TIDAL

a software tool for dynamic analysis of flow, heat and water quality in large, shallow water bodies

USER'S MANUAL

VERSION 6.0

Rev: 1

©ANALYTIC & COMPUTATIONAL RESEARCH, INC. 1985-2005
ALL RIGHTS RESERVED
NOTIFICATION OF COPYRIGHT

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

September 1, 2005

© ACRi
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.
During the past 25 years, TIDAL™ has evolved from a simple mathematical model for flow and heat transfer into a comprehensive Computational Fluid Dynamics 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.

TIDAL™ 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, TIDAL™ 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 TIDAL™ 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 TIDAL™, 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 TIDAL™ to emerge as a leading software in its field of application.

TIDAL™ 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 TIDAL™ are currently available.

TIDAL™ 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 TIDAL™ 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 TIDAL™ 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

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOTIFICATION OF COPYRIGHT</td>
<td>2</td>
</tr>
<tr>
<td>SOFTWARE LICENSE AGREEMENT</td>
<td>3</td>
</tr>
<tr>
<td>WARRANTY</td>
<td>4</td>
</tr>
<tr>
<td>NOTICE</td>
<td>4</td>
</tr>
<tr>
<td>PREFACE</td>
<td>5</td>
</tr>
<tr>
<td>ACKNOWLEDGEMENTS</td>
<td>6</td>
</tr>
<tr>
<td>NOTATION</td>
<td>14</td>
</tr>
<tr>
<td>CHAPTER 1</td>
<td>18</td>
</tr>
<tr>
<td>OVERVIEW AND INTRODUCTION</td>
<td>18</td>
</tr>
<tr>
<td>1.1 OVERVIEW OF CAPABILITIES</td>
<td>19</td>
</tr>
<tr>
<td>1.2 VALIDATION AND PREVIOUS APPLICATIONS</td>
<td>21</td>
</tr>
<tr>
<td>1.3 STRUCTURE OF TIDAL™</td>
<td>23</td>
</tr>
<tr>
<td>1.4 SCOPE AND LIMITATIONS</td>
<td>24</td>
</tr>
<tr>
<td>1.4.1 System of Equations</td>
<td>24</td>
</tr>
<tr>
<td>1.4.2 Spatial Dimensionality</td>
<td>24</td>
</tr>
<tr>
<td>1.4.3 Problem Geometry</td>
<td>24</td>
</tr>
<tr>
<td>1.4.4 Temporal Dependence</td>
<td>24</td>
</tr>
<tr>
<td>1.4.5 Spatial Dependence</td>
<td>24</td>
</tr>
<tr>
<td>1.4.6 Methods for Solving Governing Equations</td>
<td>24</td>
</tr>
<tr>
<td>1.4.7 Boundary Conditions</td>
<td>25</td>
</tr>
<tr>
<td>1.4.8 Sources and Sinks</td>
<td>25</td>
</tr>
<tr>
<td>1.4.9 Tides</td>
<td>25</td>
</tr>
<tr>
<td>1.4.10 Winds</td>
<td>25</td>
</tr>
<tr>
<td>1.4.11 Flooding and Draining</td>
<td>25</td>
</tr>
<tr>
<td>1.4.12 Chemical Reactions</td>
<td>26</td>
</tr>
<tr>
<td>1.4.13 Operational and Output Control</td>
<td>26</td>
</tr>
<tr>
<td>CHAPTER 2</td>
<td>28</td>
</tr>
<tr>
<td>GOVERNING EQUATIONS</td>
<td>28</td>
</tr>
<tr>
<td>2.1 THE GENERAL TRANSPORT EQUATION</td>
<td>29</td>
</tr>
<tr>
<td>2.2 GOVERNING EQUATION FOR MASS BALANCE (CONTINUITY EQUATION)</td>
<td>30</td>
</tr>
<tr>
<td>2.3 GOVERNING EQUATIONS FOR MOMENTUM</td>
<td>31</td>
</tr>
<tr>
<td>2.4 GOVERNING EQUATIONS FOR HEAT TRANSFER</td>
<td>32</td>
</tr>
<tr>
<td>2.5 GOVERNING EQUATIONS FOR MASS TRANSPORT</td>
<td>33</td>
</tr>
<tr>
<td>CHAPTER 3</td>
<td>35</td>
</tr>
<tr>
<td>AUXILIARY EQUATIONS</td>
<td>35</td>
</tr>
<tr>
<td>3.1 SPECIFICATION OF TIDE</td>
<td>36</td>
</tr>
<tr>
<td>3.2 BOTTOM FRICTION COEFFICIENT</td>
<td>37</td>
</tr>
<tr>
<td>3.3 WIND STRESS COEFFICIENT</td>
<td>38</td>
</tr>
<tr>
<td>Section</td>
<td>Title</td>
</tr>
<tr>
<td>---------</td>
<td>-----------------------------------------------------------------------</td>
</tr>
<tr>
<td>3.4</td>
<td>ATMOSPHERIC PRESSURE</td>
</tr>
<tr>
<td>3.5</td>
<td>CORIOLIS PARAMETER</td>
</tr>
<tr>
<td>3.6</td>
<td>FLUID DENSITY</td>
</tr>
<tr>
<td>3.7</td>
<td>FLUID SPECIFIC HEAT</td>
</tr>
<tr>
<td>3.8</td>
<td>THERMAL DIFFUSIVITY</td>
</tr>
<tr>
<td>3.9</td>
<td>MASS TRANSPORT DIFFUSIVITY</td>
</tr>
<tr>
<td>3.10</td>
<td>FLUID, HEAT OR MASS SOURCES OR SINKS</td>
</tr>
<tr>
<td>3.11</td>
<td>BOUNDARY CONDITIONS</td>
</tr>
<tr>
<td>3.11.1</td>
<td>General Boundary Condition</td>
</tr>
<tr>
<td>3.11.2</td>
<td>Dirichlet Boundary Condition</td>
</tr>
<tr>
<td>3.11.3</td>
<td>Neumann Boundary Condition</td>
</tr>
<tr>
<td>3.11.4</td>
<td>Mixed Boundary Condition</td>
</tr>
<tr>
<td>3.12</td>
<td>INITIAL CONDITIONS</td>
</tr>
<tr>
<td>3.13</td>
<td>FIELD VARIABLE INTERRELATIONSHIPS</td>
</tr>
<tr>
<td>4.1</td>
<td>THE GRID ARRANGEMENT</td>
</tr>
<tr>
<td>4.2</td>
<td>DISCRETIZATION SCHEME</td>
</tr>
<tr>
<td>4.3</td>
<td>SOLUTION METHOD</td>
</tr>
<tr>
<td>4.4</td>
<td>NUMERICAL STABILITY AND ACCURACY</td>
</tr>
<tr>
<td>4.4.1</td>
<td>General Considerations</td>
</tr>
<tr>
<td>4.4.2</td>
<td>Time Scale of Wave Propagation</td>
</tr>
<tr>
<td>4.4.3</td>
<td>Time Scale of Diffusion</td>
</tr>
<tr>
<td>4.4.4</td>
<td>Time Scale of Convection</td>
</tr>
<tr>
<td>4.4.5</td>
<td>Other Pertinent Time Scales</td>
</tr>
<tr>
<td>4.4.6</td>
<td>Grid Peclet Number</td>
</tr>
<tr>
<td>4.5</td>
<td>CONVERGENCE</td>
</tr>
<tr>
<td>5.1</td>
<td>SPECIFICATION OF PROBLEM GEOMETRY</td>
</tr>
<tr>
<td>5.2</td>
<td>SPECIFICATION OF SPATIAL AND TEMPORAL GRID</td>
</tr>
<tr>
<td>5.2.1</td>
<td>Basic Considerations</td>
</tr>
<tr>
<td>5.2.2</td>
<td>Computer Resources</td>
</tr>
<tr>
<td>5.2.3</td>
<td>Numerical Stability and Accuracy</td>
</tr>
<tr>
<td>5.2.4</td>
<td>Scale of Heterogeneity</td>
</tr>
<tr>
<td>5.2.5</td>
<td>Scale of Physical Resolution</td>
</tr>
<tr>
<td>5.2.6</td>
<td>Sources and Sinks</td>
</tr>
<tr>
<td>5.2.7</td>
<td>Boundary Conditions</td>
</tr>
<tr>
<td>5.2.8</td>
<td>Input and Output Requirements</td>
</tr>
<tr>
<td>6.1</td>
<td>OVERVIEW OF KEYWORD COMMAND LANGUAGE</td>
</tr>
<tr>
<td>6.2</td>
<td>ORDER OF INPUT COMMANDS</td>
</tr>
</tbody>
</table>
6.3 SPECIFICATION OF GRID AND PROBLEM DIMENSIONALITY ................................. 75
6.4 INPUT AND OUTPUT FILE UNITS .................................................................... 76
6.5 UNITS OF PHYSICAL QUANTITIES .................................................................... 78
6.6 DIAGNOSTIC OUTPUT SPECIFICATION ....................................................... 79
6.7 DEPENDENT VARIABLES OF TIDAL™ ......................................................... 80
6.8 TABULAR OUTPUT OF FIELD VARIABLES .................................................... 81
6.9 ARCHIVAL AND POST-PROCESSING OUTPUT .............................................. 84
6.10 RESTART OUTPUT .......................................................................................... 85
6.11 MODIFICATION OF INPUT DATA DURING SIMULATIONS ......................... 86
6.12 DIMENSIONING PARAMETER ....................................................................... 88

CHAPTER 7.......................................................................................................................90

DESCRIPTION OF KEYWORD COMMANDS ............................................................... 90

7.1 KEYWORD COMMAND NOTATION ..................................................................... 91
7.2 GENERAL INPUT FEATURES FOR FREEFORM™ COMMANDS ....................... 92
  7.2.1 Interactive or Run Time Input ....................................................................... 92
  7.2.2 File Name Specification .............................................................................. 92
  7.2.3 Subregion Specification ............................................................................ 93
  7.2.4 Subregion Identification .......................................................................... 94
  7.2.5 Boundary Identification .......................................................................... 95
  7.2.6 General Functional Relationships .......................................................... 97
  7.2.7 Command Input of Functional Form ....................................................... 99
  7.2.8 Examples of Functional Form of Input .................................................... 102
  7.2.9 User-Defined Input Options ..................................................................... 105
  7.2.10 Specification of Fluid Phase ................................................................. 105
  7.2.11 Specification of Frequency for Output .................................................... 105

7.3 DESCRIPTION OF KEYWORD COMMANDS ...................................................... 106
  COMMAND ALLOCATE ..................................................................................... 107
  COMMAND BANNER ....................................................................................... 109
  COMMAND BLOCK .......................................................................................... 110
  COMMAND BOUNDARY .................................................................................. 116
  COMMAND CLOSE ........................................................................................... 122
  COMMAND CONDUCTIVITY ............................................................................ 123
  COMMAND CONNECTIVITY ............................................................................. 133
  COMMAND CONVERGENCE ............................................................................ 143
  COMMAND COORDINATE ............................................................................. 148
  COMMAND CORIOLIS ..................................................................................... 155
  COMMAND CORRELATION ............................................................................. 156
  COMMAND CPU ................................................................................................ 159
  COMMAND DEBUG .......................................................................................... 160
  COMMAND DECAY ........................................................................................... 163
  COMMAND DEFINE ........................................................................................ 165
  COMMAND DENSITY ......................................................................................... 166
  COMMAND DIAGNOSTIC ................................................................................. 171
  COMMAND DIFFUSION ..................................................................................... 173
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>DISABLE</td>
<td>174</td>
</tr>
<tr>
<td>ELSE</td>
<td>176</td>
</tr>
<tr>
<td>END</td>
<td>177</td>
</tr>
<tr>
<td>ENDF</td>
<td>178</td>
</tr>
<tr>
<td>FILE</td>
<td>179</td>
</tr>
<tr>
<td>FIX</td>
<td>181</td>
</tr>
<tr>
<td>FLUX</td>
<td>184</td>
</tr>
<tr>
<td>FRICTION</td>
<td>191</td>
</tr>
<tr>
<td>GEOMETRY</td>
<td>193</td>
</tr>
<tr>
<td>GRAVITY</td>
<td>199</td>
</tr>
<tr>
<td>GRID</td>
<td>201</td>
</tr>
<tr>
<td>HISTORY</td>
<td>206</td>
</tr>
<tr>
<td>IF-ELSE-ENDIF</td>
<td>213</td>
</tr>
<tr>
<td>INCLUDE</td>
<td>215</td>
</tr>
<tr>
<td>INDENT</td>
<td>216</td>
</tr>
<tr>
<td>INITIAL</td>
<td>217</td>
</tr>
<tr>
<td>INLET</td>
<td>218</td>
</tr>
<tr>
<td>INTEGRATION</td>
<td>219</td>
</tr>
<tr>
<td>LAND</td>
<td>220</td>
</tr>
<tr>
<td>LIMIT</td>
<td>221</td>
</tr>
<tr>
<td>LOCATE</td>
<td>222</td>
</tr>
<tr>
<td>MATERIAL</td>
<td>246</td>
</tr>
<tr>
<td>MATRIX</td>
<td>255</td>
</tr>
<tr>
<td>OPEN</td>
<td>268</td>
</tr>
<tr>
<td>OPTION</td>
<td>269</td>
</tr>
<tr>
<td>OUTLET</td>
<td>273</td>
</tr>
<tr>
<td>OUTPUT</td>
<td>275</td>
</tr>
<tr>
<td>PAUSE</td>
<td>284</td>
</tr>
<tr>
<td>PRINT</td>
<td>285</td>
</tr>
<tr>
<td>QUIT</td>
<td>295</td>
</tr>
<tr>
<td>READ</td>
<td>296</td>
</tr>
<tr>
<td>REFERENCE</td>
<td>298</td>
</tr>
<tr>
<td>RELAX</td>
<td>303</td>
</tr>
<tr>
<td>RENAME</td>
<td>304</td>
</tr>
<tr>
<td>SAVE</td>
<td>305</td>
</tr>
<tr>
<td>SCALE</td>
<td>315</td>
</tr>
<tr>
<td>SCREEN</td>
<td>316</td>
</tr>
<tr>
<td>SELECT</td>
<td>317</td>
</tr>
<tr>
<td>SET</td>
<td>318</td>
</tr>
<tr>
<td>SOLVE</td>
<td>334</td>
</tr>
<tr>
<td>SOURCE</td>
<td>341</td>
</tr>
<tr>
<td>SPECIFIC</td>
<td>356</td>
</tr>
<tr>
<td>STACK</td>
<td>358</td>
</tr>
<tr>
<td>STATISTICS</td>
<td>361</td>
</tr>
</tbody>
</table>
LIST OF TABLES

TABLE 6.2.1: FUNCTIONAL UNITS OF TIDAL™ COMMANDS............................................. 74
TABLE 6.4.1: I/O FILE UNITS AND THEIR DEFAULT ASSIGNMENT................................. 77
TABLE 6.7.1: DEPENDENT VARIABLES OF TIDAL™ ..................................................... 80
TABLE 6.8.1: STANDARD VARIABLES OF TIDAL™ ....................................................... 82
TABLE 6.8.2: SUPPLEMENTARY VARIABLES OF TIDAL™ ............................................. 83
TABLE 6.11.1: ILLUSTRATION OF A TWO-SEGMENT SIMULATION SEQUENCE............. 87
TABLE 7.2.1: VALID SUBREGION IDENTIFICATION MODIFIERS ..................................... 94
TABLE 7.2.2: BOUNDARY IDENTIFICATION INDEX ....................................................... 95
TABLE 7.2.3: VALID INPUT FUNCTIONAL FORMS.......................................................... 100
TABLE 7.2.4: VALID INDEPENDENT VARIABLES.......................................................... 101
TABLE 7.2.5: NUMERICAL INPUT FOR THE TABLE SPECIFICATION............................. 101
TABLE 7.2.6: NUMERICAL INPUT FOR SIMPLE ANALYTIC FUNCTIONS.......................... 101
TABLE 7.2.7: NUMERICAL INPUT FOR THE SERIES FUNCTIONS.................................. 101
TABLE 7.2.8: NUMERICAL INPUT FOR USER SPECIFIED FUNCTIONS........................... 101
TABLE 7.2.9: EXAMPLES OF SIMPLE ANALYTIC FUNCTIONS..................................... 102
TABLE 7.2.10: EXAMPLES OF ANALYTIC FUNCTION SERIES...................................... 103
TABLE 7.2.11: EXAMPLES OF TABULAR FUNCTIONS.................................................... 104
TABLE 7.2.12: VALID MODIFIERS FOR PHASE DESIGNATION..................................... 105
LIST OF FIGURES

FIGURE 1.1.1: PROCESSES INCORPORATED INTO TIDAL™ ........................................ 20
FIGURE 4.1.1: CARTESIAN GRID ARRANGEMENT ...................................................... 53
FIGURE 4.1.2: CYLINDRICAL GRID ARRANGEMENT .................................................. 53
FIGURE 4.1.3: TYPICAL GRID ELEMENT IN CARTESIAN COORDINATES ................ 54
FIGURE 4.1.4: TYPICAL GRID ELEMENT IN CYLINDRICAL COORDINATES .......... 54
FIGURE 4.1.5: TYPICAL GRID ELEMENT IN X-Y PLANE ......................................... 55
FIGURE 7.2.1: BOUNDARY INDEX NOTATION .......................................................... 96
## NOTATION

