRACks for particle start at: (1.22, 10.0, 19.3) print TABLes also
TRACk particle at: (1.22, 10.0) print TABLes 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)
TRACks at: (35.0, 5.27) STOP TIME 200. start at 0 yrs; output every 20 steps
TRACks for particle start at: (1.22, 10.0, 19.3) STOP at FREE surface.
TRACks at: (35.0, 5.27) STOP ELAPsed TIME 100. on file "PARTICLE.TRK"

MODE 2: Particle Tracking Factors

SYNTAX TRAC {FACT} [subrgn] [MULT | DIVI] {N1}

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 Sections 7.2.3 and 7.2.4. If no subregion is specified, the entire computational domain is selected
MULT	Particle velocity in the selected region is multiplied by **N1**. This is the default option.
DIVI	Particle velocity in the selected region is divided by **N1**.
N1	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

COMMAND **TRANSFER**

PURPOSE To specify the transfer of mass flux and fluid properties from one subregion to another.

MODE 1: **Direct Transfer of Computed Flux**

SYNTAX **TRAN [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 **LOCA**TE command with **MATC**hed 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 1st surface ("donor") of a matched pair of surfaces is transferred to the 2nd 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 **LOCA**L 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 **LOCA**TE **MATC**H command. A subregion must be defined; there is no default value.

EXAMPLES

TRANsfer FLUXes from 1st surface (entrance) of subregion ID=Q_TRANSFER to the 2nd surface (exit)
TRANsfer FLUXes from 1st surface of SELEcted subregion to the 2nd with LOCAL momentum

MODE 2: **Transfer of Computed Amount as Flux or Source**

SYNTAX **TRAN {FLUX|SOUR} [func(ξ)] [LOCA] [subrgn] [TOTA] [VOLU | AREA] [INTE] {N1} [fname | N2, .,Nn]**

FLUX 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 1st of a matched pair of surfaces and transferred to the corresponding elements just downstream of the 2nd surface. This is the default mode.

SOUR Similar to **FLUX** modifier except that the mass is transferred as a source rather than as flux.

func One of the modifiers listed in Table 7.2.3 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 7.2.4 If no independent variable is specified, the variable is assumed to be time.

LOCA See Mode 1 Specification.

subrgn The sub region for flux or source transfer. If the **SOURCE** modifier without **AREA** is present, then the sub region must have been previously defined by a LOCATE **MATCh** or a **LOCATE CORRelated** command. Otherwise the sub region must have been previously defined by a **LOCATE MATCh** command. A subregion must be defined; there is no default value.

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 quantity over whole of the **subrgn**. In this case, the source is distributed equally to all the elements of the **subrgn** unless the **VOLUme** or **AREA** modifiers are present.

VOLU If the **TOTAL** modifier is absent, the source for each element is computed as: **Q = q δV.** Here **q** is computed from **func(ξ)** and **N1,…,Nn** and, **δV** is the volume of the element. If the **TOTAL** modifier is present, the amount for each element is equal to: **Q = q δV / V**, where **V** is the volume of the total **subrgn**.

AREA If **TOTAL** modifier is absent, the source for each element is computed as: **Q = q δA;** if **TOTAL** modifier is present, the source is computed as: **Q = q δA / A.** Here **δA** is the area of the element boundary and **A** is the total area of at the 1st (inlet) surface.

INTE 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 **INTE**rnal modifier is present then the wall treatment is retained and the wall diffusive flux is included in the computations.

fname The name of the file from which numerical values **N2** through **Nn** are read. This option can be used only if the selected function is a table or one of the series functions. See Section 7.2.2 for further information.

N1, .., Nn The numerical constants and coefficients for the selected function. See Section 7.2.7 for more details. There are no default values for this input.

EXAMPLES

Generic examples for this command are given in Section 7.2.8. The keyword **TRAN**SFER must replace the keyword and the dependent variable (Φ) 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 1st surface (entrance) of subregion ID=Q_TRANSFER to the 2nd 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 **TRAN {DRAG} [FLUX | SOUR] [LOCA] [subrgn] [VOLU | AREA] [REVE] [INTE] {N1} [N2, N3]**

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 1st of a matched pair of surfaces and transferred to the corresponding elements just downstream of the 2nd surface. In all other respects command mode is identical to Mode 1. The flow rate is computed as:

$$q = \rho\, \delta A \left[\frac{1}{2\, C_D} \frac{P_{in} - P_{out}}{\rho} \left(\frac{\rho D}{\mu} \right)^{\beta} \right]^{\frac{1}{2-\beta}},$$

where:

- ρ is the fluid density
- δA is area of a segment (each element boundary) of the 1st surface through which the flow is extracted,
- C_D is the drag coefficient,
- P_{in} 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_{out} 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_D 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 **LOCA**TE 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 1st surface to the 2nd surface irrespective of the pressure gradient. If P_2 exceeds P_1 the source is set to zero. If this modifier is present then reverse flow is allowed if the pressure gradient is from the 2nd to the 1st surface.

N1 The drag coefficient, C_D. There is no default value for this input.

N1 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.

N3 The exponent β for the drag coefficient. The default value is zero.

COMMENTS

This mode of the TRANSFER command is automatically selected if the DRAG modifier is present and there are no more that 3 numerical values on the command.