<table>
<thead>
<tr>
<th>Symbol</th>
<th>MEANING</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>GENERIC</td>
</tr>
<tr>
<td>c</td>
<td>A constant or exponent</td>
<td>various</td>
</tr>
<tr>
<td>CD</td>
<td>Drag coefficient</td>
<td>-----</td>
</tr>
<tr>
<td>Cf</td>
<td>Bottom friction coefficient</td>
<td></td>
</tr>
<tr>
<td>Cf1</td>
<td>1st component of Cf</td>
<td></td>
</tr>
<tr>
<td>Cf2</td>
<td>2nd component of Cf</td>
<td></td>
</tr>
<tr>
<td>CP</td>
<td>Specific heat</td>
<td>(L^2 t^{-2} T^{-1})</td>
</tr>
<tr>
<td>Cw</td>
<td>Wind stress coefficient</td>
<td></td>
</tr>
<tr>
<td>Cw1</td>
<td>1st component of Cw</td>
<td></td>
</tr>
<tr>
<td>Cw2</td>
<td>2nd component of Cw</td>
<td></td>
</tr>
<tr>
<td>f</td>
<td>A frequency</td>
<td>(t^{-1})</td>
</tr>
<tr>
<td>F</td>
<td>General transport variable</td>
<td>various</td>
</tr>
<tr>
<td>g</td>
<td>Gravitational acceleration</td>
<td>(L t^{-2})</td>
</tr>
<tr>
<td>h</td>
<td>Depth of water below datum</td>
<td>L</td>
</tr>
<tr>
<td>H</td>
<td>Total instantaneous water depth</td>
<td>L</td>
</tr>
<tr>
<td>m</td>
<td>Rate of injection of fluid per unit volume</td>
<td>(M L^{-3} t^{-1})</td>
</tr>
<tr>
<td>m_v</td>
<td>Volumetric mass injection rate</td>
<td>(t^{-1})</td>
</tr>
<tr>
<td>N</td>
<td>Coordinate normal to a boundary</td>
<td>L</td>
</tr>
<tr>
<td>P_a</td>
<td>Atmospheric pressure</td>
<td>(M L^{-1} t^{-2})</td>
</tr>
<tr>
<td>q_m</td>
<td>Rate of injection or withdrawal of fluid</td>
<td>(t^{-1})</td>
</tr>
<tr>
<td>Q</td>
<td>Net flow to from a reservoir</td>
<td>(L^3 t^{-1})</td>
</tr>
<tr>
<td>s</td>
<td>Reaction or decay rate of a property</td>
<td>(M L^{-3} t^{-1})</td>
</tr>
<tr>
<td>S</td>
<td>Source of a fluid property</td>
<td>various</td>
</tr>
<tr>
<td>t</td>
<td>time</td>
<td>t</td>
</tr>
<tr>
<td>T</td>
<td>Thermodynamic Temperature</td>
<td>T</td>
</tr>
<tr>
<td>U</td>
<td>Fluid velocity in x-direction</td>
<td>(L t^{-1})</td>
</tr>
<tr>
<td>V</td>
<td>Fluid velocity in y-direction</td>
<td>(L t^{-1})</td>
</tr>
<tr>
<td>V_i</td>
<td>Fluid velocity in i-direction</td>
<td>(L t^{-1})</td>
</tr>
</tbody>
</table>

... Notation continued
### NOTATION - Continued

<table>
<thead>
<tr>
<th>Symbol</th>
<th>MEANING</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>GENERIC</td>
</tr>
<tr>
<td>vv</td>
<td>Wind Speed</td>
<td>L t⁻¹</td>
</tr>
<tr>
<td>w₀</td>
<td>Reference Wind speed</td>
<td>L t⁻¹</td>
</tr>
<tr>
<td>W</td>
<td>Fluid velocity in z-direction</td>
<td>L t⁻¹</td>
</tr>
<tr>
<td>x</td>
<td>x-coordinate</td>
<td>L</td>
</tr>
<tr>
<td>y</td>
<td>y-coordinate</td>
<td>L</td>
</tr>
<tr>
<td>z</td>
<td>z-coordinate</td>
<td>L</td>
</tr>
</tbody>
</table>

### GREEK SYMBOLS

<table>
<thead>
<tr>
<th>Symbol</th>
<th>MEANING</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>GENERIC</td>
</tr>
<tr>
<td>α</td>
<td>A coefficient</td>
<td>various</td>
</tr>
<tr>
<td>β</td>
<td>A coefficient</td>
<td>various</td>
</tr>
<tr>
<td>ε</td>
<td>A small quantity</td>
<td>-----</td>
</tr>
<tr>
<td>Φ</td>
<td>A field variable or source of boundary</td>
<td>various</td>
</tr>
<tr>
<td>γ</td>
<td>Net decay rate of species</td>
<td>M L⁻³ t⁻¹</td>
</tr>
<tr>
<td>Γₑ</td>
<td>Effective Diffusion coefficient</td>
<td>ML⁻¹ t⁻¹</td>
</tr>
<tr>
<td>Γ_F</td>
<td>Diffusivity coefficient</td>
<td>L² t⁻³</td>
</tr>
<tr>
<td>η</td>
<td>Water surface elevation above datum</td>
<td>L</td>
</tr>
<tr>
<td>λ</td>
<td>Decay rate constant</td>
<td>t⁻¹</td>
</tr>
<tr>
<td>Θ</td>
<td>Angle of wind vector with x-axis</td>
<td>degrees</td>
</tr>
<tr>
<td>ρ</td>
<td>Fluid density</td>
<td>M L⁻³</td>
</tr>
<tr>
<td>σ</td>
<td>Generation fraction for species</td>
<td>-----</td>
</tr>
<tr>
<td>Ω</td>
<td>Coriolis Parameter</td>
<td>-----</td>
</tr>
</tbody>
</table>
### SUBSCRIPTS

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>PERTAINING TO</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>Mass species</td>
</tr>
<tr>
<td>F</td>
<td>General fluid transport property F</td>
</tr>
<tr>
<td>i</td>
<td>The $i^{th}$ coordinate direction or $i^{th}$ phase</td>
</tr>
<tr>
<td>inj</td>
<td>Injected fluid</td>
</tr>
<tr>
<td>T</td>
<td>Temperature</td>
</tr>
<tr>
<td>t</td>
<td>The time coordinate</td>
</tr>
<tr>
<td>x</td>
<td>The $x$ direction</td>
</tr>
<tr>
<td>y</td>
<td>The $y$ or $r$ direction</td>
</tr>
<tr>
<td>z</td>
<td>The $z$ or $\theta$ direction</td>
</tr>
</tbody>
</table>

### SUPERSCRIPTS

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>PERTAINING TO</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td>Pertaining to property $F$</td>
</tr>
<tr>
<td>k</td>
<td>Pertaining to the $k^{th}$ chemical species</td>
</tr>
<tr>
<td>n</td>
<td>Pertaining to the $n^{th}$ chemical species</td>
</tr>
</tbody>
</table>
CHAPTER 1

OVERVIEW AND INTRODUCTION

The TIDAL™ is a general-purpose software tool for solution of the fluid flow, heat and mass transfer problems in shallow water bodies. 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™ and acrPLOT™ 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 6.00, 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 TIDAL™.
1.1 OVERVIEW OF CAPABILITIES

The TIDAL™ software package is a comprehensive mathematical model for simulation of fluid flow, heat transfer, and mass transport processes in shallow water bodies. 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 TIDAL™ easily without extensive training.

TIDAL™ can be used to simulate transient or steady state problems in a water body with irregular coastline, complex bathymetry, and islands. The water body may contain rivers, sources, inlets and outlets. It may have coastal plains or tidal flats which get inundated with or drained of water from time to time. TIDAL™ 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 occur between the different components of the flow system. In the TIDAL™ software package, these components may be employed in either a coupled or uncoupled mode. Some of the important features are:

- Transient or steady state simulation
- Complex bathymetry and irregular coastline
- Flooding and draining of coastal areas
- Multiple water quality parameters
- Inherently mass-conservative numerical method
- Arbitrarily non-uniform grid
- Time-dependent options for physical input
- Versatile boundary condition options
- Arbitrary sources and sinks
- 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
- Option to solve any or all of the governing equations
- Dynamic coupling between flow, heat and mass transport
FIGURE 1.1.1: PROCESSES INCORPORATED INTO TIDAL™
1.2 VALIDATION AND PREVIOUS APPLICATIONS

TIDAL™ 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 TIDAL™ is in integrating these disparate elements into a software package that is general, flexible, economical, and easy to use. TIDAL™ 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 TIDAL™ are currently available. A partial list of these publications is given in Appendix A. Specific applications of the software have included:

- Water quality impact after setting new buildings in the North Lake of Tunis. Calculation of currents induced by gates opening with respect to tide level, Tunisia
- Technical study of the Private Marina of King of Jordan. Flushing process of the Marina, Kingdom of Jordan
- ISOLE Research Contract for European Community. Contribution to a Prototype of numerical Tool supporting the management of tourism impact of islands, including water quality and risks of coastal pollution by oils spills. Islands selected : Mykonos Island (Greece) and Elba Island (Italy), European Community
- Definition of an outfall in Seine River associated to UVE Plant of Rouen, Normandy, France
- Calculation of currents induced by outfalls in the lake of Saclay, Paris, France
- Alpha project Beirut - Numerical modelling of the flushing process of the inner basin –1998, Lebanon
- Study of the geometry of the Darse of Vigneux Harbor to avoid sedimentation. Calculations of current circulation, Paris, France
- Improvement of water quality around Dieppe Harbor. Dispersion of Bacteria close to the beaches and evaluation of water quality parameters, France
- Currents induced by a seiche in basins of “Port Tudy”, Brittany, France
- Discharge of Pollutants and Water Quality Study, Laita Estuary, Brittany, France
- Hydrodynamics Resulting from a Water Treatment Plant, Acheres, Paris, France
- Coastal Management, Proposed Marina at Port Hammamet, Morocco
- Environmental Impact of Effluent on BOD/DO and Water Quality, Ría de Avilés, Spain
- Tidal Circulation and Pollutant Transport, Bombay Harbour, India
- OTEC Discharge Modeling, Kahe Offshore, Oahu, Hawaii
- Modeling of Urban Effluent Outfall, Cote de Nacre, France
- Analysis of Impact of Cold Water Discharge, Rio Odiel, Spain
- Impact of Offshore Construction, Prudhoe Bay & Simpson Lagoon, Alaska
- Circulation and Oil Spill Analysis, Beaufort Sea, Alaska
- Nuclear Power Plant Discharge, Halileh, Persian Gulf, Iran
- Urban Water Resource Study, Kaneohe Bay, Oahu, Hawaii
- Sewage and Urban Effluent Dispersion, St. Cyprien, France
- Water Treatment Plant Discharge, Lake Manzala, Egypt
- Offshore Oil Platform Water Level and Circulation Study, Bombay High, India
- Thermal Power Plant Discharge, St. John’s River, Florida
- Industrial Effluent Discharge, Coquimbo Bay, Chile
- Storm Surge Analysis, Lake Michigan, Michigan
- Hurricane Water Level Analysis, St. Rosalie, Louisiana
- Storm Surge Analysis, Long Island Sound, New York
- Maximum Water Level Study, Persian Gulf, Iran.
1.3 STRUCTURE OF TIDAL™

TIDAL™ 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 objective, a simple and convenient method is employed to define the geometry and physics of the system. The domain of interest is considered to be composed of water or land zones. By default the whole of the region of interest is considered to be a contiguous water zone. The land zones may be specified by simple user commands. A land zone may consist of a single 'element' of the grid; it may consist of a set of contiguous elements or it may be spread over scattered elements of the grid. The field variables, such as velocity, water heights, 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 1000 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 TIDAL™. 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 TIDAL™ 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

TIDAL™ numerically solves a variable set of equations for fluid velocity, instantaneous water heights, temperature and one or more chemical species. These equations are supplemented by constitutive equations, equations of state, and initial and boundary conditions. The equations are coupled through convection and source effects. These equations may be solved individually or simultaneously in a coupled or uncoupled manner, depending on the needs of a specific problem and the options selected by the user.

The current version of TIDAL™ 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) depth-averaged shallow water equations. However, it provides for elements that allow solution of three-dimensional (3D) temperature and chemical species equations. However in such a case, the user must provide the 3 dimensional velocity field or a methodology to generate the 3D velocity field from the 2D depth-averaged values.

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 is defined in terms of cartesian (x,y,z) coordinates. The horizontal plane is represented by x-y in the cartesian system. The z direction is always the direction of the vertical coordinate. The bathymetry and coastline of the water body may be complex and irregular and it may include any number of islands.

1.4.4 Temporal Dependence

By default, TIDAL™ operates in time-dependent mode. However, steady-state problems can also be solved. Except for the spatial grid, all problem parameters can change with time. The values of some parameters, such as boundary conditions and source terms for fluid, heat, and mass, can change arbitrarily. Such quantities can be specified as analytic functions, in the form of tables, or in a step-wise manner at specified time intervals.

1.4.5 Spatial Dependence

The initial and boundary conditions, bathymetry, tides, wind field, atmospheric pressure, bottom friction and, values of sources and sinks, are allowed to vary over the spatial grid.

1.4.6 Methods for Solving Governing Equations

For solution of the flow equations, a modified form of the coupled ADI method, originally devised by Leendertse (1970), is used. For water quality parameters 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 may be approximated by Picard, a modified Picard or a Newton-Raphson method. The flux terms may be approximated by the dependent variable or its change from the current state. The elements used to define
the problem geometry can vary in size, but their shape is restricted to that of a rectangular parallelepiped. Three distinct discretization schemes are available: the hybrid (Runchal, 1972), exponential (Spalding, 1972) and CONDIF (Runchal, 1987b) schemes.

The resulting matrix of algebraic equations can be solved by one or more of several matrix inversion algorithms. The available options include the explicit method of Point Successive Over-Relaxation, the semi-implicit method of the Alternating Direction Implicit (ADI), and the implicit methods of 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 TIDAL™ 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 Tides

Tides may be specified at any of the four vertical boundaries of the water body. The tidal variation may be determined in one of three ways: (1) It may be composed of one or more sinusoidal components; (2) It may be in a tabular form; or (3) It may be a function of water inflow or outflow from a connecting reservoir. Each of the boundaries may have an independent tidal specification and the tide may have a phase shift (lag or lead) or it may vary linearly along a boundary.

1.4.10 Winds

Wind speed and direction may be specified as functions of time, space or other variables at one or more locations on the surface of the water body. The specification in time may be as an analytic function or in tabular form. If the spatial variation of wind is specified at a discrete number of locations, then the values at other locations may be computed by an inverse or inverse-squared power law.

1.4.11 Flooding and Draining

TIDAL™ accounts for inundation and drainage of coastal areas and tidal flats. The water elevation at each location in the field of interest is monitored at every time step. If the elevation exceeds a pre-specified minimum then that location is allowed to dynamically interact with the rest of the water body. Conversely if the water elevation falls below the minimum then that location is made inactive.
1.4.12 Chemical Reactions

TIDAL™ incorporates a multi-step chemical kinetic algorithm for chemical reactions.

1.4.13 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

_TIDAL™_ solves a set of coupled transport equations for fluid velocities, instantaneous water depth, temperature, and concentration of chemical species in a large shallow water body. The number of species equations to be solved can easily be varied to accommodate specific user requirements. It is able to accommodate chemical reactions, 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 Shallow Water Approximation to the 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:

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

where

- $t$ is the time coordinate,
- $\alpha$ is a coefficient for the accumulative term,
- $\rho$ is the mass density of the fluid,
- $F$ is the transported property,
- $\beta$ 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,
- $\Gamma_e^F$ 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:

$$\frac{\partial}{\partial t} (\alpha F) + \frac{\partial}{\partial x} (\beta U F) + \frac{\partial}{\partial r} (\beta r V F) + \frac{\partial}{\partial z} (\beta W F) = \frac{\partial}{\partial x} (\Gamma_e^F \frac{\partial F}{\partial x}) + \frac{1}{r} \frac{\partial}{\partial r} (\Gamma_e^F \frac{\partial F}{\partial r}) + \frac{1}{r} \frac{\partial}{\partial z} (\Gamma_e^F \frac{\partial F}{\partial z}) + m_{inj} F_{inj} + S_F \cdot s_F \cdot \alpha F. \tag{2.1.2}$$

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 ($\theta$).
2.2 GOVERNING EQUATION FOR MASS BALANCE (CONTINUITY EQUATION)

The continuity equation for a shallow water system is written as (Stoker, 1957):

\[
\frac{\partial \eta}{\partial t} + \frac{\partial}{\partial x} (H U) + \frac{\partial}{\partial y} (H V) = H m_v ,
\]  
(2.2.1)

with

\[
\eta = H - h ,
\]  
(2.2.2)

where:

- \(t\) is the time coordinate,
- \(\eta\) is the water surface elevation measured positive upwards from the reference datum plane,
- \(x, y\) are a set of horizontal, mutually orthogonal, cartesian coordinates,
- \(H\) is the total instantaneous water depth,
- \(U, V\) are the horizontal velocity components in the x and y direction, respectively,
- \(m_v\) is the vertically-integrated average rate of mass injection (> 0) or withdrawal (< 0) of fluid per unit volume divided by the fluid density, and
- \(h\) is the depth of water measured positive downwards from the reference datum plane.
2.3 GOVERNING EQUATIONS FOR MOMENTUM

The vertically integrated momentum balance equations for a shallow water body are given by (Stoker, 1957):

\[
\frac{\partial U}{\partial t} + U \frac{\partial U}{\partial x} + V \frac{\partial U}{\partial y} = -g \frac{\partial \eta}{\partial x} - \frac{1}{\rho} \frac{\partial P_a}{\partial x} + \Omega V + \frac{C_w w^2}{H} \cos \Theta - \frac{C_f |V|}{H} U + S_U,
\]

\[
\frac{\partial V}{\partial t} + U \frac{\partial V}{\partial x} + V \frac{\partial V}{\partial y} = -g \frac{\partial \eta}{\partial y} - \frac{1}{\rho} \frac{\partial P_a}{\partial y} - \Omega U + \frac{C_w w^2}{H} \sin \Theta - \frac{C_f |V|}{H} V + S_V,
\]

\[
|V| = \sqrt{U^2 + V^2}
\]

where:

- \(g\) is the constant of gravitational acceleration,
- \(\rho\) is the fluid mass density,
- \(P_a\) is the atmospheric pressure,
- \(\Omega\) is the coriolis parameter,
- \(C_w\) is a wind stress coefficient,
- \(w\) is the wind speed,
- \(\Theta\) is the wind angle measured from the positive direction of \(x\) towards the positive direction of \(y\) axis,
- \(C_f\) is the bottom friction coefficient,
- \(S_U\) is the average rate of \(x\)-momentum generated (> 0) or dissipated (< 0) per unit volume divided by the mass density of water, and
- \(S_V\) is the average rate of \(y\)-momentum generated (> 0) or dissipated (< 0) per unit volume divided by the mass density of water.
2.4 GOVERNING EQUATIONS FOR HEAT TRANSFER

The governing equation for heat transfer is based on the equation for the conservation of thermal energy:

\[
\frac{\partial}{\partial t} (\alpha T) + \frac{\partial}{\partial x_i} (\alpha V_i T) = \frac{\partial}{\partial x_i} (\alpha \Gamma_T \frac{\partial T}{\partial x_i}) + \alpha S_T ,
\]

(2.4.1)

where:
- \(\alpha\) is the water depth, \(H\), for 2D vertically integrated simulations and is unity for 3D simulations,
- \(T\) is the water temperature,
- \(x_i\) is the cartesian coordinate in the \(i^{th}\) direction,
- \(V_i\) is the velocity component in the \(i^{th}\) direction,
- \(\Gamma_T\) is the effective thermal diffusivity coefficient, and
- \(S_T\) is the average rate of heat generated (> 0) or dissipated (< 0) per unit volume divided by the mass density and specific heat of water.
2.5 GOVERNING EQUATIONS FOR MASS TRANSPORT

The equation for conservation of mass species, in a manner analogous to that for heat transfer, is written as:

\[
\frac{\partial}{\partial t} (\alpha C^k) + \frac{\partial}{\partial x_i} (\alpha V_i C^k) = \frac{\partial}{\partial x_i} (\alpha \Gamma_C \frac{\partial C^k}{\partial x_i}) + \alpha S^k - \alpha \gamma^k,
\]

(2.5.1)

where:
- \(C^k\) is the mass concentration of the \(k\)-th species in fluid,
- \(\Gamma_C\) is the effective mass diffusivity coefficient,
- \(S^k\) is the source of the \(k\)-th species,
- \(\gamma^k\) is the net decay of the \(k\)th species.