EXAMPLES

TRANsfer DRAG C_drag = 0.10 for subregion ID=Q_TRANSFER
TRANsfer as FLUX DRAG C-drag = 0.5 Length scale = 0.2 beta=0.25 ID=Q_TRANSFER
TRANsfer DRAG as SOURCE: CD = 0.10 Hydraulic Dia = 0.2 beta =0.25 by VOLUme ID=Q_TRANSFER
TRANsfer DRAG as FLUX: CD = 0.10 Hydraulic Dia = 0.2 beta =0.25 by AREA ID=Q_TRANSFER
TRANsfer 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 **TRAN {DRAG} [FLUX | SOUR] [LOCA] [subrgn] [VOLU | AREA] [REVE] [INTE] {N1, N2, N3, N4} [N5]**

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 1st of a matched pair of surfaces and transferred to the corresponding elements just downstream of the 2nd surface. The flow rate is computed as:

$$q = \rho N \frac{\pi D^2}{4} \left[\frac{1}{2 C_D} \frac{P_{in} - P_{out}}{\rho} \left(\frac{\rho D}{\mu} \right)^{\beta} \frac{D}{L} \right]^{\frac{1}{2-\beta}},$$

where:

- ρ is the fluid density,
- N is the number of holes,
- D is the diameter of holes,
- C_D is the drag coefficient,
- P_{in} 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_{out} is the average pressure across the outlet surface of the sub region, and
- 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_D 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.

N1 The drag coefficient, C_D, for the holes. There is no default value for this input.

N2 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.

N3 Total number of holes, N. There is no default value for this input.

N4 Average length of the holes, L. There is no default value for this input.

N5 The exponent β for the drag coefficient. The default value is zero.

COMMENTS

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

TRANsfer DRAG C_drag= 0.10 , dia= 0.001, number=200, Length= 0.005, for subregion ID=Q_TRANSFER
TRANsfer DRAG C_drag= 0.10 dia= 0.001, number=200, L= 0.005, beta=0.25 ID=Q_TRANSFER
TRANsfer DRAG C_drag= 0.10 dia= 0.001, n=200, L= 0.005, distribution by VOLUme ID=Q_TRANSFER
TRANsfer DRAG C_drag= 0.10 dia= 0.001, n=200, L= 0.005, distribution by AREA ID=Q_TRANSFER
TRANsfer DRAG C_drag= 0.10 dia= 0.001, n=200, L= 0.005, Allow REVErse flow; ID=Q_TRANSFER

COMMAND TURBULENCE

PURPOSE To select the appropriate turbulence model and specify the empirical constants.

MODE 1: Selection of Two Equation Turbulence Model

SYNTAX TURB {model} [LOWR] [COMP]

model The turbulence model to be used. By default the standard k-ε model is used.

model	INTERPRETATION
CHIE	Chien (1982) low Reynolds number model is used.
CUBI	Cubic k-ε model of Shih et al. (1997) is employed.
LRR	Launder, Reece and Rodi (1975) Reynolds stress model is used.
LS	Launder-Sharma (1974) low Reynolds number model is used.
MENT	Menter (1996) two layer blended k-ε / k-ω model is employed.
QUAD	Quadratic k-ε model of Shih et al. (1997) is employed.
RNG	RNG enhancements to the k-ε model are employed.
RSM	Reynolds stress model of Craft et al. (1996) & Iocovides et al. (1996) is used
SSA	So and Sarkar (1997) low Reynolds number model is used.
SST	Menter (1996) low Reynolds number shear-stress transport model is used.
WILC	Wilcox (1993) k-ω model is employed.
YSM	Yang and Shih (1993) low Reynolds number model is used.

LOWR 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 **QUAD**ratic and **CUBI**C models, the Yang and Shih (1993) low Reynolds option is used.

SECO This modifier is effective only in the presence of the **CUBI**C modifier. In this case, a second option of the Yang & Shih (1993) low Reynolds number enhancement is selected.

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.

COMMENTS

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 LOWReynolds number extensions
TURBulence model k-e with CHIen low Reynolds number extensions
TURBulence model LRR version of rsm
TURBulence model RSM of launder & Craft

MODE 2: Selection of LES Turbulence Model

SYNTAX TURB {LES} [SMAG | DEDU | DYNA] [C_s]

SMAG Smagorinsky (1963) Large Eddy Simulation Model is used. This is the default option.

DEDU The 2nd order Deductive LES of Lee and Meecham (1996) is used.

DYNA 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 C_s is ignored. However, if the modifier **SMAG**orinsky is simultaneously present, then specified value of the C_s constant is added to the value computed by the model.

C_s The Smagorinsky constant C_s (>0) . The default value is 0.05.

COMMENTS

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 DYNAMIC LES
TURBulence model is DYNAMIC LES with default value of SMAGorinsky constant
TURBulence model is DYNAMIC LES with SMAGorinsky constant Cs=0.04

MODE 3: **Empirical Constants for Turbulent Flows**

SYNTAX **TURB [N1, N2, N3, N4, N5]**

N1 The constant κ (>0) of the log law of wall (Equation 3.11.8). The default value is 0.4187.

N2 The constant E (>0) of the log law of wall. The default value is 9.793.

N3 The constant C_μ (>0) of the k - ε model (Equation 3.2.2). The default value is 0.09.

N4 The constant C_{e1} (>0) of the k - ε model (S_F term in Table 2.2.2). The default value is 1.44.

N5 The constant C_{e2} (>0) of the k - ε model (s_F term in Table 2.2.2). The default value is 1.92.

COMMENTS

The default values are for the 'standard" k-e 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

COMMAND USER

PURPOSE To specify the user identification for the problem.

SYNTAX USER followed by character information

COMMENTS

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

COMMAND **VISCOSITY**

PURPOSE To specify the viscosity and to select associated options.

MODE 1: **Generic Functional Form for Viscosity**

SYNTAX **VISC {func [ξ]} [phase | EFFE] [ADD] [subrgn] {fname | N1 ..., Nn}**

func One of the modifiers listed in Table 7.2.3 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.

ξ One of the independent variables listed in Table 7.2.4. If no variable is specified, then the independent variable is assumed to be time.

phase The phase for which the input is specified. **See Section 7.2.10 for available options**. By default the input pertains to the 1st phase of the fluid. This modifier is available only for the multi-phase versions of the **ANSWER™** Software Tool.