For a species with both decay and regeneration, the net decay is given by:

\[
\gamma^k = \lambda^k C^k - m \sum_m \sigma^m \lambda^m C^m,
\]

(2.5.2)

where \(\lambda^k\) is the decay rate for the \(k\)th species, and \(m \sigma^m\) is the fraction of decay of the \(m\)th species which generates the \(k\)th species.
Several auxiliary relations are required to solve the governing transport equations. Solution of these equations requires the specification of pressure, wind and bottom stresses, fluid properties, constitutive relations, source terms, and initial and boundary conditions. The mathematical framework of TIDAL™ 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 SPECIFICATION OF TIDE

The most general method for specifying the tide at a boundary is by using the BOUNDARY or SET commands with ETA modifier; these will be described later with the description of the commands. In addition, five options are provided for most commonly encountered types of tides. In the 1st option, the tide at an open boundary is specified as a tabulated function of time; the instantaneous values are then computed by linear interpolation from the tabulated values. In the 2nd option, the tide is specified as a tabulated function of time at two locations on the tidal boundary. The value at any location on the boundary, at any time, is then obtained by linear interpolation in both temporal and spatial domain.

The 3rd and 4th options specify the tide as harmonic components; these are:

\[
\eta = \sum_{n=1}^{N} a_n \cos \left( \frac{2\pi}{T_n} t + \phi_n \right),
\]

and

\[
\eta = \sum_{n=1}^{N} a_n \sin \left( \frac{2\pi}{T_n} t + \phi_n \right),
\]

where \(\eta\) is instantaneous tide height, \(N\) is the number of tidal harmonic components and, \(a_n\), \(T_n\), and \(\phi_n\) are, respectively, the amplitude, period and phase lag of the \(n\)th tidal component.

The 5th option allows a tidal boundary to be determined by an "imaginary" reservoir connected to it. In this case the governing equation for tidal elevation at the boundary is:

\[
\frac{\partial \eta}{\partial t} = \frac{Q}{Q_R},
\]

where \(Q\) is the net flow to or from the reservoir, and \(Q_R\) is a constant factor called the reservoir capacity factor. If there is a net outflow of water from the solution domain to the reservoir then the sign is positive; otherwise it is negative.
3.2 BOTTOM FRICTION COEFFICIENT

Three options are provided for calculation of the bottom friction coefficient, $C_f$. In the first option, the bottom friction coefficient is directly specified as a constant and uniform value throughout the water body. In the second option, the friction coefficient is calculated according to the Manning's formula:

$$C_f = g \left( \frac{C_{f1}}{C_{f2}} \right)^2 \frac{1}{H^{1/3}} \quad (3.2.1)$$

where $C_{f1}$ and $C_{f2}$ are empirical constants.

In the third option, the friction coefficient is calculated according to the Chezy's formula:

$$C_f = g \frac{1}{C_{f1}^2} \left[ C_{f2} + \frac{1}{\sqrt{H}} \right]^2, \quad (3.1.2)$$

where $C_{f1}$ and $C_{f2}$ are empirical constants.
3.3 WIND STRESS COEFFICIENT

The wind stress coefficient, $C_w$, is calculated from the non-linear function of the wind speed as:

$$C_w = C_{w1} + C_{w2} \max \left\{ 0 , \left( 1 - \frac{w_0}{w} \right)^2 \right\},$$

(3.3.1)

where $C_{w1}$ and $C_{w2}$ and $w_0$ are empirical constants and $w$ is the wind speed.
3.4 ATMOSPHERIC PRESSURE

Atmospheric pressure variation, $P_a$, is assumed to be given by the user. Any arbitrary variation can be specified by the user through the available input commands.
3.5 CORIOLIS PARAMETER

The coriolis parameter is twice the vertical component of earth’s angular velocity (ω) about the vertical axis. For a water body located at a latitude of φ; the Coriolis parameter, Ω, is defined by:

\[ \Omega = 2 \omega \sin \varphi \]  \hspace{1cm} (3.5.1)

This parameter is considered to be constant for the entire water body. For rotation on earth \( \omega = 7.292115 \times 10^{-5} \) radians/second.
3.6 FLUID DENSITY

The fluid density, ρ, is assumed to be uniform.
3.7 FLUID SPECIFIC HEAT

Specific heat of the fluid is assumed to be constant and uniform.
3.8 THERMAL DIFFUSIVITY

Thermal required for the solution of the temperature equations is assumed to be uniform. The thermal diffusivity is assumed to be the effective thermal conductivity divided by the specific heat and density of the fluid. Since the equations assume a total effective value for the diffusivity, the specified value must include both molecular diffusivity and any turbulent contribution.
3.9 MASS TRANSPORT DIFFUSIVITY

Mass diffusivity required for the solution of the species equations is assumed to be uniform. Since the equations assume a total effective value for the diffusivity, the specified value must include both molecular diffusivity and any turbulent contribution.
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:

\[
S_F = S_F(\xi),
\]

where \( S_F \) represents any of the flow, heat or mass sources or sinks, and \( \xi \) 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:

\[
S_F = q_m F_{inj},
\]

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:

\[
S_C^k = f_S(C_S^k - C^k),
\]

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:

\[
\frac{\partial C}{\partial t} = -\lambda_k C,
\]

where \( \lambda_k \) is the reaction-rate constant. With \( t_{1/2} \) as the half-life of the \( k \)th species, the \( \lambda_k \) is given by:

\[
\lambda_k = \frac{\ln(2)}{t_{1/2}^k}
\]

A fraction of the decay of the \( k \)th species may generate the next species in the chain; this fraction is denoted by \( \sigma^m \) and, along with \( t_{1/2} \), 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:

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

where \(F\) represents any of the dependent variables, \(N\) is the direction coordinate normal to the boundary, and \(\Gamma, h_F, F_o,\) 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:

\[ F = F_o \]  (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:

\[ - \Gamma \frac{\partial F}{\partial N} = q_F \]  (3.11.3)

Here, \(\Gamma\) is unity if the gradient of the variable is specified or it represents the appropriate component of the effective diffusion and dispersion tensor (\(\Gamma_e\) 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:

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

In this equation, \(\Gamma\) 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.
3.12 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:

\[ F_P = F_0 + \sum_{n=1}^{N} a_n F_n, \]  

(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, \( F_0 \) 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.
(This page left intentionally blank.)
CHAPTER 4
NUMERICAL BASIS

The continuity and momentum equations of TIDAL™ are solved by a coupled ADI method. Other 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 $U$, $\delta L$ and $\Gamma$, respectively, as the velocity component, grid interval and diffusivity in a given direction, the grid Peclet number, $Pe$, is defined as:

$$Pe = \frac{\Gamma}{U \delta L}$$

(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:
\[ A_P (F_P^{k+1} - F_P^k) + \sum_K A_P \left[ \Theta (F_P^{k+1} - F_K^{k+1}) + (1 - \Theta) (F_P^{k+1} - F_K^k) \right] \]
\[ = \delta V s_{FP} - s_{FP} \left[ \Theta F_P^{k+1} + (1 - \Theta) F_P^k \right] \]

In Equation 4.2.2, the subscript P denotes the node point at which the discrete equation is derived, \( \delta V \) 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 \( \Theta \) determines whether the numerical scheme is explicit (\( \Theta = 0 \)), implicit (\( \Theta = 1 \)), or semi-implicit (\( 0 < \Theta < 1 \)). The choice of \( \Theta \) 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 ($F_{i+1}$) 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 solution methods, numerical instability may occur when the physical process being simulated exhibits non-linearity or when strong coupling exists between various components of the model. Numerical instability may exhibit itself either as "weak instability" or "exponential growth". Weak instability usually can be identified as a step-to-step oscillation about a mean value. This type of instability becomes obvious by examining the diagnostic output. Exponential growth can be identified by an uncontrolled growth in the values of state variables.

The hybrid, exponential 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 TIDAL 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 Wave Propagation

The characteristic time scale for propagation of a wave effect is given by:

\[ \delta t_C = \delta L / c, \]

where \( L \) is a representative grid interval and \( c \) is the celerity of the wave. In shallow water, the wave celerity is essentially equal to \( gH \).

This time scale is important for transient flow problems; all long wave disturbances propagate across a grid interval of length \( L \) in this time. 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.

### 4.4.3 Time Scale of Diffusion

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

\[ \delta t_{\text{diff}} = \alpha_e \delta L^2 / (2 \Gamma_e) \]

where \( \alpha_e \) is a representative value of the storage coefficient, and \( \Gamma_e \) is a representative value of the effective diffusivity.

The general implications of this time scale are similar to those of the wave 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 \( L \) as the characteristic velocity component and grid interval, respectively, the convection time scale is defined as:

\[ \delta t_{\text{conv}} = \delta L / U \]

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

\[ Co = \frac{\delta t}{\delta t_{\text{conv}}} = \frac{U \delta t}{\delta L} \]

where \( \delta t \) 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, \( t_{\text{conv}} \).

### 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 (β 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 TIDAL™ 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 form are:

\[ A_{ij} \phi_i = b_j \]  \hspace{1cm} (4.5.1)

where \( A_{ij} \) is the matrix of coefficients, \( \phi_i \) 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:

\[ \phi_i^k = T_{ij} \phi_j^{k-1} + c_i \]  \hspace{1cm} (4.5.2)

where \( \phi_i^k \) is the \( k \)th approximation to \( \phi_i \), \( T_{ij} \) 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:

\[ R_1 = \left| \sum_{j=1}^{m} A_{ij} \phi_j - b_i \right| \]  \hspace{1cm} (4.5.3)

\[ R_2 = \left| 1 - \frac{\phi_i^k}{\phi_i^{k-1}} \right| ; \quad \phi_i^{k-1} > \phi_{\text{min}} \]  \hspace{1cm} (4.5.4)

\[ R_3 = \left| \phi_i^k - \phi_i^{k-1} \right| \]  \hspace{1cm} (4.5.5)

where \( \phi_{\text{min}} \) 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:

\[ R_{\text{max}} = \frac{1}{N} \sum_{n=1}^{N} (R_n) \leq \varepsilon, \]  \hspace{1cm} (4.5.6)

\[ R_{\text{max}} = \max \left( R_n \right) \leq \varepsilon, \quad i = 1, 2, 3, \ldots, N, \]  \hspace{1cm} (4.5.7)

where \( \varepsilon \) 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 basic geometrical elements or zones, called “sub-domains”. These divide the domain of interest into a number of subregions each representing either water or land. 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) coordinates. For the vertically integrated system, the z coordinate is replaced by the bathymetry of the water body in terms of the depth below datum (h) and surface elevation from datum (η). The domain of interest is specified in terms of basic geometric elements called "zones" or "sub-domains". A zone may represent either water or land. Those zones of the water body where the depth of water is larger than a pre-specified minimum (10^-3) are considered to be water zones; all other zones are considered to be land. The user may override this specification by explicit commands. TIDAL™ has two modes of operation. In the first, these "land" and "water" zones stay that way throughout the simulation. In the second mode, run-up and draw-down may occur and the land and water zones are constantly monitored and updated based upon the instantaneous water depth.

Once the physical domain of the problem has been identified in terms of land and water, the next step is to impose a discrete spatial grid over the domain of interest (See Section 4.1). Criteria and suggestions for the selection of this grid are discussed in Section 5.2. Each grid node is associated with an element or control volume that is formed by the mid-points of the grid lines (Figure 4.1.2). The location of each grid node is identified by its physical coordinates (x,y,z) and by a unique set of grid indices (I,J,K) where I, J and K refer, respectively, to the x, y and z directions. All the state variables except velocity components are defined at the grid nodes. The U, V and W velocity components, as shown in Figure 4.1.2, are defined at locations which are, respectively, midway between the grid nodes in the x, y and z directions. The index notation that is employed is such that the velocity components at the west, south and bottom face of the control volume are denoted by the same I, J and K indices as the F values at the grid node.

The grid must be selected in a manner that makes the physical boundaries coincident with the control-volume boundaries. An exception to this rule is allowed for the outermost zones at the boundaries of the domain of interest. These boundaries may coincide with either the control-volume boundaries or the grid node locations. This arrangement allows the fluxes across the zone boundaries to be correctly incorporated into the TIDAL™ algorithm. The fluxes at the outermost boundaries are computed from the specified boundary conditions. The Dirichlet boundary conditions may be specified at the grid nodes or element interfaces; all other types of boundary conditions are always specified at the element interface. If the outer boundary coincides with the element interface, then the outermost node lies outside the domain of interest and acts as a fictitious node for implementing 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 ($\delta x$, $\delta y$ and $\delta z$, or $\delta \theta$) and temporal ($\delta t$) 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

TIDAL™ 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 TIDAL™.
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 TIDAL™ AND THEIR FUNCTIONS

<table>
<thead>
<tr>
<th>NO.</th>
<th>KEYWORD</th>
<th>INPUT FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ALLOCATE</td>
<td>Allocate table space for storage of user input</td>
</tr>
<tr>
<td>2</td>
<td>BANNER</td>
<td>Print user &amp; program identification to output file</td>
</tr>
<tr>
<td>3</td>
<td>BLOCKAGE</td>
<td>Specify blockage or solid objects in flow field</td>
</tr>
<tr>
<td>4</td>
<td>BOUNDARY</td>
<td>Override built-in boundary conditions</td>
</tr>
<tr>
<td>5</td>
<td>CLOSE</td>
<td>Close specified output device immediately</td>
</tr>
<tr>
<td>6</td>
<td>CONDUCTIVITY</td>
<td>Specify conduction or diffusion constants, coefficients and options</td>
</tr>
<tr>
<td>7</td>
<td>CONNECTIVITY</td>
<td>To specify element and vertex connectivity for unstructured grid</td>
</tr>
<tr>
<td>8</td>
<td>CONVERGENCE</td>
<td>Specify convergence criterion</td>
</tr>
<tr>
<td>9</td>
<td>COORDINATE</td>
<td>Specify grid coordinates</td>
</tr>
<tr>
<td>10</td>
<td>CORIOLIS</td>
<td>Specify Coriolis force constant</td>
</tr>
<tr>
<td>11</td>
<td>CORRELATION</td>
<td>Computation of correlations between variables and locations</td>
</tr>
<tr>
<td>12</td>
<td>CPU</td>
<td>Specify number of Central Processing Units for parallel processing</td>
</tr>
<tr>
<td>13</td>
<td>DEBUG</td>
<td>Specify debug options</td>
</tr>
<tr>
<td>14</td>
<td>DECAY</td>
<td>Specify decay constants for chemical species</td>
</tr>
<tr>
<td>15</td>
<td>DEFINE</td>
<td>Specify value of a symbolic variable</td>
</tr>
<tr>
<td>16</td>
<td>DENSITY</td>
<td>Select fluid mass density options</td>
</tr>
<tr>
<td>17</td>
<td>DIAGNOSTIC</td>
<td>Diagnostic output options</td>
</tr>
<tr>
<td>18</td>
<td>DIFFUSION</td>
<td>Specify conduction or diffusion constants, coefficients and options</td>
</tr>
<tr>
<td>19</td>
<td>DISABLE</td>
<td>Disable certain built-in default options</td>
</tr>
<tr>
<td>20</td>
<td>ELSE</td>
<td>Define the “ELSE” part of the IF-ELSE-ENDIF construct</td>
</tr>
<tr>
<td>21</td>
<td>END</td>
<td>End of a problem</td>
</tr>
<tr>
<td>22</td>
<td>ENDIF</td>
<td>Terminate the IF-ELSE-ENDIF construct</td>
</tr>
<tr>
<td>23</td>
<td>FILE</td>
<td>Open or close selected I/O device</td>
</tr>
<tr>
<td>24</td>
<td>FIX</td>
<td>Fixed pressure, temperature or concentration nodes</td>
</tr>
<tr>
<td>25</td>
<td>FLUX</td>
<td>Compute flux of fluid, heat or chemical species crossing a sub-domain</td>
</tr>
<tr>
<td>26</td>
<td>FRICTION</td>
<td>Specify bottom and wind stress friction coefficients</td>
</tr>
<tr>
<td>27</td>
<td>GEOMETRY</td>
<td>Specify or modify problem geometry</td>
</tr>
<tr>
<td>28</td>
<td>GRAVITY</td>
<td>Specify constants of gravitational acceleration</td>
</tr>
<tr>
<td>29</td>
<td>GRID</td>
<td>Number of grid nodes in the x, y and z directions</td>
</tr>
<tr>
<td>30</td>
<td>HISTORY</td>
<td>Provide time history output at selected nodes</td>
</tr>
<tr>
<td>31</td>
<td>IF</td>
<td>To initiate the IF-ELSE-ENDIF construct</td>
</tr>
<tr>
<td>32</td>
<td>INCLUDE</td>
<td>To include a file at run time in the input data stream</td>
</tr>
</tbody>
</table>
### TABLE 6.1.1: KEYWORDS OF TIDAL™ AND THEIR FUNCTIONS (continued)