EFFE The input is applied to the effective or mixture viscosity of the fluid. By default, the specified viscosity is assumed to be molecular viscosity of the fluid. For a turbulent flow, the effective mixture viscosity is computed as the sum of the molecular and turbulent viscosity. If this modifier is present, then the specified values replace any internally computed values.

ADD The input is added to the current value of viscosity. This modifier is effective only if the **EFFE**CTIVE modifier is also present.

subrgn The subregion for which the input is specified. See Sections 7.2.3 and 7.2.4. If no subregion is specified, the entire computational domain is selected.

N1, .., Nn The numerical constants and coefficients for the selected function. See Section 7.2.7 for more details. There are no default values for this input.

COMMENTS

This mode of the VISCOSITY command is currently available only for ANSWER™ Software Tool

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 **ALWA**YS 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 **VISC**OSITY command is recommended as the preferred mode of input.

The effective viscosity may also be set by the **COND**UCTIVITY 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 **VISC**OSITY, **SET** and **COND**UCTIVITY commands may be specified in combination. In this case, the **COND**UCTIVITY commands will take precedence over all other commands.

EXAMPLES

VISCosity molecular value = 2.E-5

Generic examples for this command are given in Section 7.2.8. The command keyword (VISC) must

replace the keyword used in these examples.

MODE 2: **Pre-Defined Viscosity Functions**

SYNTAX **VISC { [CONS | POLY | EXPO | USER] | [NODE] } [FIEL] [phase] [μ*] [α] [β] [γ]**

CONS The fluid viscosity is constant. This is the default option.

POLY polynomial variation in fluid viscosity according to the equation:

$$\frac{\mu}{\mu^*} = 1 + \alpha\ (T^* - T)\ + \beta\ (T^* - T)^2 + \gamma\ (T^* - T)^3$$

In this equation T* is a reference fluid temperature that can be set with the **REFE**RENCE command.

EXPO Exponential variation in fluid viscosity according to the equation:

$$\frac{\mu}{\mu^*} = \exp\left[\frac{\alpha}{T + T_a}\right]$$

In this equation T_a is the base to convert temperature to absolute units. It can be set with the **REFE**RENCE command.

USER The viscosity is specified by a user specified function.

NODE The viscosity is specified for each node by a separate command such as **INIT**IAL, **READ** or **SET** command. In such a case, this command can be used to modify the specified value according to the one of the relations specified on this command.

FIEL By default the viscosity at the boundaries is computed from the specified relation or option selected by the user. If this modifier is present, then the boundary value of viscosity is set equal to the field value at the nearest neighbor.

phase The fluid phase for which the input is specified. See Section 7.2.10 for available options. If no phase modifier exists, the input is assumed to be for the 1st phase of the fluid.

μ* The reference viscosity. The reference value is used only to determine the K_j of the 2nd and 3rd phases from the K* specified for the primary phase by the HYDRaulic command. For the primary phase, only the viscosity ratio, rather than the absolute viscosity value, is required. The default value is 1 for all phases.

α The coefficient in the viscosity relations. The default value is 1436 K if the modifier **EXPO**NENTIAL is present; otherwise, it is zero.

β, γ The coefficients in the **POLY**NOMIAL equation. The default value is 0 for both.

COMMENTS

This mode of the VISCOSITY **command is currently available only for PORFLOW™ Software Tool**

EXAMPLES

VISCosity UNIFORM for primary fluid: mu* = 1.002E-3 (water at 20 deg C)
VISCosity UNIForm SECOndary phase: mu* = 8E-2 (Engine oil at 20 deg C)
VISCosity EXPOnential option: mu* = 0.544E-3, b = 1450 K for FIELD values
VISCosity POLYnomial option: mu* = 8.0E-4, a=1.E-4, b=5.E-5, c=0.
VISCosity NODE basis and multiply by POLYnomial: mu* = 8.0E-4, a=1.E-4, b=5.E-5, c=0.
VISCosity UNIForm for THIRd phase: mu* = 1.E-2

COMMAND WALL

PURPOSE To specify wall boundary or boundary conditions for a problem.

MODE 1: Specify Walls at Undefined Outer Boundaries

SYNTAX WALL [ORTH] [MOVI]

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 **BOUN**DARY command.

COMMENTS

If this command is present then all the outer boundaries of the domain of computation that are not explicitly defined through the **INLE**T, **OPEN**, **OUTL**ET, **PERI**ODIC and **SYMM**ETRY 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 (**ADIA**BATIC 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; I walls ORTHogonal
WALL by default at undefined external boundaries; ORTHogonal and MOVIng

MODE 2: **Specify Boundary Conditions at Walls**

SYNTAX **WALL {Φ} [FIX | ADIA | GRAD | EXTR]**

Φ	One or more symbols that denote the dependent variables for which the wall boundary conditions are specified. Valid symbols are listed in Table 6.7.1. 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 **BOUN**DARY, **INIT**IAL, **READ** or **SET** commands
ADIA	The gradient of the specified variables at the wall is set to zero.
GRAD	Same as **ADIA**BATIC; 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.

COMMENTS

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 **BOUN**DARY and **ADIA**BATIC 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]**

dir The orientation index for the wall. See Section 7.2.5 for available choices. There is no default value; a value must be specified.

subrgn The subregion to be identified as a wall. See Sections 7.2.3 and 7.2.4. 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 **BOUN**DARY, **INIT**IAL, **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 **BOUN**DARY command.

ADIA The gradient of the specified temperature at the wall is set to zero.

COMMENTS

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 **BOUN**DARY, **INIT**IAL, **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 7.2.5 for available choices. There is no default value; a value must be specified.

subrgn The subregion to be removed from an existing wall. See Sections 7.2.3 and 7.2.4. If no subregion is specified, the outermost **dir** oriented boundary of the entire computational domain is selected.