<table>
<thead>
<tr>
<th>NO.</th>
<th>KEYWORD</th>
<th>INPUT FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>33</td>
<td>INDE</td>
<td>To allow command input to be indented</td>
</tr>
<tr>
<td>34</td>
<td>INITIAL</td>
<td>Initial conditions for state variables</td>
</tr>
<tr>
<td>35</td>
<td>INLET</td>
<td>To specify an inflow boundary for the domain of computation</td>
</tr>
<tr>
<td>36</td>
<td>INTEGRATION</td>
<td>Index for selection of integration profile</td>
</tr>
<tr>
<td>37</td>
<td>LAND</td>
<td>Define Land Boundary</td>
</tr>
<tr>
<td>38</td>
<td>LIMIT</td>
<td>Set upper and lower limits for dependent variables</td>
</tr>
<tr>
<td>39</td>
<td>LOCATE</td>
<td>Specify location of sub-regions or boundaries</td>
</tr>
<tr>
<td>40</td>
<td>MATERIAL</td>
<td>Specify material types and properties</td>
</tr>
<tr>
<td>41</td>
<td>MATRIX</td>
<td>Specify Options for solution of matrix of equations</td>
</tr>
<tr>
<td>42</td>
<td>OPEN</td>
<td>Specify an open boundary through which fluid may enter or leave</td>
</tr>
<tr>
<td>43</td>
<td>OPTION</td>
<td>Modify built-in default options</td>
</tr>
<tr>
<td>44</td>
<td>OUTLET</td>
<td>Specify an outflow boundary for the domain of computation</td>
</tr>
<tr>
<td>45</td>
<td>OUTPUT</td>
<td>Frequency and extent of tabular output</td>
</tr>
<tr>
<td>46</td>
<td>PAUSE</td>
<td>Cause a temporary pause in processing</td>
</tr>
<tr>
<td>47</td>
<td>PRINT</td>
<td>Print flow rate, sources and statistical measures of variables</td>
</tr>
<tr>
<td>48</td>
<td>QUIT</td>
<td>Terminate solution process</td>
</tr>
<tr>
<td>49</td>
<td>READ</td>
<td>Read initial conditions from archive file</td>
</tr>
<tr>
<td>50</td>
<td>REFERENCE</td>
<td>Reference values for key variables</td>
</tr>
<tr>
<td>51</td>
<td>RELAX</td>
<td>Relaxation factors for governing variables</td>
</tr>
<tr>
<td>52</td>
<td>RENAME</td>
<td>Allows renaming of output variables listed in Table 6.8.1</td>
</tr>
<tr>
<td>53</td>
<td>SAVE</td>
<td>Frequency of output to archive file</td>
</tr>
<tr>
<td>54</td>
<td>SCALE</td>
<td>Internal scaling of specified input</td>
</tr>
<tr>
<td>55</td>
<td>SCREEN</td>
<td>Echo some of the diagnostic output to screen</td>
</tr>
<tr>
<td>56</td>
<td>SELECT</td>
<td>Specify location of sub-regions or boundaries</td>
</tr>
<tr>
<td>57</td>
<td>SET</td>
<td>Set value of a variable as a function of space and time</td>
</tr>
<tr>
<td>58</td>
<td>SOLVE</td>
<td>Start of solution of equations</td>
</tr>
<tr>
<td>59</td>
<td>SOURCE</td>
<td>Specify source, injection or withdrawal terms</td>
</tr>
<tr>
<td>60</td>
<td>SPECIFIC HEAT</td>
<td>Select specific heat options and constants for the fluid</td>
</tr>
<tr>
<td>61</td>
<td>STACK</td>
<td>To specify stack transformation operations</td>
</tr>
<tr>
<td>62</td>
<td>STATISTICS</td>
<td>Obtain output of statistical parameters of variables</td>
</tr>
<tr>
<td>63</td>
<td>SYMMETRY</td>
<td>To identify an external boundary as a symmetry plane or axis</td>
</tr>
<tr>
<td>64</td>
<td>TIDE</td>
<td>Specify history of tide at boundary</td>
</tr>
</tbody>
</table>

... Table 6.1.1 continued
### TABLE 6.1.1: KEYWORDS OF TIDAL™ AND THEIR FUNCTIONS (continued)

<table>
<thead>
<tr>
<th>NO.</th>
<th>KEYWORD</th>
<th>INPUT FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>65</td>
<td>TIME</td>
<td>Set initial time for simulations</td>
</tr>
<tr>
<td>66</td>
<td>TITLE</td>
<td>Problem title specification</td>
</tr>
<tr>
<td>67</td>
<td>TRAC</td>
<td>Compute Particle tracks and corresponding elapsed time in flow field</td>
</tr>
<tr>
<td>68</td>
<td>USER</td>
<td>User identification for input and output files</td>
</tr>
<tr>
<td>69</td>
<td>WRITE</td>
<td>Generate output of variables</td>
</tr>
</tbody>
</table>
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** 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 TIDAL™ COMMANDS

<table>
<thead>
<tr>
<th>ORDER</th>
<th>FUNCTION</th>
<th>RELATED KEYWORD COMMAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Identification</td>
<td>BANNER, TITLE, USER</td>
</tr>
<tr>
<td>2</td>
<td>Grid specification</td>
<td>CONNECTIVITY, COORDINATE, GEOMETRY, GRID, SCALE</td>
</tr>
<tr>
<td>3</td>
<td>Problem definition</td>
<td>BLOCKAGE, GRAVITY, INCLUDE, INLET, LOCATE, MATERIAL, OPEN, OPTION, OUTLET, RENAME, SELECT, SYMMETRY</td>
</tr>
<tr>
<td>4</td>
<td>Initial and boundary conditions</td>
<td>BOUNDARY, FIX, INITIAL, INLET, OPEN, OUTLET, READ, SET, SYMMETRY, TIDE, TIME</td>
</tr>
<tr>
<td>5</td>
<td>Fluid Properties and constants</td>
<td>CONDUCTIVITY, DENSITY, DIFFUSIVITY, REFERENCE, SPECIFIC</td>
</tr>
<tr>
<td>6</td>
<td>Nature of Flow</td>
<td>CORIOLIS, FRICTION</td>
</tr>
<tr>
<td>7</td>
<td>Source and sink specifications</td>
<td>DECAY, SOURCE</td>
</tr>
<tr>
<td>8</td>
<td>Solution options</td>
<td>DISABLE, INTEGRATION, LIMIT, MATRIX, RELAX</td>
</tr>
<tr>
<td>9</td>
<td>Output control</td>
<td>CLOSE, CORRELATION, DEBUG, DIAGNOSTIC, FLUX, HISTORY, PRINT, OUTPUT, SAVE, SCREEN, STACK, STATISTICS, TRACK, WRITE</td>
</tr>
<tr>
<td>10</td>
<td>Operational control</td>
<td>ALLOCATE, CONVERGENCE, DEFINE, ELSE, END, ENDF, FILE, IF, INDENT, PAUSE, QUIT, SOLVE</td>
</tr>
</tbody>
</table>

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 **HISTORY** 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

<table>
<thead>
<tr>
<th>UNIT #</th>
<th>DEFAULT FILE NAME</th>
<th>FUNCTION OF THE FILE</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>Console</td>
<td>Read user input commands from standard input device.</td>
</tr>
<tr>
<td>16</td>
<td>xxx.out</td>
<td>Write to Standard Output Device. The string <code>xxx</code> in this file name is replaced by the file name (without extension) of the input command file specified by the user.</td>
</tr>
<tr>
<td>81</td>
<td>ACRINIT.ACR</td>
<td>ACRi Initialization file</td>
</tr>
<tr>
<td>82</td>
<td>ACRIMSG.ACR</td>
<td>ACRi file for diagnostic and error messages, and formats</td>
</tr>
<tr>
<td>Auto</td>
<td>xxx_LCNS.ACR</td>
<td>ACRi user License file. The string <code>xxx</code> is replaced by a set of characters that is unique to each user.</td>
</tr>
<tr>
<td>Auto</td>
<td>acr_DEBUG.TMP</td>
<td>User specific diagnostic and debug output (DEBUG).</td>
</tr>
<tr>
<td>Auto</td>
<td>acr_GRID_QUALITY.TMP</td>
<td>Grid quality information (DEBUG GEOMETRY).</td>
</tr>
<tr>
<td>Auto</td>
<td>acr_MTRXxxx.TMP</td>
<td>Solution Matrix Coefficients (DEBUG MATRIX). The string <code>xxx</code> is replaced by an identifier for the variable for which the output is generated</td>
</tr>
<tr>
<td>Auto</td>
<td>xxx_FLUX.TMP***</td>
<td>Write data for fluxes crossing a sub-region boundary (FLUX).</td>
</tr>
<tr>
<td>Auto</td>
<td>acr_GRID_XYZ.TMP</td>
<td>Internally generated ACRi format grid file (GRID PLOT3D).</td>
</tr>
<tr>
<td>Auto</td>
<td>xxx_DIAGNOSTIC.TMP***</td>
<td>Internally generated file with step by step diagnostics of convergence history for each variable for which a transport equation is solved (DIAGNOSTIC).</td>
</tr>
<tr>
<td>Auto</td>
<td>acr_HISTORY.TMP</td>
<td>Time-history data file (HISTORY).</td>
</tr>
<tr>
<td>Auto</td>
<td>acr_HISTORY_XYZ.TMP</td>
<td>Time-history at specified locations (HISTORY COORDINATE).</td>
</tr>
<tr>
<td>Auto</td>
<td>acr_SOURCE.TMP</td>
<td>Time-history of source of variables (HISTORY SOURCE).</td>
</tr>
<tr>
<td>Auto</td>
<td>acr_STORAGE.TMP</td>
<td>Time-history of storage of variables (HISTORY STORAGE).</td>
</tr>
<tr>
<td>Auto</td>
<td>acr_REDBLACK.TMP</td>
<td>Red/Black matrix split information (MATRIX REDBLACK).</td>
</tr>
<tr>
<td>Auto</td>
<td>acr_NOZZLE.TMP</td>
<td>Nozzle spray trajectory information (NOZZLE TRAJECTORY).</td>
</tr>
<tr>
<td>Auto</td>
<td>acr_PRINT_FORCE.TMP</td>
<td>Time-history of forces or moments on a surface (PRINT FORCE).</td>
</tr>
<tr>
<td>11</td>
<td>xxx.SAV***</td>
<td>Write data file for restart, archiving and post-processing (SAVE).</td>
</tr>
<tr>
<td>Auto</td>
<td>xxx_RESTART.TMP***</td>
<td>ACRi restart file (SAVE RESTART, READ RESTART).</td>
</tr>
<tr>
<td>12</td>
<td>xxx_TABLE.SAV***</td>
<td>Write archive data file in tabular form (SAVE TABLE).</td>
</tr>
<tr>
<td>Auto</td>
<td>acr_STATS.TMP</td>
<td>Statistics information pertaining to variables (STATISTICS).</td>
</tr>
<tr>
<td>Auto</td>
<td>xxx_TRACK.TMP***</td>
<td>Particle Track information for the flow field (TRACK).</td>
</tr>
<tr>
<td>Auto</td>
<td>acr_XYZ_CRNR.TMP</td>
<td>Corner coordinates for each element (WRITE CORNERS).</td>
</tr>
<tr>
<td>Auto</td>
<td>acr_VRTX_MAP.TMP</td>
<td>Vertex Mapping information automatically generated if needed.</td>
</tr>
</tbody>
</table>

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

TIDAL™ 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 TIDAL™**

<table>
<thead>
<tr>
<th><strong>TIDAL™ Name</strong></th>
<th><strong>Mathematical Symbol</strong></th>
<th><strong>Description of the Variable</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td>U</td>
<td>Velocity component in x-direction</td>
</tr>
<tr>
<td>V</td>
<td>V</td>
<td>Velocity component in y or r-direction</td>
</tr>
<tr>
<td>W</td>
<td>W</td>
<td>Velocity component in z or θ-direction</td>
</tr>
<tr>
<td>ETA</td>
<td>η</td>
<td>Fluid elevation above datum</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>Fluid Temperature</td>
</tr>
<tr>
<td>C</td>
<td>C</td>
<td>Mass concentration of 1st species in fluid</td>
</tr>
<tr>
<td>C2, C3, ... , Ck</td>
<td>Ck</td>
<td>Mass concentration of 2nd through kth species in fluid</td>
</tr>
<tr>
<td>User defined</td>
<td>none</td>
<td>User-defined variables with ALLOCATE or RENAME commands</td>
</tr>
</tbody>
</table>
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 corners (IMAX-1 times JMAX-1). For an unstructured grid the vertex coordinate file (CONNECTIVITY command) determines the number of vertices.

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

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

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

**Output for Node Based Variable**: Table 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 OUTPUT and SAVE commands.

**Output for Suppilimentary Variables**: Table 6.8.2 lists some supplementary variables for which output can be obtained. The output for these can be directed to the Standard Output Device or a user-specified file. This output is controlled by the WRITE command.

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

**Output for Coefficients and Components of Transport Equation**: Output for different coefficients and components of each transport equations can be obtained during the solution process. Please see the OUTPUT and SAVE commands for a detailed description of this output.
### TABLE 6.8.1: STANDARD VARIABLES OF TIDAL™

<table>
<thead>
<tr>
<th>TIDAL™ Name</th>
<th>Mathematical Symbol</th>
<th>Description of the Variable</th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td>U</td>
<td>Velocity component in x-direction</td>
</tr>
<tr>
<td>V</td>
<td>V</td>
<td>Velocity component in y or r-direction</td>
</tr>
<tr>
<td>W</td>
<td>W</td>
<td>Velocity component in z or θ-direction</td>
</tr>
<tr>
<td>ETA</td>
<td>η</td>
<td>Fluid elevation above datum</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>Fluid Temperature</td>
</tr>
<tr>
<td>C</td>
<td>C</td>
<td>Mass concentration of 1st species in fluid</td>
</tr>
<tr>
<td>C2,C3, ..., Ck</td>
<td>C^k</td>
<td>Mass concentration of 2nd through kth species in fluid</td>
</tr>
<tr>
<td>H</td>
<td>H</td>
<td>Total fluid depth = h + η</td>
</tr>
<tr>
<td>DEPT</td>
<td>h</td>
<td>Bathymetric depth below datum</td>
</tr>
<tr>
<td>PA</td>
<td>P_a</td>
<td>Atmospheric pressure</td>
</tr>
<tr>
<td>WIND</td>
<td>w</td>
<td>Wind speed</td>
</tr>
<tr>
<td>ANGL</td>
<td>θ</td>
<td>Angle that the wind vector makes with the positive direction of x</td>
</tr>
<tr>
<td>RHO</td>
<td>ρ</td>
<td>Mass density of fluid</td>
</tr>
<tr>
<td>CP</td>
<td>C_p</td>
<td>Specific Heat of fluid mixture</td>
</tr>
<tr>
<td>SPEED</td>
<td>V</td>
<td>Modulus of Velocity = Sqrt(u^2+v^2+w^2)</td>
</tr>
<tr>
<td>EDYN</td>
<td>0.5 V^2</td>
<td>Dynamic energy of fluid</td>
</tr>
<tr>
<td>PDYN</td>
<td>0.5 ρ V^2</td>
<td>Dynamic pressure of fluid</td>
</tr>
<tr>
<td>PTOTAL</td>
<td>P+0.5 V^2</td>
<td>Total pressure of fluid</td>
</tr>
<tr>
<td>VOL</td>
<td>δV</td>
<td>Volume of the Element</td>
</tr>
<tr>
<td>X</td>
<td>x</td>
<td>X Coordinate or the Node Location</td>
</tr>
<tr>
<td>Y</td>
<td>y</td>
<td>Y Coordinate or the Node Location</td>
</tr>
<tr>
<td>Z</td>
<td>z</td>
<td>Z Coordinate or the Node Location</td>
</tr>
<tr>
<td>User defined</td>
<td>---</td>
<td>User-defined variables with ALLOCATE or RENAME commands</td>
</tr>
</tbody>
</table>
# TABLE 6.8.2: SUPPLEMENTARY VARIABLES OF TIDAL™

<table>
<thead>
<tr>
<th>TIDAL™ Name</th>
<th>Variable Basis</th>
<th>Variable Type</th>
<th>Description of the Variable</th>
</tr>
</thead>
<tbody>
<tr>
<td>XC, YC, ZC</td>
<td>Vertex</td>
<td>Real</td>
<td>x, y and z coordinates for element vertices</td>
</tr>
<tr>
<td>VOLF</td>
<td>Face</td>
<td>Real</td>
<td>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.</td>
</tr>
<tr>
<td>VOLR</td>
<td>Face</td>
<td>Real</td>
<td>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.</td>
</tr>
<tr>
<td>AFX, AFY, AFZ</td>
<td>Face</td>
<td>Real</td>
<td>The x, y and z components of the area vector for an element face.</td>
</tr>
<tr>
<td>FC</td>
<td>Face</td>
<td>Real</td>
<td>Convective mass flux across an element interface.</td>
</tr>
<tr>
<td>FD</td>
<td>Face</td>
<td>Real</td>
<td>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)</td>
</tr>
<tr>
<td>NCRN</td>
<td>Vertex</td>
<td>Integer</td>
<td>Element to vertex connectivity</td>
</tr>
<tr>
<td>NBRS</td>
<td>List</td>
<td>Integer</td>
<td>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.</td>
</tr>
<tr>
<td>NFACE</td>
<td>List</td>
<td>Integer</td>
<td>Element to surface number connectivity. For each element, the list contains the surface numbers that comprise the element boundary.</td>
</tr>
<tr>
<td>LINK</td>
<td>List</td>
<td>Integer</td>
<td>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.</td>
</tr>
<tr>
<td>MTYP</td>
<td>Node</td>
<td>Integer</td>
<td>Material Type index</td>
</tr>
</tbody>
</table>
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 output 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 specified frequency.
6.10 RESTART OUTPUT

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

<table>
<thead>
<tr>
<th>TITLE</th>
<th>EXAMPLE OF A TWO SEGMENT CALCULATION WITH OUTPUT CHANGES</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>// TRAVELLING SINE WAVE IN A UNIFORM BASIN</td>
</tr>
<tr>
<td></td>
<td>//////////////////////////////////////////////////////////////////////////////////////</td>
</tr>
<tr>
<td></td>
<td>/</td>
</tr>
<tr>
<td>GRID</td>
<td>52 by 3</td>
</tr>
<tr>
<td>COOR</td>
<td>dinate X RANGE = 20</td>
</tr>
<tr>
<td></td>
<td>/</td>
</tr>
<tr>
<td>LAND</td>
<td>FROM (1,1) TO (100,1)</td>
</tr>
<tr>
<td>LAND</td>
<td>FROM (100,1) TO (100,100)</td>
</tr>
<tr>
<td>LAND</td>
<td>FROM (1,1) TO (1,100)</td>
</tr>
<tr>
<td>LAND</td>
<td>FROM (1,100) TO (100,100)</td>
</tr>
<tr>
<td>INIT</td>
<td>ial DEPTH is 1 everywhere</td>
</tr>
<tr>
<td>GRAV</td>
<td>itational constant = 1</td>
</tr>
<tr>
<td>CORI</td>
<td>olis parameter = 0</td>
</tr>
<tr>
<td>TIDE</td>
<td>AT WEST boundary: COSine with 1 set: period=-10, amp=-0.1, phase=-2.5</td>
</tr>
<tr>
<td>HIST</td>
<td>ory at locations (2,2) (13,2) (26,2) (39,2) (50,2)</td>
</tr>
<tr>
<td>HIST</td>
<td>ory on file 'SINEWAVE.HIS'</td>
</tr>
<tr>
<td>DIAG</td>
<td>nostic node 2,2</td>
</tr>
<tr>
<td>OUTP</td>
<td>ut in NARROW mode</td>
</tr>
<tr>
<td></td>
<td>/</td>
</tr>
<tr>
<td>SOLV</td>
<td>ve for 7.52 non-dimensional units in steps of 0.07</td>
</tr>
<tr>
<td>SAVE</td>
<td>E on file: 'SINEWAVE.ARC'</td>
</tr>
<tr>
<td>OUTP</td>
<td>ut H E</td>
</tr>
<tr>
<td></td>
<td>/</td>
</tr>
<tr>
<td>SOLV</td>
<td>ve FOR 2.63 units in steps of 0.07</td>
</tr>
<tr>
<td>SAVE</td>
<td>E</td>
</tr>
<tr>
<td>OUTP</td>
<td>ut U, ETA and H</td>
</tr>
<tr>
<td></td>
<td>/</td>
</tr>
<tr>
<td>END</td>
<td></td>
</tr>
</tbody>
</table>
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.
CHAPTER 7

DESCRIPTION OF KEYWORD COMMANDS

This chapter describes the keyword commands that comprise the user's interface with the TIDAL™ software package. This interface is based on the ACRi format-free command language FREEFORM™. Knowledge of the structure and syntax of this language is essential for understanding the descriptions of keyword commands in this chapter. The command language is fully described in Appendix B of this User's Manual. Suggestions on the preparation of input for TIDAL™ 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_{\text{freq}}**  
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 “FORMATTED” mode. Some commands also allow an “UNFORMATTED” mode of input or output. If allowed, this mode is selected by the “UNFORMATTED” 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 LOCATE or SELECT 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 STATION 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 LOCATE or SELECT 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 LOCATE or SELECT command. The choices available for the 'subrgn' modifier are summarized in Table 7.2.2.