COMMENTS

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

MODE 5: **Drill a Hole in a Previously Specified Wall**

SYNTAX **WALL {HOLE} {dir} [subrgn] {N1} [N2, . Nn] [IJK | ELEM] [Nn+1,..Nm] [Nm+1]**

HOLE Wall segments falling within the range specified by the input are removed from the wall previsouly specified by the same **dir** and **subrgn** modifiers. The hole is circular or elliptic based on the specification of the direction normals for the plane.

dir The orientation index for an existing wall. See Section 7.2.5 for available choices. There is no default value; a value must be specified.

subrgn The subregion designation of an existing wall. See Sections 7.2.3 and 7.2.4. If no subregion is specified, the outermost **dir** oriented boundary of the entire computational domain is selected.

N1 The diameter of the hole to be "drilled" in the specified wall.

N2, . Nn 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 circular hole. In the presence of **IJK** or **ELEM** modifier these are interpreted as given below.

IJK The numerical input **[N2, . Nn]** 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 This option can be used only for structured grids. Only the internal elements can be specified.

ELEM The numerical input **[N2, . Nn]** specifies the element which is at the center of the hole. Only 1 value must be specified. This option can be used for structured or unstructured grids. Only the internal elements can be specified.

Nn+1, . Nn+3 The components of a vector normal to the plane in which the center of the hole is located. The values may be in arbitray units and are internally converted to unit normals. Up to 3 values may be specified. At least one value must be specified. **An elliptic hole can be drilled by appropriate choice of the normal vector.**

Nm+1 The tolerence 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 tangentail distance is ≤ to the radius of the hole, then the wall segment is removed **provided its normal distance from the plane is less than the tolerance**. By default the tolerance is set to 10^{30} 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. If specified, the intervening values for all 3 components of the unit normal (even if zero) must be specified.

COMMENTS

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

COMMAND **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 {Φ} [STAC] [fname] [subrgn]**

Φ One or more of the symbols that represent the variables for which output is desired. The valid symbols are listed in Tables 6.8.1 and 6.8.2. There is no default value.

STAC The output variable is scaled by coefficients a and b specified on a previous **STAC**K **WRIT**E command according to:

$$\Phi_{out} = a\ \Phi_{computed} + b$$

The STAC modifier is effective only for **real** variables. It is ignored for integer variables. It is also ignored if no previous **STAC**K **WRIT**E command was specified.

fname The name of the file to which the output is directed. See Section 7.2.2 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.

COMMENTS

This command provides an alternative mode of output in comparison with the **OUTP**UT 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.

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 {Φ} [option] [walltype] [SUMM] [STAC] [subrgn] [dir] [fname]**

Φ One or more of the symbols that represent the variables for which output is desired. The valid symbols are those listed in Table 6.8.1 and Table 6.8.2. 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
INLE	Only the boundaries specified by the INLET command are selected.
OUTL	Only the boundaries specified by the OUTLET command are selected.
OPEN	Only the boundaries specified by the OPEN command are selected.
IO	All boundaries specified by INLET, OUTLET or OPEN command are selected.
BOUN	All external boundaries of the specified (or default) subregion are selected.
SYMM	Only the boundaries specified by the SYMMETRY command are selected.
WALL	Only the walls are selected.

walltype The type of wall to be selected for output if the **WALL** modifier is present.

walltype	INTERPRETATION
ALL	All walls are selected. This is the default option.
EXTE	Only the exterior walls of the computational domain are selected.
INTE	Only the walls located in the interior of the computational domain are selected.
BLOC	Only the walls of the blocks (BLOCK 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.

STAC 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 7.2.5 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 SUMMary 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 6.8.1 and Table 6.8.2. Up 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
LINE	The values are computed by inverse linear distance interpolation from the computed values at nearest neighbors. This is the default option.
SQUA	The values are computed by inverse distance squared interpolation from the computed values at nearest neighbors.
AVER	The values are computed by arithmetic average of computed values at nearest neighbors.
NEAR	The value is set equal to that at the nearest neighbor.
X	Same as **LINE** except that the distance is set equal to the separation in the x-coordinate.
Y	Same as **LINE** except that distance is set equal to the separation in the y-coordinate.
Z	Same as **LINE** 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 **WRIT {Φ} {BLOC} [HEAD] [FIEL] {fname}**

Φ One or more of the symbols that represent the variables for which output is desired. The valid symbols are those listed in Table 6.8.1, and the node or element-based variables listed in Table 6.8.2. 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 1st element.

HEAD By default only the numeric values for the selected variable(s) are written to the file without any header information. If the **HEAD**ER 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 **BLOC**K 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 7.2.2 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 7.2.2 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^{th} element.

fname The name of the file to which the output is directed. See Section 7.2.2 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 **CONN**ECTIVITY command.

fname The name of the file to which the output is directed. See Section 7.2.2 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'

(This page left intentionally blank.)

REFERENCES

Abramzon, B. and W.A. Sirignano, 1989. *Droplet Vaporization Model for Spray Combustion Calculations,* International Journal of Heat and mass Transfer, 32, pp. 1605-18.

Bittener, J.D. 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.

Bray, K.N.C., 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.

Candel, S., D. Veynante, F. Lacas, E. Maistret, N. Darabiha and T. Poinsot, 1990. *Flamelet Descriptions of Turbulent Combustion.* Invited Lecture, 9th International Heat Transfer Conf., Aug. 19-24, 1990, Jerusalem, Israel.

Chien, K-Y., 1982. *Predictions of Channel and Boundary-Layer Flows with a Low-Reynolds-Number Turbulence Model*, AIAA, **20**, 1, p 33-38.