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

TABLE 7.2.1: VALID SUBREGION IDENTIFICATION MODIFIERS
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 command with a PAIRED 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 ($\xi$, $\eta$, $\zeta$ for a curvilinear grid). This orientation may or may not be related to the global framework in which the (x, y, z) coordinates of the element vertices are specified. The local orientation for an element is uniquely determined by the manner in which its vertices are numbered. The notation is illustrated in Figure 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 ($\xi$) vector points from vertex 1 to vertex 2, the local y ($\eta$) vector from vertex 1 to vertex 4 and, for 3D elements, the local z ($\zeta$) vector from vertex 1 to vertex 5. For a structured grid the local and global framework are consistent with each other. However, for an unstructured grid, the local framework may change from element to element since the vertex numbering may be completely arbitrary. In such a case a more general description is used which consists of the specification of a paired list of element and surface numbers (see LOCA 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**

<table>
<thead>
<tr>
<th>Orientation</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-</td>
<td>The outward normal is in the negative direction of the local x or $\xi$ coordinate. It is equivalent to the specification of surface number 1 of Figure 7.2.1.</td>
</tr>
<tr>
<td>X+</td>
<td>The outward normal is in the positive direction of the local x or $\xi$ coordinate. It is equivalent to the specification of surface number 2 of Figure 7.2.1</td>
</tr>
<tr>
<td>Y-</td>
<td>The outward normal is in the negative direction of the local y or $\eta$ coordinate. It is equivalent to the specification of surface number 3 of Figure 7.2.1</td>
</tr>
<tr>
<td>Y+</td>
<td>The outward normal is in the positive direction of the local y or $\eta$ coordinate. It is equivalent to the specification of surface number 4 of Figure 7.2.1</td>
</tr>
<tr>
<td>Z-</td>
<td>The outward normal is in the negative direction of the local z or $\zeta$ coordinate. It is equivalent to the specification of surface number 5 of Figure 7.2.1.</td>
</tr>
<tr>
<td>Z+</td>
<td>The outward normal is in the positive direction of the local z or $\zeta$ coordinate. It is equivalent to the specification of surface number 6 of Figure 7.2.1.</td>
</tr>
</tbody>
</table>
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:

\[ \Phi = \Phi(\xi) \]  

(7.2.1)

here \( \Phi \) represents a field variable, or a source or boundary value for a dependent variable, and \( \xi \) represents time, a spatial coordinate or a field variable. A library of built-in tabular and analytic functions for \( \Phi \) is provided. In addition, it is possible to specify user-defined functions as discussed in Section 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 \( \Phi \) is specified as an arbitrary function of \( \psi \) in a linear, piece-wise approximation:

\[ \Phi = A_1; \quad \xi_1 \leq \xi < \xi_2 \]
\[ \Phi = A_n + \alpha (A_{n+1} - A_n) \frac{\xi - \xi_{n-1}}{\xi_n - \xi_{n-1}}; \quad \xi_{n-1} < \xi < \xi_n; \quad 1 < n \leq N, \]
\[ \Phi = A_N; \quad \xi > \xi_N, \]  

(7.2.2)

where \( A_n \) are arbitrary constants and \( N \) is the total number of sets in the table. The factor \( \alpha \) 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 \( \xi \) used in the above interpolation relation is computed as:

\[ \xi_{used} = \text{mod}(\xi, \xi_N - \xi_i). \]

(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:

\[ \Phi_n = A_n; \quad 1 < n \leq N, \]  

(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, following analytic functions are currently available:

\[
\begin{align*}
\Phi &= A, \\
\Phi &= A + B \, \xi, \\
\Phi &= A + B \, \xi + C \, \xi^2 + D \, \xi^3 + E \, \xi^4, \\
\Phi &= A \left( \xi + B \right)^C + D, \\
\Phi &= A \sin \left( B \, \xi + C \right) + D, \\
\Phi &= A \cos \left( B \, \xi + C \right) + D, \\
\Phi &= A \tan \left( B \, \xi + C \right) + D, \\
\Phi &= A \arcsin \left( B \, \xi + C \right) + D, \\
\Phi &= A \arccos \left( B \, \xi + C \right) + D, \\
\Phi &= A \arctan \left( B \, \xi + C \right) + D, \\
\Phi &= A \exp \left( B \, \xi + C \right) + D, \\
\Phi &= A \ln \left( B \, \xi + C \right) + D, \\
\Phi &= \left( A + B \xi \right)^E \\
& \quad \left( C + D \xi \right)^F,
\end{align*}
\]

\[
\begin{align*}
\Phi &= A_0 + \sum_{n=1}^{N} A_n \left[ \xi + B_n \right] C_n, \\
\Phi &= A_0 + \sum_{n=1}^{N} A_n \sin \left[ 2\pi \frac{\xi + C_n}{B_n} \right], \\
\Phi &= A_0 + \sum_{n=1}^{N} A_n \cos \left[ 2\pi \frac{\xi + C_n}{B_n} \right], \\
\Phi &= A_0 + \sum_{n=1}^{N} A_n \tan \left[ 2\pi \frac{\xi + C_n}{B_n} \right], \\
\Phi &= A_0 + \sum_{n=1}^{N} A_n \exp \left[ -\frac{\xi + C_n}{B_n} \right], \\
\Phi &= A_0 + \sum_{n=1}^{N} A_n \ln \left[ \frac{B_n \xi + C_n}{B_n} \right].
\end{align*}
\]

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 BOUNDARY, CONDUCTION, DIFFUSION, DISTRIBUTION, FLOW, GAS, RETARDATION, SET, SOURCE, SPECIFIC heat and STORAGE 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 \( \xi \), respectively. Valid input options for \( \text{func} \) and \( \xi \) 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, \( \xi \) at the same location. However, if the modifier “STAC” is present on a command then the function is evaluated with the value of \( \xi \) at the most recent location specified by the STACK command.
### TABLE 7.2.3: VALID INPUT FUNCTIONAL FORMS

<table>
<thead>
<tr>
<th>User Specification of Function</th>
<th>Dependent Variable Computed From</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Function</strong></td>
<td><strong>Option</strong></td>
</tr>
<tr>
<td>TABL</td>
<td>Equation 7.2.2; $\alpha = 1$</td>
</tr>
<tr>
<td>TABL</td>
<td>STEP</td>
</tr>
<tr>
<td>TABL</td>
<td>PERI</td>
</tr>
<tr>
<td>TABL</td>
<td>STEP</td>
</tr>
<tr>
<td>NODE</td>
<td>Equation 7.2.4</td>
</tr>
<tr>
<td>CONS</td>
<td>Equation 7.2.5</td>
</tr>
<tr>
<td>LINE</td>
<td>Equation 7.2.6</td>
</tr>
<tr>
<td>POLY</td>
<td>Equation 7.2.7</td>
</tr>
<tr>
<td>POJE</td>
<td>Equation 7.2.8</td>
</tr>
<tr>
<td>SIN or SINE</td>
<td>Equation 7.2.9</td>
</tr>
<tr>
<td>COS or COSI</td>
<td>Equation 7.2.10</td>
</tr>
<tr>
<td>TAN or TANG</td>
<td>Equation 7.2.11</td>
</tr>
<tr>
<td>ASIN</td>
<td>Equation 7.2.12</td>
</tr>
<tr>
<td>ACOS</td>
<td>Equation 7.2.13</td>
</tr>
<tr>
<td>ATAN</td>
<td>Equation 7.2.14</td>
</tr>
<tr>
<td>EXP or EXPO</td>
<td>Equation 7.2.15</td>
</tr>
<tr>
<td>LN or LOG</td>
<td>Equation 7.2.16</td>
</tr>
<tr>
<td>RATI</td>
<td>Equation 7.2.17</td>
</tr>
<tr>
<td>POJE</td>
<td>Equation 7.2.18</td>
</tr>
<tr>
<td>SIN or SINE SERIes</td>
<td>Equation 7.2.19</td>
</tr>
<tr>
<td>COS or COSI SERIes</td>
<td>Equation 7.2.20</td>
</tr>
<tr>
<td>TAN or TANG SERIes</td>
<td>Equation 7.2.21</td>
</tr>
<tr>
<td>EXP or EXPO SERIes</td>
<td>Equation 7.2.22</td>
</tr>
<tr>
<td>LN or LOG SERIes</td>
<td>Equation 7.2.23</td>
</tr>
<tr>
<td>USER</td>
<td>User-defined function called for each element separately.</td>
</tr>
<tr>
<td>USER GLOBal</td>
<td>User-defined function called only once for the computational domain.</td>
</tr>
</tbody>
</table>
### TABLE 7.2.4: VALID INDEPENDENT VARIABLES

<table>
<thead>
<tr>
<th>User Specification</th>
<th>Denotes Independent Variable</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \xi )</td>
<td>Denotes Independent Variable</td>
</tr>
<tr>
<td>( t )</td>
<td>Denotes Independent Variable</td>
</tr>
<tr>
<td>( x )</td>
<td>Coordinate ( x )</td>
</tr>
<tr>
<td>( y )</td>
<td>Coordinate ( y )</td>
</tr>
<tr>
<td>( r )</td>
<td>Coordinate ( r )</td>
</tr>
<tr>
<td>( z )</td>
<td>Coordinate ( z )</td>
</tr>
<tr>
<td>( \theta )</td>
<td>Coordinate ( \theta )</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Numerical Value</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>N1</td>
<td>The number of sets of input in the table that are specified by Equation 7.2.2.</td>
</tr>
<tr>
<td>N2, ..., Nn</td>
<td>The pairs of values, ( \xi_n ) and ( \theta_n ) (in that order), for Equation 7.2.2. A total of 2*N1 values (N1 sets) must be specified.</td>
</tr>
</tbody>
</table>

### TABLE 7.2.6: NUMERICAL INPUT FOR SIMPLE ANALYTIC FUNCTIONS

<table>
<thead>
<tr>
<th>Numerical Value</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>N1, ..., Nn</td>
<td>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.</td>
</tr>
</tbody>
</table>

### TABLE 7.2.7: NUMERICAL INPUT FOR THE SERIES FUNCTIONS

<table>
<thead>
<tr>
<th>Numerical Values</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>N1</td>
<td>The number of sets of input in the series specified by Equations 7.2.18 through 7.2.23.</td>
</tr>
<tr>
<td>N2, ..., Nn-1</td>
<td>The triplet of values ( 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.</td>
</tr>
<tr>
<td>Nn</td>
<td>The datum ( A_0 ) for the function.</td>
</tr>
</tbody>
</table>

### TABLE 7.2.8: NUMERICAL INPUT FOR USER SPECIFIED FUNCTIONS

<table>
<thead>
<tr>
<th>Numerical Value</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>N1</td>
<td>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.</td>
</tr>
<tr>
<td>N2, ..., Nn</td>
<td>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.</td>
</tr>
</tbody>
</table>
### 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 `SOURCE` command would specify the source for that variable and, the `BOUNDARY` command may specify the value of the variable or that of the flux of the variable.

#### TABLE 7.2.9: EXAMPLES OF SIMPLE ANALYTIC FUNCTIONS

<table>
<thead>
<tr>
<th>Illustrative Commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>///// Constant boundary temperature of 30.</td>
</tr>
<tr>
<td>BOUNDary value at X+ is CONStant T = 30.</td>
</tr>
<tr>
<td>///// Source for temperature is a constant value for active subregion</td>
</tr>
<tr>
<td>SOURce for T: = 100 W/m^3 in currently SELEcted subregion</td>
</tr>
<tr>
<td>///// Field values of T set as a power law of pressure in previously identified subregion</td>
</tr>
<tr>
<td>SET T as POWER law: [1.020 * (P + 0.) ^ 0.50 + 100] in subregion ID=ZON1</td>
</tr>
<tr>
<td>///// Boundary temperature is a sinusoidal function of time</td>
</tr>
<tr>
<td>BOUNDary value at Y+: function: T = 10. * SIN (0.003 * TIME +0.5) +10.</td>
</tr>
<tr>
<td>///// Field values of temperature set as a polynomial function of coordinate X</td>
</tr>
<tr>
<td>SET T: POLYnomial in X: (10., 0.5, 0.,-0.1, 0.)</td>
</tr>
<tr>
<td>///// Temperature is a cosine function of time</td>
</tr>
<tr>
<td>SET T as 100 * COSIne (0.00274 * TIME )</td>
</tr>
<tr>
<td>///// Temperature is an exponentially decaying function of time</td>
</tr>
<tr>
<td>SET T as 100 * EXP (-0.001 * TIME ) +100.</td>
</tr>
<tr>
<td>///// Source for temperature is a linear function of Pressure</td>
</tr>
<tr>
<td>SOURce LINEar function (T = 0. -0.10 * P)</td>
</tr>
<tr>
<td>///// Source for temperature is a cosine function of time</td>
</tr>
<tr>
<td>SOURce T COSIne function of TIME: 100, 0.00274, 0., +10.5</td>
</tr>
<tr>
<td>///// Thermal conductivity for temperature is a linear function of temperature itself</td>
</tr>
<tr>
<td>CONDuctivity for T LINEar function: 0. -0.10 * T</td>
</tr>
</tbody>
</table>
### TABLE 7.2.10: EXAMPLES OF ANALYTIC FUNCTION SERIES

<table>
<thead>
<tr>
<th>Illustrative Commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>///// Boundary P is a function of T (temperature)</td>
</tr>
<tr>
<td>BOUNddy value for P at Y- boundary: COSIne SERIes in T: 3 sets</td>
</tr>
<tr>
<td>(0., 5, 1), (100., 10, 1), (200, 20, 3) datum = 10.</td>
</tr>
<tr>
<td>///// Field values for temperature set as sinusoidal function of Y coordinate</td>
</tr>
<tr>
<td>SET T as SINE SERIes in Y: 3 sets: (0, 5, 1), (100, 10, 1), (200, 20, 3)</td>
</tr>
<tr>
<td>///// Source for temperature is a 3 term cosine series in velocity U</td>
</tr>
<tr>
<td>SOURce T as COSIne SERIes in U: 3 sets</td>
</tr>
<tr>
<td>(0., 5, 1), (100., 10, 1), (200, 20, 3) datum = 10.</td>
</tr>
<tr>
<td>///// Boundary flux for T is a 3 part sinusoidal function of temperature</td>
</tr>
<tr>
<td>BOUNddy T at Y- boundary FLUX type: SINE SERIes in T: 3 sets</td>
</tr>
<tr>
<td>(0., 5, 1), (100., 10, 1), (200, 20, 3) datum= 0., h=1.E-4</td>
</tr>
<tr>
<td>///// Source for temperature is a function of temperature (itself)</td>
</tr>
<tr>
<td>SOURce T SINE SERIes in T: 3 sets: (0, 5, 1), (100, 10, 1), (200, 20, 3)</td>
</tr>
<tr>
<td>///// Source for T as an exponential function of time</td>
</tr>
<tr>
<td>SOURce T: EXPOnential SERIes in TIME; with 3 terms</td>
</tr>
<tr>
<td>(0.1, 1.0) (0.05, 0.1) (0.001,.01), base value = 0.</td>
</tr>
<tr>
<td>///// Field value of U velocity is set as a function of</td>
</tr>
<tr>
<td>SET value for U as COSIne SERIes in T: 3 sets</td>
</tr>
<tr>
<td>(0., 5, 1), (100., 10, 1), (200, 20, 3) datum = 10.</td>
</tr>
<tr>
<td>///// Pressure set as a 24 part cosine series; coefficients are contained in file named 'COSVALS'</td>
</tr>
<tr>
<td>SET P as COSIne SERIes of T: 24 sets from file 'COSVALS'</td>
</tr>
</tbody>
</table>
## TABLE 7.2.11: EXAMPLES OF TABULAR FUNCTIONS

<table>
<thead>
<tr>
<th>Illustrative Commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>/// The value of P at the boundary is a tabular function of time</td>
</tr>
<tr>
<td><strong>BOUNdary P value at Y- boundary: TABLe of values: 3 sets (TIME, value)</strong></td>
</tr>
<tr>
<td>(0., 0.01), (100., 0.10 ), (200, -0.20)</td>
</tr>
<tr>
<td>/// The flux of P at the boundary is a tabular function of time</td>
</tr>
<tr>
<td><strong>BOUNdary T at X-: FLUX is a TABle in TIME: 3 sets</strong></td>
</tr>
<tr>
<td>(0., 5), (100., 15 ), (200, 7.5), h=0.001</td>
</tr>
<tr>
<td>/// The field values of temperature are set as a tabular function of time</td>
</tr>
<tr>
<td><strong>SET T as TABLe of values: 3 sets: (0., 0.01), (100., 0.10 ), (200, -0.20)</strong>*</td>
</tr>
<tr>
<td>/// As above except that step-wise, rather than the linear, interpolation is used</td>
</tr>
<tr>
<td><strong>SET T is STEP wise TABLe: 3 sets: (0., 0.01), (100., 0.10 ), (200, -0.20)</strong>*</td>
</tr>
<tr>
<td>/// As above except that the values will be repeated after 200 time units</td>
</tr>
<tr>
<td><strong>SET T is PERIodic TABLe with STEP wise interpolation:</strong></td>
</tr>
<tr>
<td>3 sets: (0., 0.01), (100., 0.10 ), (200, -0.20)</td>
</tr>
<tr>
<td>/// Table with linear interpolation but time input set to result in steep step-like interpolation</td>
</tr>
<tr>
<td><strong>SET T for as TABLe of values: with 6 sets: (TIME, value)</strong></td>
</tr>
<tr>
<td>(0., 0.0) (1.0000, 0.0) (1.0001, 1.0)</td>
</tr>
<tr>
<td>(2., 1.0) (2.0001, 0.5) (3.0000, 0.2)</td>
</tr>
<tr>
<td>/// Table with linear interpolation but input from a file</td>
</tr>
<tr>
<td><strong>SET T for domain as function of TIME: 20 sets from file 'TIMEVALS'</strong>*</td>
</tr>
</tbody>
</table>
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 DENSITY, MATRIX, and VISCOSITY 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>
<thead>
<tr>
<th>User Specification of phase</th>
<th>Interpretation For The Corresponding Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIRS</td>
<td>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.</td>
</tr>
<tr>
<td>SECO</td>
<td>The input is for the second phase of the fluid.</td>
</tr>
<tr>
<td>THIR</td>
<td>The input is for the third phase of the fluid. This modifier is available only for the PORFLOW™ Software Tool.</td>
</tr>
<tr>
<td>GAS</td>
<td>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 SECOND modifier for a liquid-gas, and the THIRD modifier for a liquid-liquid-gas simulation.</td>
</tr>
<tr>
<td>VAPO</td>
<td>Same as GAS Modifier</td>
</tr>
</tbody>
</table>

7.2.11 Specification of Frequency for Output

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

\[ \text{ALLO} \{ \text{TABL} | \text{MATE} | \text{ZONE} \} \{N1\} \]

**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**

\[ \text{ALLO}cte \ \text{TABLE} \text{ space for 50000 words of problem specific input} \]
\[ \text{ALLO}cte \text{ space for 1000 MATERial Types} \]
\[ \text{ALLO}cte \text{ space for 1000 ZONEs} \]
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 or 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  

\[
\text{BLOC } \{\text{subrgn}\} \{ \text{MOVE | MOVI} \} \{ \text{FLUI | EXCL} \}
\]

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 \text{SET}, \text{INITIAL} or one of the other commands for specification of field values.