Chorin, A.J., 1967. *A Numerical Method for Solving Incompressible Viscous Flow Problems, J. Computational Physics, Vol. 2, pp. 12-26.*

Craft, T.J., N.Z. Ince and B.E. Launder, 1996. *Recent developments in second-moment closure for buoyancy-affected flows*. Dynamics ofAtmospheres and Oceans, **23**, p.99-114.

DeRis, J., 1979. *Fire Radiation - A Review*, Seventeenth Symposium (International) on Combustion, The Combustion Institute, Pittsburgh, PA.

Edelman, R., J. Boccio and G. Weilerstein, 1973. *The Role of Mixing and Kinetics in Combustion Generator NO_x*, AIChE Sym. on Control of NO_x Emissions in Direct Combustion Power Sources.

Edwards, D.K. and A. Balakrishnan, 1973. *Thermal Radiation by Combustion Gases,* Int. J. Heat Mass Transfer, 16, pp. 25-40.

Gaydon, A.G. and H.G. Wolfhard, 1979. *Flames*, Fourth Edition, Chapman and Hall Ltd., London.

Gordon, S. 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.

Hamaker, H.C., 1947. *Radiation and Heat Conduction in Light-Scattering Material,* Phillips Research Report, Volume 2, pp. 55-67.

Harten, A., 1983. *High Resolution Schemes for Hyperbolic Conservation Laws. Journal of Computational Physics, Vol. 49, pp. 357-393.*

Harten, A., 1984. *On a Class of High Resolution Total-Variation-Stable Finite-Difference Schemes,* SIAM J. Num. Anal., Vol. 21, pp. 1-23.

Hautman, D.J., 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.

Haynes, B.S. and H.G. Wagner, 1981. Progress in Energy and Combustion Science, 7, 229 pp.

Homann, K.H., 1967. *Carbon Formation in Premixed Flames,* Combustion and Flame, 11, pp. 265-287.

Iacovides, H., 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.

Jones, W. P., and J. H. Whitelaw, 1982. *Calculation Methods for Reacting Turbulent Flows: A Review.* Combustion and Flame, **48**, p 1-26.

Karki, K.C., 1986. *A Calculation Procedure for Viscous Flows at All Speeds in Complex Geometries,* Ph.D. Dissertation, Univ. of Minnesota.

Lander, B.E. 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.

Launder, B.E., 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.

Launder, B.E. and D.B. Spalding, 1972. *Lectures in Mathematical Models of Turbulence, Academic Press.*

Leckner, B., 1972. *Spectral and Total Emissivity of Water Vapor and Carbon Dioxide,* Combustion and Flame, 19, pp. 33-48.

Lee, S., H-J. Kim and A.K. Runchal, 2003., Large Eddy Simulation of Unsteady Flows in Turbomachinery, 2003, submitted to the J. Power and Energy.

Lee, S. 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.

Leonard, B.P., 1979. *A Stable and Accurate Convective Modeling Procedure based on Quadratic Upstream Interpolatio,* Comp. Methods Appl. Mech. Eng., 19,59-98.

Magnussen, B.F., B.H. Hjertager, J.G. Olson and D. Bhaduri, 1978. *Effects of Turbulent Structure and Local Concentrations on Soot Formation and Combustion in C_2H_2 Diffusion Flames,* Seventeenth Symposium (International) on Combustion, The Combustion Institute, Pittsburgh, PA.

Menter, F.R., 1996. *A Comparison of Some Recent Eddy-Viscosity Turbulence Models.* J. Fluids Eng., ASME, **118**, p 514-519.

Modak, A.T., 1979. *Radiation from Products of Combustion,* Fire Research, 1, pp. 339-361.

Mongia, H.C. 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.

Nagle, J. and R.F. Strickland-Constable, 1962. *Oxidation of Carbon Between 1000-2000°C,* Proc. Fifth Conf. on Carbon, Volume I, Pergamon Press, New York.

Palmer, H.B. 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.

Patankar, S.V. 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.

Perry, R.H. and C.H. Chilton, 1973. *Chemical Engineers' Handbook, 5th Edition,* McGraw Hill, New York.

Pope, S.B., and S.M. Correa, 1986. *Joint PDF Calculations of non-Equilibrium Turbulent Diffusion Flame,* Paper # 53, 21st Sym. (International) on Combustion, Munich, August 3-8.

Ranz, W.E. and W.R. Marshall, 1952. *Evaporation from Drops,* Chemical Engineering Progress, 48, pp. 141-46 and pp. 173-180.

Roache, P.J., 1972. *Computational Fluid Dynamics,* Hermosa Publishers, Albuquerque, New Mexico.

Runchal, A.K., 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.

Runchal, A.K., 1977. *Comparative Criteria for Finite Difference Formulations for Problems of Fluid Flow*, Int. J. Num. Methods in Eng., 11, pp. 1667-1679.

Runchal, A.K., 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.

Runchal, A. K., 1987b. *CONDIF: A Modified Central-Difference Scheme for Convective Flows,* Int'l J. Num. Methods in Engg., 24. 1593-1608.

Sarkar, A. and So, R.M.C., 1997. *A critical Evaluation of Near-Wall Two-Equation Models Against Direct Numerical Simulation Data*, Int. J. Heat & Fluid Flow.

Shih T.-H., 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.

Siddall, R.G., 1972. *Flux Methods for the Analysis if Radiant Heat Transfer,* Fourth Symp. on Flames and Industry, British Flame Research Committee and the Institute of Fuel, Imperial College, London, September.

Smagorinsky, J., 1963. General Circulation Experiments with the Primitive Equations, Part I: the Basic Experiment, *Monthly Weather Rev.*, **91**, pp. 99-164.

Spalding, D.B., 1971. *Mixing and Chemical Reaction in Steady Confined Turbulent Flames,* Thirteenth Symposium (International) on Combustion, The Combustion Institute, Pittsburgh, PA.

Spalding, D.B., 1972. *A Novel Finite-Difference Formulation for Differential Expressions Involving Both First and Second Derivatives,* Int. J Num, Methods Eng., 4, p. 551-559

Srivatsa, S.K., 1983. *Computation of Soot and NO_x Emissions from Gas Turbine Combustors,* NASA CR-167930, NASA, Cleveland, OH.

Street, J.C. and A. Thomas, 1955. *Carbon Formation in Pre-Mixed Flames,* Fuel, 34, pp. 4-36.

Varga, R.S., 1962. *Matrix Iterative Analysis,* Prentice Hall, Englewood Cliffs, New Jersey.

Wilcox, D.C., 1993. *Comparison of Two-Equation Turbulence Models for Boundary Layers with Pressure Gradient.* AIAA, **31,** 8, p 1414-1421.

Yang, Z. 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

PARTIAL LIST OF PUBLICATIONS

ANSWER™ has been extensively used over the last 30 years. More than 100 publications and project reports on the benchmarking, verification and application of ANSWER™ are currently available. This appendix presents a partial list of these publications.

D. B. Schein and W.C. Meecham, 2001. Computation of Jet Noise at Realistic Mach and Reynolds Numbers Using Large-Eddy Simulation and Lighthill's Analogy. FEDSM2001-18144, ASME 2001 Fluids Eng. Division Summer Meeting, May 29-June 1, New Orleans, LA.

G**.N. Kumar and H. Mongia, 2000.** Results Of A DOE On the Film Cooling Effectiveness Of a Modern Combustor with Machined Ring Liners. AIAA-2000-2235, 36th AIAA/ASME/ASEE Joint Propulsion Conference, July 2000, Huntsville, AL

G. N. Kumar and H.C. Mongia, 2000. Assessment Of Advanced Turbulence Models And Unstructured Code for Calculating the Film Effectiveness Of a Modern Film-Cooled Combustor. AIAA-2000-0333, AIAA 38th Aerospace Sciences Meeting, January 10-13, Reno, NV.

G. N. Kumar and H.C. Mongia, 2000. Validation Of Near Wall Turbulence Models For Film Cooling Applications In Combustors. AIAA-2000-0480, AIAA 38th Aerospace Sciences Meeting, January 10-13, Reno, NV.

A. K. Runchal and M. M. Rao, 2000. Dynamically Programmable and User Friendly CFD Software for Industrial Applications. Invited Lecture – Intn'l Sym. on Challenges and New Directions in Computation of Internal Flows, IIT Madras, Chennai, Jan 7-8.

V. Sundararajan, 2000. CFD Experiences in Aero Engine Development. Invited Lecture – Intn'l Sym. on Challenges and New Directions in Computation of Internal Flows, IIT Madras, Chennai, Jan 7-8.

Grant Swenson, 1999. Numerical Simulations of Combustion Instabilities in Gas Turbine Combustors with Applications. Ph.D. Thesis, California Institute of Technology, Pasadena, CA.

G.N. Kumar and H. Mongia, 1999, 'Validation of Turbulence Models for Wall Jet Computations as Applied to Combustor Liners. AIAA-99-2250, 35th AIAA/ASME/ASEE Joint Propulsion Conference, Los Angeles, California, June 20-24, 1999.

D. B. Schien and W.C. Meecham, 1999. Hot Jet Noise Computed from Time-Accurate Simulation of Compressible Free Jet Turbulence. Sixth International Congress on Sound and Vibration, July,1999, Copenhagen, Denmark

S. Lee, A.K. Runchal and J-O Han, 1999. Subgrid-Scale Modeling in Large-Eddy Simulation and its Application to Flow about Yawed Cylinder and Cavity Flows. 3rd ASME/JSME FED Conf., Washington, D.C.

S. Lee, J-S Choi, J-O Han and A.K. Runchal, 1999. A Large-Eddy Simulation of the Turbulent Flow in the Vicinity of an Upright Wall-Mounted Half Cyliner and Aerodynamic Sound Prediction. 3rd ASME/JSME Joint FED Conf., Washington, D.C.

H.C. Mongia, 1998. Aero-Thermal Design and Analysis of Gas Turbine Combustion Systems: Current Status and Future Direction. Paper # AIAA 98-3982, 34th AIAA/ASME/SAE/ASEE, Propulsion Meeting, July 13-15, 1998, Cleveland, OH.

T.J. Held and H.C. Mongia, 1998. **Emissions Modeling of Gas Turbine Combustors using a Partially-Premixed Laminar Flamelet Model. Paper# AIAA 98-3950, 34th AIAA Propulsion Meeting, July 13-15, 1998, Cleveland, OH.**

T.J. Held and H.C. Mongia, 1998. "Application of a Partially Premixed Laminar Flamelet Model to a Low Emissions Gas Turbine Combustor," ASME Paper No. 98-GT-217, Intern'l Gas Turbine & Aeroengine Congress.

T.J. Held and H.C. Mongia, 1998. "Emissions Modeling of Gas Turbine Combustors Using a Partially-Premixed Laminar Flamelet Model," AIAA Paper No. 98-3950

Robert Goler, 1998. Effect of Temperature Gradients on Flow Around a Building**.** M.S. Thesis, The Univ. of Adelaide, Dept of Applied Math., Adelaide, Australia.

D.B. Schein, 1998. Exhaust mixing noise computed from time-accurate simulation of compressible free jet

turbulence. Paper 3pEA3, 136th Meeting, Acoustical Society of America, Norfolk, Virginia.

S. Lee, 1998. Numerical Study of Aerodynamic Noise from a Rotating Cylinder Using Large-Eddy Simulation. 2nd Year Report to KOSEF.