MOVI  Same as MOVE.

FLUI  See Mode 1 of command.

EXCL  See Mode 1 of command.

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_aver ] [ 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 \( \Psi \) is the specific gravity \((=\rho/\rho^*)\) 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: \( \alpha_4 = 0.003865456, \alpha_3 = -0.019667482, \alpha_2 = 0.039753091, \alpha_1 = -0.039825129, \alpha_0 = 0.019705054, \alpha_{-1} = -0.003827208. \)

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

<table>
<thead>
<tr>
<th>( \Psi = \rho/\rho^* )</th>
<th>1.0</th>
<th>1.05</th>
<th>1.10</th>
<th>1.15</th>
<th>1.20</th>
</tr>
</thead>
<tbody>
<tr>
<td>( dr/dt ) (m/s) x10^6</td>
<td>3.783</td>
<td>2.409</td>
<td>1.368</td>
<td>0.5371</td>
<td>0.01146</td>
</tr>
<tr>
<td>( dr/dt ) (m/day)</td>
<td>0.327</td>
<td>0.208</td>
<td>0.118</td>
<td>0.046</td>
<td>0.001</td>
</tr>
</tbody>
</table>

**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 \( \lambda \).

\( \lambda \)
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 \( \beta \).

\( \beta \)
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, \( \Psi \). By default, \( \Psi \) is computed at the fluid element next to the wall between the fluid and the solid (salt). If this modifier is present, \( \Psi \) is taken at a location which corresponds to the element directly in line with the wall element but located at the plane defined by the \( I, J, K, \) 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 $N * N_{23D}$ where $N$ is number of elements in the blocks and $N_{23D}$ 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.

<table>
<thead>
<tr>
<th>option</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>VALU</td>
<td>Value of the variable at the boundary node is specified (Equation 3.11.2). This is the default option.</td>
</tr>
<tr>
<td>FLUX</td>
<td>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.</td>
</tr>
<tr>
<td>GRAD</td>
<td>The normal gradient of the variable (Equation 3.11.3 with ( \Gamma = 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.</td>
</tr>
<tr>
<td>MIXE</td>
<td>A combination of the variable value and its gradient or flux (Equation 3.11.4) at the grid element interface is specified.</td>
</tr>
<tr>
<td>FLOW</td>
<td>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.</td>
</tr>
<tr>
<td>EXTR</td>
<td>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.</td>
</tr>
</tbody>
</table>

subopt: The nature of the MIXEd boundary condition.

<table>
<thead>
<tr>
<th>subopt</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLUX</td>
<td>Mixed boundary condition is specified in terms of the flux of the variable (Equation 3.11.4). This is the default option.</td>
</tr>
<tr>
<td>GRAD</td>
<td>Mixed boundary condition is specified in terms of the gradient of the variable (Equation 3.11.4 with ( \Gamma = 1 )).</td>
</tr>
</tbody>
</table>

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.
The boundary value of the variable (VALU option, \( F_0 \) 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; \( F_0 \) of Equation 3.11.4). There is no default value.

The transfer coefficient (\( h_F \) of Equation 3.11.4) for the MIXEd option. This input is ignored for other options. There is no default value.

In general, boundary conditions at any boundary segment may be specified only once. Two independent BOUNDARY specifications must not refer to the same element for the same boundary orientation index. Unpredictable consequences may arise if the boundary information for any segment is repeated. However, the specification for a boundary segment that was previously identified by a unique identity (ID=ids; see Section 7.2.4 and LOCATE command) may be replaced by a new specification provided the previously specified boundary condition is explicitly disabled by the BOUNDARY OFF (see Mode 3 specification) command.

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

**EXAMPLES**

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 VALUE, or MIXEd 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 GRADient 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 (BOUNDARY) must replace the keyword used in these examples. The boundary orientation (dir) identifier and the transfer coefficient must also be specified, if appropriate.

/// Some additional examples
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 at 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 ] [ 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:

\[ P_{\text{Total}} = P + \frac{1}{2} \rho \left( u^2 + v^2 + w^2 \right), \]

where \( P_{\text{Total}} \) is the total pressure, \( P \) is the local pressure, \( \rho \) 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.

\( \xi \)

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.

APPLICABILITY

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 (BOUNDARY) must replace the keyword used in these examples. The boundary orientation (dir) identifier and the transfer coefficient must also be specified, if appropriate.

///// Some additional examples

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:

- ηS > ηF  ➔  ηB = ηS;  VnB = VnF
- ηS ≤ ηF  ➔  ηB = ηF;  VnB = 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:

- ηF > ηS  ➔  ηB = ηS;  VnB = VnF
- ηF ≤ ηS  ➔  ηB = ηF;  VnB = 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 F0 of Equation 3.11.2 since this option can be used only to specify the VALUE 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.

APPLICABILITY

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 (BOUNDARY) must replace the keyword used in these examples and the ETA, GATE and one of the IN or OUT modifiers must be specified. The dir boundary orientation must also be specified.

/ / / An additional example (2 lines)

BOUNdary ETA at X+ is GATE IN type: SINE function of TIME
Amplitude=0.1 frequency: 1.405634E-4, phase=0; offset=0, for region ID=SOUTH
MODE 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 LOCATE or SELECT 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.

<table>
<thead>
<tr>
<th>filetype</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEBU</td>
<td>Debug output file (DEBUG) is selected for the operation.</td>
</tr>
<tr>
<td>FLUX</td>
<td>Flux output file (FLUX) is selected for the operation.</td>
</tr>
<tr>
<td>HIST</td>
<td>Time History file (HISTORY) is selected for the operation.</td>
</tr>
<tr>
<td>SAVE</td>
<td>Archive file (SAVE) is selected for the operation.</td>
</tr>
<tr>
<td>TABL</td>
<td>Tabulated archive file (SAVE TABLE) is selected for the operation.</td>
</tr>
<tr>
<td>TRAC</td>
<td>Particle track file (TRACK) is selected for the operation.</td>
</tr>
</tbody>
</table>

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  CLOS e SAVE file  
CLOS e archive TABLe output file now  
CLOS e particlle TRAC file immediately  
CLOS e file by name ‘MYOLDFILE’
COMMAND          CONDUCTIVITY

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

MODE 1:           Constant Conductivity or Diffusivity

SYNTAX            COND \( \{ \Phi \} \) \{ \Gamma_{\Phi} \} \{ \text{subrgn} \}

\( \Phi \)           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.

\( \Gamma_{\Phi} \)    The numerical value (>0) for conductivity 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 Conductivity for the temperature equation
CONDuctivity for T =1.E-6 for subdomain ID=BLK1 ! Thermal Conductivity for the temperature equation
CONDuctivity for C =1.E-6 ! Diffusivity for the species 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. 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 CONTACT Modifier.

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

\[
q_Φ = \frac{A_f (Φ_1 - Φ_2)}{\Gamma_1 + \Gamma_2 + \frac{1}{h}}
\]

where, \(q_Φ\) is flux from node “1” to node “2”, \(A_f\) is the interface area, \(ds_1\) and \(ds_2\) 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:

\[
q_Φ = q_Φ_{\text{internal}} + h A_f (Φ_1 - Φ_2)
\]

TOTA
The diffusive flux at the surface is computed as:

\[
q_Φ = h A_f (Φ_1 - Φ_2)
\]

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, αφ, βφ]

RICH

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

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

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

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

where \( g_i \) is the gravitational acceleration in the \( i^{th} \) direction (specified by the dir modifier), \( \rho \) 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:

\[
- \frac{\mu_\phi}{\rho} \frac{\partial p}{\partial x_i} \frac{\partial \rho}{\partial x_i}
\]

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

dir

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

UNLI

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

ZERO

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

α, β, αφ, βφ

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:

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

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

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

\[
Q_D = 0; \quad \Phi_B - \Phi_P = 0
\]

For highly skewed grids, above relation may be violated. If the modifier POSI is specified, then the diffusive flux \( Q_D \) is computed only from the values of \( \Phi_B \) and \( \Phi_P \) and any contribution of the 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 $\Gamma_{ij}$ are included in computations. It is assumed that the off-diagonal components ($i \neq j$) are zero. If this modifier is present, the off-diagonal components of the diffusion tensor are also included (see Mode 4 for mathematical expression of diffusive flux). It should be noted, that this mode automatically and simultaneously invokes Mode 4 of command.

This mode of the command is currently available only for the PORFLOW™ software.

Unless individual values of the diffusion components are specified by Mode 2 of the command, the general form of the diffusivity tensor is that given by Scheidegger (1961):

$$\Gamma_{ij} = \alpha_T \delta_{ij} V^j + (\alpha_L - \alpha_T) \frac{V_i V_j}{V},$$

$$V = (V_i V_i)^{1/2},$$

where $\alpha_L$ and $\alpha_T$ are, respectively, the longitudinal and transverse dispersivities for the porous matrix, $\delta_{ij}$ is the Kronecker delta function and, $V$ 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, $\theta_E$, 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 `BOUNDARY` modifier is specified, then the boundary value of conductivity or diffusivity, for all elements next to boundary, is set equal to the value at the interior node of the element.

**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

COND uction 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:

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

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

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

`CONN`ectivity information on file “VERT2ELM.CNC”
`CONN`ectivity of `VERT`ices to elements on file “VERT2ELM.CNC”
Assume that the vertex connectivity record for this element reads as follows:

<table>
<thead>
<tr>
<th>Cell #</th>
<th>Cell Type</th>
<th># of Corners</th>
<th>Vertex List</th>
</tr>
</thead>
<tbody>
<tr>
<td>?</td>
<td>1</td>
<td>4</td>
<td>1, 2, 3, 4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Side Number</th>
<th>Vertex Numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1, 4</td>
</tr>
<tr>
<td>2</td>
<td>2, 3</td>
</tr>
<tr>
<td>3</td>
<td>1, 2</td>
</tr>
<tr>
<td>4</td>
<td>3, 4</td>
</tr>
</tbody>
</table>

The right hand rule implies that: \( (\vec{r}_{12} \times \vec{r}_{13}) \cdot \vec{k} > 0 \), 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.)
Assume that the vertex connectivity record for this element reads as follows:

<table>
<thead>
<tr>
<th>Cell #, Cell Type, # of Corners,</th>
<th>Vertex List</th>
</tr>
</thead>
<tbody>
<tr>
<td>? 6 8</td>
<td>1, 2, 3, 4, 5, 6, 7, 8</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Side Number</th>
<th>Vertex Numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1, 4, 8, 5</td>
</tr>
<tr>
<td>2</td>
<td>3, 2, 6, 7</td>
</tr>
<tr>
<td>3</td>
<td>2, 1, 5, 6</td>
</tr>
<tr>
<td>4</td>
<td>4, 3, 7, 8</td>
</tr>
<tr>
<td>5</td>
<td>1, 2, 3, 4</td>
</tr>
<tr>
<td>6</td>
<td>5, 8, 7, 6</td>
</tr>
</tbody>
</table>

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

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

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

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

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

<table>
<thead>
<tr>
<th>Element Type</th>
<th># of Vertices</th>
<th># of Sides</th>
<th>Geometry</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2D</td>
<td>Triangle</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>4</td>
<td>2D</td>
<td>Quadrilateral</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>4</td>
<td>3D</td>
<td>Tetrahedron</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>5</td>
<td>3D</td>
<td>Quad-Based Pyramid</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>5</td>
<td>3D</td>
<td>Triangle-based Prism</td>
</tr>
<tr>
<td>6</td>
<td>8</td>
<td>6</td>
<td>3D</td>
<td>Hexahedron</td>
</tr>
</tbody>
</table>

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, NVVRTX,( NV(K),K=1,NVRTX)
```

where `IFILE` is an internally assigned file unit number, `M` is the element number, `MTYPE` is the element type, `NVVRTX` 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**  
Connetivity for HYBRID elements on file “MIXED_ELEMENTS.CNC”
Figure 3: Relation between Vertex Numbers and Side Numbers for a Triangle illustrating the application of the right hand rule. (2D only).

Triangle

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

<table>
<thead>
<tr>
<th>Cell #</th>
<th>Cell Type</th>
<th># of Corners</th>
<th>Vertex List</th>
</tr>
</thead>
<tbody>
<tr>
<td>?</td>
<td>1</td>
<td>3</td>
<td>1, 2, 3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Side Number</th>
<th>Vertex Numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1, 2</td>
</tr>
<tr>
<td>2</td>
<td>2, 3</td>
</tr>
<tr>
<td>3</td>
<td>3, 1</td>
</tr>
</tbody>
</table>

The right hand rule implies that: \( \vec{r}_{12} \times \vec{r}_{13} \cdot \vec{k} > 0 \), 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).
Figure 5: Relation between Vertex Numbers and Side Numbers for a Pyramid illustrating the application of the right hand rule. (3D only).
Figure 6: Relation between Vertex Numbers and Side Numbers for a Triangular Prism illustrating the application of the right hand rule. (3D only).
MODE 3: Connectivity for Quad or Hex Elements with Split Sides

SYNTAX

```
CONN { SPLI } {fname}
```

SPLI

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

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

The 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