John S. Lang, 1998. Formulating Design Criteria for Hydraulic jet Mixing by Modeling. J. American Water Works Association. To be published.

D.B. Schein, 1997. Large-eddy simulation of compressible free jet turbulence applied to computation of exhaust mixing noise. Paper 4pNS3; 134th Meeting, Acoustical Society of America, San Diego, CA.

D. B. Schein and W.C. Meecham, 1997. Computation of Jet Noise Using Large-Eddy Simulation and Lighthill's Analogy. ASME 4th Intn'l Sym. on Fluid-Structure Interaction, Aeroelasticity, Flow-Induced Vibration and Noise, AD-Vol. 53-**1**. 439-446. IMECE, Dallas, Texas

S. Lee and W.C. Meecham, 1996. Computation of Noise from Homogeneous Turbulence and from a Free Jet. Int. J. Acoustics & Vibration, **1**, 1, 35-47.

S. Lee 1993. Computation of Aerosound from Turbulent Flow Fields Using Large-Eddy Simulation Proc. Noise '93 Conference p. 619-624, 1993 Student paper Prize from INCE.

S. Lee and W.C. Meecham, 1993. Quadrupole Directivity of Noise from Homogeneous Turbulence and Turbulent Free Jet as Examples. J. Acoustical Soc. Am. To be published.

S. Lee and W.C. Meecham, 1993. The Numerical Prediction of Noise Generated by a Jet Impinging on a Plate Using Large-Eddy Simulation and the Lighthill Analogy. ASME WAM, NCA-Vol. 15/FED-Vol. 168, p. 101-108, Flow Noise Modeling, Measurement, and Control, New Orleans, December.

R.C. Hall, 1993. Heathrow Tunnels CFD Analysis of Ventilation. WS Atkins Science & Technology, Woodcote Grove, Ashley Road, Epsom, Surrey KT18 5BW, U.K.

A.K. Runchal and S.K. Bhatia, 1993. Flow and Heat Transfer in a Three-Dimensional Cubical Cavity. Second U.S. Congress on Computational Mechanics, August 16-18, Washington, D.C.

A.K. Runchal and S.K. Bhatia, 1993. ASME Benchmark Study: ANSWER Predictions for Backward Facing Step and Lid-Driven Cubical Cavity. CFD Triathlon, ASME FED Summer Conference, June 20-24, Washington, D.C.

S. Lee and W.C. Meecham, 1992. A Computation of Turbulent Flow Field for a Square jet and far Field Aerosound Using Large-Eddy Simulation. Bulletin American Physical Society**, 37**, 8, Nov. 1992

S. Lee and W.C. Meecham, 1992. A Large-Eddy Simulation of Aerosound Using Homogeneous Turbulence as Example. J. Acoustical Soc of America, **92**, 4,Part 2, 124th Meeting, New Orleans, 31Oct-4 Nov.

Seungbae Lee, , 1992. Subgrid-Scale Modeling in Large-Eddy Simulation and its Application to Aerosound. Ph.D. Thesis, Mechanical Engineering, University of California, Los Angeles.

A.K. Runchal, 1992. ANSWER: A Benchmark Study for Backward Facing Step. 1992 WAM, ASME, Anaheim, CA, Nov. 8-13.

M.A. Zook and A.K. Runchal, 1992. Application of Computational Fluid Modeling to the Development of Semiconductor Chemical Vapor Deposition Systems. J. Vacuum Sci. Technology, A 10(4), p. 850-855.

D. Rafinejad, J. Monkowski, L. Wright and A.K. Runchal, 1991. Prediction of Flow and Heat Transfer in a Novel Thermal CVD Reactor. Comp. Techniques and Numerical Heat Transfer on PCs and Workstations, p. 85-92, ASME Book # H00674, Ed. D.W. Pepper, A.P. Emery and M.B. Kelleher.

S. Candel, D. Veynante, F. Lacas, E. Maistret, N. Darabiha and T. Poinsot, 1990. Flamelet Descriptions of Turbulent Combustion. Invited Lecture, 9th International Heat Transfer Conf., Aug. 19-24, 1990, Jerusalem, Israel.

C.E. Janeke, L. Bester and A.K. Runchal, 1990. AIRLINES: A Computational Tool for HVAC Design, To Appear.

P.K.C. Wang, E. Behaegel and A.K. Runchal, 1990. Iterative Solution of a Free-Boundary Problem Arising in Microscopic Particle Manipulation Inside a Liquid Layer. 9th Intn'l Conf. on Analysis and Optimization of Systems, June 12-14, 1990, Antibe, France, Springer Verlag, Berlin.

L.S. Caretto and A. K. Runchal 1989. Ramjet Combustor Mathematical and Engineering Design. Paper # 89-2799, AIAA/ASME/SAE/ASEE 25th Joint Propulsion Conference, July 10, 12, 1989, Monterey, California.

L.S. Caretto and A. K. Runchal 1989. Capabilities of Personal Computers for Numerical Convective Heat Transfer. ASME National Heat Transfer Conference, Aug. 6-9, 1989, Philadelphia.

M. Cazalens and M. Desaulty, 1989. Implantation dans le Code ECRIN de Technique de Capture de Chocs (FCT). Report # YKC 4685/89, SNECMA, Paris, France.

P. Guevel, O. Hembise, G. Barrot, J.F. Couchouron and T. Godot, 1989. Optimisation des formes d'entree du bassin d'essais des chaluts de Boulogne sur Mer. To appear

G. Karadimas, 1989. Application of Computational Systems to Aircraft Engine Components Development. 9th Intn'l Symp. on Air Breathing Engines, Sept. 3-8, 1989, Athens, Greece.

E. Maistret, N. Darabiha, T. Poinsot, D. Veynante, F. Lacas, S. Candel and E. Esposito, 1989. Recent developments in the coherent Flamelet Descriptions of Turbulent Combustion. Numerical Combustion, Springer Verlag, Berlin, pp. 98-117.

R. Shekhar and J.W. Evans, 1989. Mathematical Modeling of Flow in Pachuca (Air-Agitated) Tanks: Masters Thesis, Part II. Dept. of Materials Science and Mineral Engineering, UC Berkeley, CA.

O. Hembise and G. Barrot, 1988. Singularite en T: MVI Technicatome. Report 88.61.73. Principia Recherche Developpement, S.A., Sophia Antipolis, Nice, France.

O. Hembise and G. Barrot, 1988. Adaptation d'un 3D d'ecoulement Turbulent Reactif. SNECMA Contact # 265.898 VY, Phase II Final Report by Principia Recherche Developpement, S.A., Sophia Antipolis, Nice, France.

K.C. Karki, S.V. Patankar, A.K. Runchal, and H. C. Mongia, 1988. Improved Numerical Methods for Turbulent Viscous Recirculating Flows. Aerothermal Modeling Phase II - Final Report. NASA CR-182169, EDR 13519.

S.V. Patankar, A.K. Runchal, and H. C. Mongia, 1988. Improved Numerical Methods for Turbulent Viscous Recirculating Flows. Aerothermal Modeling Phase II - Final Report. NASA CR-182169, EDR 13519.

A.K. Runchal, 1988. ANSWER: A Computational Fluid Dynamics Code from Mainframes to Microcomputers, ASME Computer Applications Symposium, Loyola Marymont University, Los Angeles, October 21.

V.X. Tung, V.K. Dhir, F. Chang, A.R. Karagozian and F. Zhou, 1988. Enhancement of Single Phase Forced Convection Heat Transfer in Tubes and Ducts Using Staged Tangential Flow Injection. Annual Technical Report # 1 (GRI Contract # 5086-260-1535) from UCLA to Gas Research Institute, Chicago, IL.

L.S. Caretto and A. K. Runchal, 1987. Advanced Swirl Concepts Program: Volume II, Literature Survey and Analytical Modeling. Joint project report by ACRi and Marquardt Company, AFWAL-TR-87-2008, U.S. Air Force Wright Aeronautical Lab., May 1987. Restricted Distribution.

A.K. Runchal, 1987. CONDIF: A Modified Central-Difference Scheme for Convective Flows. Int'l J. Num. Methods in Engg., 24. 1593-1608.

A.K. Runchal,, M.S. Anand and H. C. Mongia, 1987. An Unconditionally-Stable Central Differencing

Scheme for High Reynolds Number Flows. AIAA-87-0060, AIAA 25th Aerospace Sciences Meeting, Jan. 12-15, 1987, Reno, Nevada.

L.S. Caretto and A. K. Runchal, 1986. Ramjet Combustor Modeling for Engineering Design. JANNAF Propulsion Meeting, New Orleans, August 26-28.

A.K. Runchal, 1986. CONDIF: A Modified Central-Difference Scheme with Unconditional Stability and Very Low Numerical Diffusion. Proc. Eighth Intn'l Heat Transfer Conference, August 17-22, 1986, San Francisco, CA.

N.K. Rizk, A.A. Mostafa and H.C. Mongia, 1986. Modeling of Gas Turbine Fuel Nozzles. ASME Winter Annual Meeting, Dec. 7-12; included in Calculations of Turbulent Reactive Flows, AMD-Vol. 81 (1986) Ed. R.M.C. So, J.H. Whitelaw and H.C. Mongia.

B.K. Sultanian and H. C. Mongia, 1986. Fuel Nozzle Air Flow Modeling. AIAA-86-1667, AIAA/ ASME/ SAE/ ASEE 22nd Joint Propulsion Conference, June 16-18, 1986, Huntsville, Alabama.

February 12, 2001

(This page left intentionally blank.)

APPENDIX B

ACRi FREEFORM™ COMMAND LANGUAGE

ACRi FREEFORM™ command language, developed by Analytic & Computational Research, Inc., provides a simple 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. The modules that implement the commands are written in FORTRAN 77. These provide for interactive input or emulate the interactive input in batch mode.

B.1 THE FREEFORM™ INPUT RECORDS

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

B.1.1 The 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.

B.1.2 The 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.

B.1.3 The 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.

B.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.

B.2.1 The Keyword

Function The keyword identifies the input group.

Structure

- The keyword may consist of any characters except separator (Section B.2.4) or terminator (Section B.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, the keyword may be preceded by leading blanks. 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, ABCDxxxxxxx (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 1st 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 B.2.5) or a terminator (Section B.2.6).

B.2.2 The Modifier

Any character information on an input record following a keyword, except that embedded in a numeric or comment field (see Sections B.2.3 and B.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 B.2.1

B.2.3 The 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 B.2.1, B.2.2 and B.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) or upper (E) 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 or e), 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, 99.9E1)
- 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, 1.23E2, 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 B.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 B.2.5).

B.2.4 The 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 B.2.4.1: Valid Separator Characters In FREEFORM™ Language

Number	Character	Description	ASCII Sequence #
1	' '	Space or Blank	32
2	#	Number or pound sign	35
3	(	Left parenthesis	40
4	)	Right parenthesis	41
5	,	Comma	44
6	:	Colon	58
7	;	Semicolon	59
8	=	Equal sign	61
9	^	Caret or Circumflex	94

.

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.

B.2.5 The Terminator

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.

B.2.6 The 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-th character in that record (Section B.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 256th 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.

B.2.7 The Prompt

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 B.2.1 through B.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 7th 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.)

APPENDIX C

EXAMPLES OF ANSWER™ INPUT AND OUTPUT

Illustrative examples of ANSWER™ input command files, and output obtained from them, are supplied under separate cover.