```
1 2 3 4
5 12 11 10
6 7 8 9
```

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

<table>
<thead>
<tr>
<th>Element #</th>
<th>Vertex Connectivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1 2 6 5</td>
</tr>
<tr>
<td>2</td>
<td>2 3 7 6</td>
</tr>
<tr>
<td>3</td>
<td>3 4 8 7</td>
</tr>
<tr>
<td>4</td>
<td>5 6 10 9</td>
</tr>
<tr>
<td>5</td>
<td>6 20 17 18</td>
</tr>
<tr>
<td>6</td>
<td>7 8 12 11</td>
</tr>
<tr>
<td>7</td>
<td>9 10 14 13</td>
</tr>
<tr>
<td>8</td>
<td>10 11 15 14</td>
</tr>
<tr>
<td>9</td>
<td>11 12 16 15</td>
</tr>
<tr>
<td>10</td>
<td>20 7 19 17</td>
</tr>
<tr>
<td>11</td>
<td>17 19 11 21</td>
</tr>
<tr>
<td>12</td>
<td>18 17 21 10</td>
</tr>
</tbody>
</table>

Then the SPLIT connectivity command is:

CONNection 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:

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

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

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

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:

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

\[ \varepsilon = \max_j \left| X_j^{\text{new}} - X_j^{\text{old}} \right| \]

\[ \varepsilon = \max_j \left| \frac{X_j^{\text{new}} - X_j^{\text{old}}}{X_j^{\text{old}}} \right| ; \left| X_j^{\text{old}} \right| \geq X_{\text{ref}} \]

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 $\epsilon$ 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 $N_2$ 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 $MATR$ command.

If no variable ($\Phi$) is specified on this command, then the convergence tolerance applies to the total flow loop consisting of the momentum and mass flow equations (see also Mode 2 of this command). In this case, the number of iterations is interpreted as the total number of times the whole set of velocity and flow equations are sequentially solved till the convergence constraints are met.

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

EXAMPLES

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 LOCALi, 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 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 COUPLED 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 COUPLED 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 (CONVERGENCE REFERENCE command) exceeds a value of 10^{30} at any stage of the solution process. If this command is present, then the convergence threshold for termination is set to the value specified by 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**  \( \text{COORDINATE} \)

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

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

**SYNTAX**  \( \text{COOR} \{\text{RANG}\} \{\text{dir}\} \{\text{CYLI}\} \{\text{DEGR}\} \{\text{NODE}\} \{\text{N1}\} \{\text{N2}\} \)

**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 \(\text{THETA}\) modifiers that, respectively, denotes the \(x, y, z, r\) or \(\theta\) 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 \(\theta\) is assumed to be in radian. If the modifier \(\text{DEGREE}\) 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**

\[
\begin{align*}
\text{COORDinate X: RANGe} &= 10. \\
\text{COORDinate X: RANGe} &= 10., \text{ increase ratio} = 1.05 \quad \text{!!! expanding grid} \\
\text{COORDinate X: RANGe} &= 10., \text{ decrease ratio} = 0.95 \quad \text{!!! contracting grid} \\
\text{COORDinate Z: RANGe} &= 6.28 \text{ implement in the NODE mode} \\
\text{COORDinate R: RANGe} &= 10, \text{ ratio} = 0.95 \\
\text{COORDinate R: RANGe} &= 10, \text{ ratio} = 0.95 \text{ CYLindrical} \\
\text{COORDinate THETA: RANGe} &= 270. \text{ DEGREes} \\
\end{align*}
\]
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 into 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), (0,1,0) and (1,1,0) 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 xij must be in the order: x11, x21, x31, ..., xn1; x12, x22, x32, ..., xn2; ... and so on. If, say, both x and y coordinates for a 2D grid are simultaneously specified, then the set of values (xij, yij) must be in the order: (x11, y11), (x21, y21), (x31, y31), ... (xn1, yn1); (x12, y12), (x22, y22), (x32, y32), ... (xn2, yn2); ... and so on. However, if the JIK modifier is present, then it is assumed that the input data is in the order xijk. The I and J values are then transposed internally to the ACRi xijk 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 ($\xi$, $\eta$, $\zeta$) coordinates are identical with the global (X,R,$\theta$) coordinates. It is required that the external normal at Face number 3 points in the R⁻ and that at Face number 4 in the R⁺ direction. Further details are given in Section 7.2.5 and the CONNECTIVITY 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 collocated at the same point in space.

EXAMPLES

COORDinates of VERTices on file ‘PROBGRID’
COMMAND  CORIOLIS

PURPOSE  To specify the Coriolis parameter for the momentum balance equations.

SYNTAX  CORI  { Ω }  

Ω  The value of Ω (≥0), the coriolis parameter.  The default value is 0.

COMMENTS  This command is implemented only for the TIDAL™ Software Tool.

EXAMPLES  CORIolis parameter for this location is 0.05
COMMAND  CORRELATION

PURPOSE  Compute and output auto and cross correlations for dependent variables.

MODE 1:  Auto Correlation for the Variables

SYNTAX  CORR  \{ \Phi \}  [ subrgn ]  [fname] [TIME] [NOW] [Vfrq] [OFF]

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

\[
\langle \Phi \rangle = \frac{1}{N} \sum_{n=1}^{N} \Phi^n \\
\langle \Phi^2 \rangle = \frac{1}{N} \sum_{n=1}^{N} (\Phi^n)^2
\]

The following correlation quantities are then computed and printed:

\[
R_{11} = \langle \Phi^2 \rangle - \langle \Phi \rangle \langle \Phi \rangle \\
C_{11} = \frac{R_{11}}{\langle \Phi \rangle \langle \Phi \rangle} \\
D_{11} = \langle \Phi \rangle - \langle \Phi \rangle
\]

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 \texttt{SOLVE} command. If an exact time at which the output is due is not simulated, then the output is obtained at the first time step after the due time. By default the output is obtained only at the end of simulations.

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

EXAMPLES

CORRrelation for variable U for the SELECTed subregion at the end of simulations
CORRrelation for variable U for the ID=LOC_AUTO every 25 steps
CORRrelation OFF for variable U for the ID=LOC_AUTO
MODE 2: Cross Correlation for Two Variables

SYNTAX

\[ \text{CORR } \{ \Phi_1, \Phi_2 \} \ [\text{subrgn}] \ [\text{fname}] \ [\text{TIME}] \ [\text{NOW}] \ [V_{frq}] \ [\text{OFF}] \]

\( \Phi_1, \Phi_2 \) Two symbols that denote the variables for which correlation is desired. Valid symbols are listed in Table 6.7.1. If \( \Phi_k \) denotes the instant value of the \( k^{th} \) variable at a given location, \( \Phi_k^n \) 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:

\[
R_{12} = \langle \Phi_1 \Phi_2 \rangle - \langle \Phi_1 \rangle \langle \Phi_2 \rangle
\]
\[
C_{12} = \frac{R_{12}}{\sqrt{\langle \Phi_1^2 \rangle - \langle \Phi_1 \rangle^2} \sqrt{\langle \Phi_2^2 \rangle - \langle \Phi_2 \rangle^2}}
\]
\[
D_{12} = (\Phi_1 - \langle \Phi_1 \rangle)(\Phi_2 - \langle \Phi_2 \rangle)
\]

\text{subrgn} The subregion for computations. If no subregion is specified, the entire domain is selected.

\text{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.

\text{TIME} By default, \( V_{frq} \) is interpreted as the frequency of output in terms of number of steps. In the presence of \text{TIME} this is the time intervals between successive outputs.

\text{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.

\text{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 \text{SOLVE} command. If an exact time at which the output is due is not simulated, then the output is obtained at the first time step after the due time. By default the output is obtained only at the end of simulations.

\text{OFF} Any previously specified \text{CORRELATION} commands for the specified variable and subregion are disabled. New commands may be subsequently specified.

EXAMPLES

\text{CORR}elation for variable U and V for the SELEc ted subregion at the end of simulations
\text{CORR}elation for variable U and T for the ID=LOC\_AUTO every 25 steps
\text{CORR}elation OFF for variable U and T for the ID=LOC\_AUTO
MODE 3: Two Point Correlation for One or Two Variables

SYNTAX

CORR \{ TWO \} \{ \phi_1 \} \{ \phi_2 \} [subrgn] [fname] [TIME] [NOW] [Vfrq] [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 CORRELATION command.

\phi_1, \phi_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 \phi_1 and \phi_2. The \phi_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 \phi_2 is taken to be the values at the 2nd set of paired elements. If two symbols are specified then \phi_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 CORRELATION 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, Vfrq 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 Vfrq specification.

Vfrq

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 CORRELATION 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: CPU

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

SYNTAX: CPU \{ NCPU \}

NCPU: The number of CPU's for parallel processing. The default value is 1.

EXAMPLES:

CPU 4 CPU's for this computer system
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 skewness angles for the faces of the element. For each face the skewness angle is defined as the angle between the normal vector at the face and the vector connecting the two grid nodes, one on either side of the face. The angle is expressed in degrees. A value of 0 implies that the grid vector is orthogonal to the face.

\[
\alpha_f = \cos^{-1}\left(\frac{\sum_{i=1}^{N_{23}} A_i \delta x_i}{A_f \delta S}\right)
\]

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] [Vfrq] [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.

Vfrq
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, Vfrq is the computation (and output) frequency in terms of number of steps. If this modifier is present, then Vfrq 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 1st four characters of the symbolic name of the corresponding variable. If the name of the variable has fewer than 4 non-blank characters, then the trailing characters will stay as `xx`. The records on each file are in the order described below.

The first record consists of: the symbolic name of the variable, the order number of the variable in the master storage array, the step number at which the output is produced, the number of internal field elements, the number of neighbors for each element, the grid dimensions in each direction and, the total number of nodes.

This is followed by one record for each field element in sequence. Each record consists of: the value of the variable (\( X_j \)), the source term (part of \( B_i \) that pertains to the source or sinks in physical units), the forcing function (\( B_i \)), the diagonal component of the matrix coefficients (\( A_{ii} \)) and, the negative of the matrix coefficient for each neighbor (\(-A_{ij}, i \neq j\)).

**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 FIELD is present, then the decay reaction is assumed to occur only inside the field and not at the boundaries. The boundary values are maintained at the specified conditions.

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] [ CD ] [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:

$$S_Φ = -0.5 \rho \Phi C_D C_Φ \left( U^2 + V^2 + W^2 \right)^{\alpha/2},$$

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.

<table>
<thead>
<tr>
<th>option</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>VOLU</td>
<td>The coefficient $C_Φ$ in the $S_Φ$ term is set equal to the volume ($δV$) of the element.</td>
</tr>
<tr>
<td>AREA</td>
<td>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.</td>
</tr>
<tr>
<td>dir</td>
<td>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.</td>
</tr>
</tbody>
</table>
| NORM   | The coefficient $C_Φ$ in the $S_Φ$ term is computed as:

$$C_Φ = \sum_i A_i \cdot V_i$$

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 NORMAL 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

**CD**
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 NORMAL 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 DENSITY 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=OBStuction
DECAy of T: DRAG type: cf=0.001, N=0.5 multiply by AREA in X- direction for SELECTed subregion
DECAy of 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 { ρ₀ } [subrgn]

ρ₀  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/m³
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 ALWAYS 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 DENSITY 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] [p*] [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.

p*
Reference density for the fluid, \(\rho^*\).

α
The exponent for the POWER law or the coefficient for the POLYNOMIAL and LINEAR relations. The default value is 0.2 for the POWER law equation and 0 otherwise.

β
The critical temperature for the of the POWER law, the coefficient for the POLYNOMIAL relation or the reference concentration for the LINEAR relation. The default value is 374.15 for POWER law and 0 otherwise.

γ
The coefficient in the POLYNOMIAL and LINEAR relations. The default value is 0.

APPLICABILITY
This mode of the command is 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

\texttt{DENS \{GAS\} [INCO | COMP] [phase] [N1]}

\textbf{GAS} Density varies according to the gas law Equation:

\[
\rho = \frac{p + p^*}{R_u (T + T_a) \sum \frac{m_j}{M_j}}
\]

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 \( M_j \) is the corresponding molecular weight.

\textbf{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 \texttt{GAS} modifier is present. This is the default option for all ACRi Software Tools except \texttt{PORFLOW™}.

\textbf{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 \texttt{GAS} modifier is present. This is the default option for the \texttt{PORFLOW™} Software Tool

\textbf{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 \( 1^{st} \) phase of the fluid. This modifier is available only for the multi-phase versions of the \texttt{PORFLOW™} and \texttt{ANSWER™} Software Tools.

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

\textbf{APPLICABILITY} This mode of the command is not available for the \texttt{TIDAL™} Software Tool.

\textbf{EXAMPLES}

- DENSITY from \texttt{GAS} law: reference value = 0.960 kg/m^3
- DENSITY from \texttt{GAS} law reference value computed from other input
- DENSITY \texttt{GAS} law in \texttt{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:

\[
\frac{ρ}{ρ^*} = \min\{\exp(α C_S), ρ_{max}\},
\]

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

\( ρ^* \)

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 SCREEN command) by default. All 9 variables are directed to the screen if the WIDE modifier is present on the SCREEN command.

The default diagnostic output depends on the equations being solved and the nature of the problem. For transient problems, time is always printed as the first value. Then up to 8 (for transient) or 9 (for steady state) other diagnostic indicators are selected. The order of selection is: values of the active field variables, flux balances and matrix residue for the variables for which the governing equations are solved.

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 (BLOCK command), then the diagnostic element will be moved to a neighboring open element.

HIGH  If this modifier is present then additional diagnostics is provided about the allocation of memory for the storage of variables.

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 CONVERGENCE command. The residue in turn is related to the flux balance disparity in the solution of the differential equations. The flux balance disparity is composed of four components: the cumulative changes in storage, inflow, outflow and decay (see FLUX command).

The convergence index is normalized by a used specified error tolerance (CONVERGENCE REFERENCE command). A value of less than unity for this index means that the residue for the reference matrix equation(s) is smaller than the specified tolerance.

If the equations are solved in the steady state mode, then steady state is assumed to be reached when the index reaches a value of less than unity. However, caution should be exercised and the solution should also be examined for changes in the values of the key variables. The step to step values (or changes) of selected variables at a diagnostic node are also printed by this command. At steady state, there should be no further changes in the values of variables. In practice, often a true steady state (except for simple flows) can only be approached asymptotically. In this case a steady state is assumed to prevail provided the changes in the values of the variables are negligible compared to some norm of the solution.

If the equations are solved in the transient mode, then the index is a measure of the accuracy of the solution at each intermediate time step.

EXAMPLES

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’
DIAGnostic 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 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 CONDUCTION 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, \( \phi \), are always solved. This modifier may be used to disable flow calculations. Flow computation may also be disabled by appropriate specification of variables on the SOLVE command.

**ENTH**  By default if enthalpy is read from an archive file by a READ command, then any user input of temperature is ignored. Any new specification of temperature by the user is ignored. However, in some cases, it may be desirable to overwrite the specified enthalpy value by computing a new value from the temperature field. If this modifier is present, then enthalpy at the start of the solution process is recomputed from the temperature field. This modifier is active only for the ANSWER™ Software Tool.

**VELO**  By default velocity, pressure and density are corrected based on the continuity equation. If this modifier is present, then only the pressure and density are corrected; velocity is retained at its value computed from the momentum equations. This modifier is active only for the ANSWER™ Software Tool.

**DPDX**  The pressure gradient terms in the momentum equations for ANSWER™ software are set to zero. This is primarily a diagnostic tool to check the relative importance of various terms in the governing momentum equations. This modifier is active only for the ANSWER™ Software Tool.

**DENS**  If this modifier is present then the effect of density in the pressure equation is ignored everywhere except in the buoyancy term. This is equivalent to the so-called Boussinesq assumption. This modifier is active only for the PORFLOW™ Software Tool.

**ALL**  If this modifier is present along with the DENSity modifier, then the effect of density changes is ignored in all the heat and mass transport equations; that is the fluid density appearing in these equations is set equal to its reference value. This modifier is active only for the PORFLOW™ Software Tool.

**EXAMPLES**

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

\[ \text{DISA} \{\Phi\} [\text{CONV}] [\text{DIFF}] [\text{OFF}] \]

\(\Phi\) One or more symbols that denote the dependent variables for which the specified feature is disabled. There is no default value.

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

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)
<table>
<thead>
<tr>
<th><strong>COMMAND</strong></th>
<th>ELSE</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PURPOSE</strong></td>
<td>To define an IF construct to control which, if any, of one or two blocks of user input statements are executed.</td>
</tr>
<tr>
<td><strong>SYNTAX</strong></td>
<td>ELSE</td>
</tr>
</tbody>
</table>

**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     ENDF

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 modifier is present.

<table>
<thead>
<tr>
<th>attribute</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>FORM</td>
<td>File is opened in the FORMATTED mode. This is the default option.</td>
</tr>
<tr>
<td>UNFO</td>
<td>File is opened in the UNFORMATTED mode</td>
</tr>
<tr>
<td>UNKN</td>
<td>File status is defined as UNKNOWN. This is the default option.</td>
</tr>
<tr>
<td>NEW</td>
<td>File status is defined as NEW. An error will occur if the file already exists.</td>
</tr>
<tr>
<td>OLD</td>
<td>File status is defined as OLD. An error will occur if the file does not exist.</td>
</tr>
<tr>
<td>BOTH</td>
<td>File is available for both READ &amp; WRITE operations. This is the default option.</td>
</tr>
<tr>
<td>READ</td>
<td>File is available only for READ operations.</td>
</tr>
<tr>
<td>WRIT</td>
<td>File is available only for WRITE operations.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>filetype</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEBU</td>
<td>Debug output file (DEBUG), Unit 17, is selected for the operation.</td>
</tr>
<tr>
<td>FLUX</td>
<td>Flux output file (FLUX), Unit 14, is selected for the operation.</td>
</tr>
<tr>
<td>HIST</td>
<td>Time History file (HISTORY), Unit 13, is selected for the operation.</td>
</tr>
<tr>
<td>SAVE</td>
<td>The default Archive file (SAVE), Unit 11, is selected for the operation.</td>
</tr>
<tr>
<td>TRAC</td>
<td>Particle track file (TRACK), Unit 18, is selected for the operation.</td>
</tr>
</tbody>
</table>

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 UNIFORMatted READ only mode
FILE OPEN NEW SAVE file 'mynewsavefile.now' on in FORMatted WRITe only mode
FILE CLOS e SAVE HISTory file now
FILE CLOS e file by name 'OLDFILE.TMP'
FILE CLOS e 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 BOUNDARY command). Multiple commands may be used to fix values in an arbitrary manner. The INITIAL, READ or SET commands may be used to define the variable values. Once specified, these values remain unchanged unless modified by a subsequent INITIAL, READ or SET command.

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:

\[ Φ^n_P = N1 + N2Φ^n_p + N3Φ^n_w + N4Φ^n_e + N5Φ^n_s + N6Φ^n_n \]

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:

\[ Φ^n_P = N1 + N2Φ^n_p + N3Φ^n_w + N4Φ^n_e + N5Φ^n_s + N6Φ^n_n + N7Φ^n_d + N8Φ^n_u \]

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 INITial, 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 } \(\Phi\) [ subrgn ]

OFF

Previously specified FIX commands for \(\Phi\), 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 \(\Phi\) symbol or only a single \(\Phi\) symbol.

\(\Phi\)

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: FLUX

PURPOSE: To compute and obtain output of the flux balance for a dependent variable for a selected subregion within the flow domain.

MODE 1: Computation and Output of Flux Balance for a Variable

SYNTAX: FLUX \( \Phi \) [subrgn] [dir] [fname] [TIME] [Vfrq_file, Vfrq] [NOW]

\( \Phi \): 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, \( V_{frq-file} \) and \( V_{frq} \) are interpreted to be the frequency of output in terms of number of steps. If this modifier is present, then \( V_{frq-file} \) and \( V_{frq} \) are interpreted to be the time interval between successive outputs.

\( V_{frq-file} \): 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 \( V_{frq-file} \). 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 the 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, $\Phi$, the units of the cumulative flux are the units of $\Phi$ multiplied by the units of the density ($\rho$) and the units of volume ($L^3$). For PORFLOW™ the fluxes for mass and transport species (but not heat) are further divided by the density of the fluid; therefore these are in volumetric rather than mass units. The units of the instantaneous fluxes are those of the cumulative fluxes divided by units of time.

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. m³) 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/m³, 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 SELEceted region every 20 steps |
| FLUX for V in SELEceted 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 SELEceted region at TIME interval of 0.4 |
## Terms and Notation Used to Report Flux Balance in Output

<table>
<thead>
<tr>
<th>#</th>
<th>Term and Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Active Subdomain:</td>
<td>Particulars of the Sub-domain for which Flux Balance is given.</td>
</tr>
<tr>
<td>2.</td>
<td>Subdomain ID:</td>
<td>ID assigned to the Sub-domain by the user or the default ID automatically assigned if none given by User.</td>
</tr>
<tr>
<td>3.</td>
<td>Time:</td>
<td>The simulation Time at which the flux summary is computed.</td>
</tr>
<tr>
<td>4.</td>
<td>Net flux disparity (Qin-Qout-Qa-Qdecay)</td>
<td>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.</td>
</tr>
<tr>
<td>5.</td>
<td>Total inflow (Qin = Qc_in+Qd_in+Qsor)</td>
<td>Total inflow into the subregion by convective flux (See Item 11), diffusive flux (See item 12) and incoming (or positive) Sources (see Item 13).</td>
</tr>
<tr>
<td>6.</td>
<td>Total outflow (Qout = Qc_out+Qd_out+Qsnk)</td>
<td>Total outflow from the subregion by convective flux (Item 14), diffusive flux (Item 15) and Sinks or outgoing (or negative) Sources (Item 16).</td>
</tr>
<tr>
<td>7.</td>
<td>Net accumulated gain (Qa = Qn-Qo)</td>
<td>Net increase in the amount of property (storage) in the subregion from the start of simulations.</td>
</tr>
<tr>
<td>8.</td>
<td>Decay from start (Qdecay)</td>
<td>Net accumulated decay in the property in the subregion from the start of simulations.</td>
</tr>
<tr>
<td>9.</td>
<td>Total initial property in region (Qo)</td>
<td>Initial amount of property present in the subregion at start of simulations.</td>
</tr>
<tr>
<td>10.</td>
<td>Total property in region now (Qn)</td>
<td>Amount of property currently present in the subregion.</td>
</tr>
<tr>
<td>11.</td>
<td>Convective influx (Qc_in):</td>
<td>Net inflow of property due to convection from all boundaries of the subregion from the start of simulations.</td>
</tr>
<tr>
<td>12.</td>
<td>Diffusive influx (Qd_in):</td>
<td>Net inflow of property due to diffusion or dispersion from all boundaries of the subregion from the start of simulations.</td>
</tr>
<tr>
<td>13.</td>
<td>Source influx (Qsor):</td>
<td>Net inflow of property due to sources in the subregion from the start of simulations.</td>
</tr>
<tr>
<td>14.</td>
<td>Convective outflux (Qc_out)</td>
<td>Net outflow of property due to convection from all boundaries of the subregion from the start of simulations.</td>
</tr>
<tr>
<td>15.</td>
<td>Diffusive outflux (Qd_out)</td>
<td>Net outflow of property due to diffusion or dispersion from all boundaries of the subregion from the start of simulations.</td>
</tr>
<tr>
<td>16.</td>
<td>Sink outflux (Qsnk):</td>
<td>Net outflow of property due to sources or sinks in the subregion from the start of simulations.</td>
</tr>
<tr>
<td>17.</td>
<td>Flux disparity due to mass balance(div*F)</td>
<td>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.</td>
</tr>
<tr>
<td>18.</td>
<td>Change from last time step</td>
<td>Change in the amount for property from the last time step.</td>
</tr>
<tr>
<td>19.</td>
<td>Instantaneous convective flux (in-out):</td>
<td>The current net inflow of convective flux at all boundaries of the subregion.</td>
</tr>
</tbody>
</table>
**MODE 2:** Convective Flux and Flux-Average Value of a Variable

**SYNTAX**

\[
\text{FLUX} \{ \text{AVER} \} [\Phi] [\text{option}] [\text{subrgn}] [\text{dir}] [\text{fname}] [\text{TIME}] [\text{NOW}] [V_{frq}] [\text{OFF}]
\]

**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.

**\( \Phi \)**

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.

<table>
<thead>
<tr>
<th>option</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>INLE</td>
<td>Boundaries specified by the INLET command are selected.</td>
</tr>
<tr>
<td>OUTL</td>
<td>Boundaries specified by the OUTLET command are selected.</td>
</tr>
<tr>
<td>OPEN</td>
<td>Boundaries specified by the OPEN command are selected.</td>
</tr>
<tr>
<td>IO</td>
<td>All boundaries specified by INLET, OUTLET or OPEN command are selected.</td>
</tr>
<tr>
<td>WALL</td>
<td>Walls specified by WALL or BLOCK command are selected.</td>
</tr>
<tr>
<td>EXTE</td>
<td>All external (or outer) boundaries of the computational domain are selected.</td>
</tr>
<tr>
<td>ALL</td>
<td>All of the above boundaries are selected.</td>
</tr>
</tbody>
</table>

**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 \( \text{TIME} \) 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\} \{\(\Phi\)\} \{subrgn\}

OFF

Any previously specified flux balance command for the specified variable and subregion is disabled. New FLUX commands may be subsequently specified.

\(\Phi\)

The symbol for the dependent variable for which the previously specified flux-balance computation is disabled. The valid symbols are listed in Table 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**  
**FRIC**

**PURPOSE**  
To specify the bottom friction or wind stress coefficients.

**MODE 1:**  
Bottom Friction Coefficients

**SYNTAX**  
FRIC [ MANN | CHEZ ], { Cf1 }, [ Cf2 ]

**MANN**  
By default the bottom friction coefficient is assumed to be constant. If this modifier is present then the bottom friction is calculated according to Manning’s formula.

\[
C_f = g \left( \frac{C_{f1}}{C_{f2}} \right)^2 \frac{1}{H^{1/3}}
\]

**CHEZ**  
If this modifier is present then the bottom friction is calculated according to Chezy’s formula.

\[
C_f = g \left( \frac{1}{C_{f1}} \right)^2 \left[ C_{f2} + \frac{1}{\sqrt{H}} \right]^2
\]

**Cf1**  
When no modifier is present then this value is equal to the bottom friction coefficient that appears in the momentum equations. When either MANN or CHEZ is present it is equal to \( C_{f1} \) of the respective formula. The default value is 0.02.

**Cf2**  
When no modifier is present this input is ignored. When either MANN or CHEZ is present it is equal to \( C_{f2} \) of the respective formula. The default value is 1.0.

**COMMENTS**

This command is implemented only for the TIDAL™ Software Tool.

**EXAMPLES**

FRICtion coefficient on the bottom: 0.003  
FRICtion coefficient on the bottom: 0.003  
FRICtion coefficient CHEZY's formula: cf1=0.001, cf2 =1.  
FRICtion coefficient MANNing's formula: cf1=0.01, cf2 = 1.5.
MODE 2: Specification of Wind Stress Coefficients and Reference Wind Speed

SYNTAX FRIC { WIND } [ C_{w1}, C_{w2}, w_0 ]

WIND If this modifier is present then the input is for the wind stress coefficients and the reference wind speed according to the formula given below.

\[
C_w = C_{w1} + C_{w2} \left\{ \max \left[ 0 , \left( 1 - \frac{w_0}{w} \right)^2 \right] \right\}
\]

C_{w1} Linear component of wind stress coefficient, C_{w1}, in the equation given above. The default value is 1.0E-6.

C_{w2} Non-linear component of wind stress coefficient, C_{w2} in the equation given above. The default value is 1.4E-6.

w_0 Reference wind speed, w_0 in the equation given above. The default value is 7.0.

EXAMPLES

FRICtion parameters for WIND: C_{w1} = 2.4E-6, C_{w2} = 0.00, w_0 = 0 m/s
FRICtion - WIND parameters: all values are set to zero
COMMAND GEOMETRY

PURPOSE To modify or specify the geometry for the computational domain.

MODE 1: Exchange Previously Specified Coordinates

SYNTAX GEOM { EXCH } { dir1 } { dir2 }

EXCH The coordinates in the two directions specified by dir1 and dir2 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.

dir1, dir2 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 $\phi_1$ and $\phi_2$ degrees, respectively, according to the following equations:

$$
\begin{align*}
  x_{\text{new}} &= x_{\text{old}} \cos \phi_1 + y_{\text{old}} \cos \phi_2 \\
  y_{\text{new}} &= x_{\text{old}} \sin \phi_1 + y_{\text{old}} \sin \phi_2
\end{align*}
$$

dir_1, dir_2 Two of the X, Y, Z, R or THETA modifiers that, respectively, denote the x, y, z, r or $\theta$ coordinates to be rotated in the plane defined by dir_1 and dir_2.

N1 The angle $\phi_1$ (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 $\phi_2$ (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:

\[ x_{\text{new}} = x_{\text{old}} \times N1 + N2 \]

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 SCAlEt X by 0.3048
GEOMetry SCAlEt X by 0.3048 add 1.00
GEOMetry SCAlEt X and R by 0.3048 add 1.00
GEOMetry SCAlEt 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 CYLindrcal with radius = 1., theta = 90
GEOMETRY is CYLindrcal 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 value; a value must be specified.

**N2**

The inner radius of the annulus. The default value 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 value 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 } { dir1 } { dir2 }
```

CART  
Convert existing cylindrical radial coordinates to Cartesian coordinates according to the transformation:

\[ x_1 = r \cos \theta \]
\[ x_2 = r \sin \theta \]

RADI  
Convert existing Cartesian coordinates to cylindrical radial coordinates according to the transformation:

\[ r = \sqrt{x_1^2 + x_2^2} \]
\[ \theta = \tan^{-1}\left(\frac{x_2}{x_1}\right) \]

dir1, dir2  
Two of the X, Y, Z, R or THETA modifiers that, respectively, denote the x1 and x2 directions in the Cartesian framework to be transformed to or from the r and \( \theta \) directions of the cylindrical coordinate system. By default r direction is assumed to be aligned with the y direction and the \( \theta \) 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 x1 and \( \theta \) at x2. For transformation to a Cartesian system the resulting x1 is stored at the same location as r and x2 at \( \theta \).

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 { gx, gy, gz }, [ g ] [ RELA ]

gx, gy, gz The components of the gravitational acceleration vector, gj, 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 gj/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, gj is -9.81 for 2D flow and gj 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 gx, gy, gz are specified in a normalized mode.

RELA By default, for the ANSWER™ Software Tool, the gravitational body force term, Fi, 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 \( \rho^* \) is the reference density. However, the computed pressure, P, is no longer equal to the thermodynamic pressure, p, but is related to it by:

\[ P = p - \rho^* \left( g_x x + g_y y + g_z z \right) \]

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, $B_i$, 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 files. 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 2D solution mode is invoked and it is assumed that all subregion input specification is also in the 2D mode. There is no default value for this input.

EXAMPLES

GRID is 32 ! 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 \texttt{COORDINATE} command. The connectivity of the vertices to the elements is specified on a file through the \texttt{CONNECTIVITY} 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 \texttt{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 \texttt{ACRi} Software Tools from the connectivity information.

COMMENTS
All subregion input specification must be in terms of grid element numbers.

EXAMPLES
- \texttt{GRID} is UNST with 100 elements \ \ \ \ \ ! 2D problem
- \texttt{GRID} is UNSTructured with 1000 elements in THREE dimensional mode
- \texttt{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 CONNECTIVITY command is not required. The coordinates of the vertices must be specified on a file through the COORDINATE command.

**VERTEX**

The VERTEX mode of the CONNECTIVITY command is invoked. Please see the CONNECTIVITY command for further details. This is the default option.

**HYBRID**

The HYBRID mode of the CONNECTIVITY command is invoked. Please see the CONNECTIVITY command for further details.

**fname**

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

**THRE**

By default the problem is assumed to be a 2D problem. This modifier must be used to specify a 3D problem.

**EXAMPLES**

GRID CONNectivity of VERTices to elements on file “VERT2ELM.CNC”
GRID CONNnectivity 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**  \( \text{HISTORY} \)

**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**  \( \text{HIST} \ (\text{subrgn}) \ [\Phi] \ (\text{fname}) \ [\text{TABL}] \ [\text{PLOT}] \ [\text{NOW} | \text{OFF} | \text{ON}] \ [\text{TIME}] \ [\text{Vfrq}] \)

*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 \( \text{ID} = \text{subrgn} \) or a SELECT/LOCATE modifier. There is no default value.

*\( \Phi \)*  
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, \( \text{Vfrq} \) is interpreted as the frequency of output in terms of number of steps. If this modifier is present, then \( \text{Vfrq} \) is interpreted as the time interval between successive outputs.

**Vfrq**  
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).

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] [ Vfrq ]

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.

Vfrq
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}  \{\phi\}  [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.

\phi

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 COORDinates x=1.35, y=0.796 every 3 steps
HISTORY of U at point with COORDinates x=1.35, y=0.796 every 20 steps with inverse SQUARe interpolation
HISTORY of U, V, W, T at COORDinates 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 COORD (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\} \{\phi\} \{subrgn\} \{fname\} \{OFF\} \{Vfrq\} \{TIME\}

SOUR
Output for the source term for \(\phi\) is required.

\(\phi\)
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 \(\phi\) and subrgn is deactivated.

Vfrq
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, \(Vfrq\) is interpreted to be the frequency in terms of number of steps. If this modifier is present, then \(Vfrq\) 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 \(\phi\) over the volume of the subrgn.

The output is printed to the file specified by fname. In addition, at the end of simulations, tables of output are printed to the standard output file. These tables are sorted by variable and subregion.

EXAMPLES

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]  [Vfrq]  [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.

Vfrq  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, Vfrq is interpreted to be the frequency in terms of number of steps. If this
modifier is present, then Vfrq 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 INCLUDE command provides for unlimited nesting in that the file being incorporated may itself contain embedded INCLUDE commands. For this particular command the file name need not be included in single or double quotes.

If the user wants to specify a file name on-the-fly at run time, then a question mark (“?”) can be inserted anywhere on the command line. In this case, the user will be prompted for the file name at run time.

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 INCLUDE 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
**KEYWORD COMMANDS**

**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 if 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 cannot start with the first non-blank character as an alphabetic character (A through Z, in upper or lower case) otherwise the leading blanks will be ignored and the “continuation” command will be interpreted as a “Keyword” command. It is a safe practice to start the continuation lines with numeric or special non-alpha characters. A recommended practice is to always start a continuation command with an ampersand (&) character. For example, consider the sequence of commands:

```
INDEnt ON from now
   LOCAte ID=ZONE1 COORDinate from (0.0, 0.0, 0.0)
   &
   3.0, 2.0)
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
```

© TIDAL User's Guide - 216 - ACRi
**COMMAND** INITIAL

**PURPOSE** To specify the initial conditions for dependent variables for structured grids

**SYNTAX** INIT \{ Φ = Φ₀ \} [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.

Φ₀ 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 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 INITIAL, READ or SET commands. These values may be changed by a subsequent command during simulations. The only effect of this command is to keep the values at the inlet boundary fixed at the value set by the user.

EXAMPLES

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  \[ \phi=N1, \phi=N2, ..., \phi=\text{Nn} \]  \{ HYBR | COND | QUIC | CENT | ACRI | UPWI \}

\( \phi \)  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 2nd 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 2nd order central-difference scheme is used for integration. The scheme uses 2nd 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 1st order upwind scheme is used computing convective fluxes whereas the diffusive fluxes are still computed by the 2nd 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 \( R_{\text{max}} \) 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   LAND

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

COMMENTS   This command is identical to the BLOCK command. Please see the BLOCK command for details.
COMMAND LIMIT

PURPOSE To specify the limiting values for field variables

SYNTAX LIMI {Φ} {ΦMIN, ΦMAX | 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.

ΦMIN The minimum permissible value for the variable. Any computed value that is less than ΦMIN is set to ΦMIN.

ΦMAX The maximum permissible value for the variable. Any computed value that is greater than ΦMAX is set to ΦMAX.

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

LIMI for T minimum = 300
LIMI for T minimum = 0 maximum = 100
LIMI 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: LOCATE [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 LOCATE or SELECT on the command can make a generic reference to the specified subregion until the next LOCATE or SELECT command. See Sections 7.2.3 and 7.2.4 for additional details.

FIEL: If the LOCATE 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 modifier is present, then the subregion comprises only the interior field nodes (or elements) and the exterior boundary nodes are not included in the subregion.

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 FIELD 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 BOUNDary 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 ID = DMN2 for grid ID = DMN2 \((0., 0.)\) to \((2.5,1.5)\) and \((3.25,1.0)\) to \((5.0,5.2)\) select INTERsection & EXCLude
LOCAte ID = DMN2 \((0., 0.)\) to \((2.5,1.5)\) and \((3.25,1.0)\) to \((5.0,5.2)\) select NOT INTERsection & EXCLude
LOCAte ID = DMN2 \((0., 0.)\) to \((2.5,1.5)\) and \((3.25,1.0)\) to \((5.0,5.2)\) select output on “DMN2.LOC”
LOCAte ID = DMN2 \((0., 0.)\) 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 FIELD and BOUNDARY modifiers.

idsub  
See Mode 1 Specification.

EXCL  
If the modifier is present, then the specified elements are excluded and the complimentary set in the computational domain is selected.

FIEL  
See Mode 1 Specification.

BOUN  
If the BOUNDARY modifier is present, then only the external boundary nodes are selected. Any elements interior to the computational domain are excluded.

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 \( N_{23D} \times N_{vertex} \) where \( N_{23D} \) is 2 for 2D and 3 for 3D geometry and \( N_{vertex} \) is the number of vertices. There is no default value; a valid set of values must be specified.

fname  
See Mode 1 Specification.

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

\[
\text{LOCA} \quad \{\text{CYLI} \mid \text{CIRC}\} \quad [\text{ID}=\text{idsub}] \quad [\text{FIEL} \mid \text{BOUN}] \quad [\text{N1} \mid \text{N2} \ldots \text{Nn}] \quad [\text{IJK} \mid \text{ELEM}] \\
[\text{Nm+1} \ldots \text{Nm}], \quad [\text{Nm+1}] \quad [\text{fname}] 
\]

\text{CYLI} \mid \text{CIRC}  
A cylindrical (circular in 2D) subregion is specified. Only the internal elements can be specified. Associated boundary nodes are automatically included unless the \text{FIELD} modifier is present.

\text{idsub}  
See Mode 1 specification.

\text{FIEL}  
See Mode 1 Specification.

\text{BOUN}  
If the \text{BOUNDARY} modifier is present, then only the external boundary nodes are selected. Any elements interior to the computational domain are excluded.

\text{N1}  
The diameter of the cylinder of the identified subregion.

\text{N2} \ldots \text{Nn}  
In the absence of the \text{IJK} or \text{ELEM} modifier, these specify the (x,y) or the (x,y,z) coordinates of the center of the cylinder. In the presence of \text{IJK} or \text{ELEM} modifier these are interpreted as given below.

\text{IJK}  
The numerical input \([\text{N2} \ldots \text{Nn}]\) 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.

\text{ELEM}  
The numerical input \([\text{N2}]\) specifies the element which is at the center of the cylinder. Only 1 value must be specified.

\text{Nm+1} \ldots \text{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.

\text{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 \(\leq\) the radius of the cylinder, then the element is included \text{providing its normal distance from the plane is less than the tolerance.} By default the tolerance is set to a large number \((1.E+30)\) so that all elements in the projected plane of the circle are captured. This input may be used to specify a different tolerance to capture only elements within a certain distance from the plane. If specified, the intervening values for all components of the unit normal (even if zero) must be specified.

\text{fname}  
See Mode 1 Specification.

EXAMPLES

\text{LOCate CIRCle} \quad \text{dia}=0.4, \text{ center coordinates} \ (2., \ 0.5) \quad !2D \\
\text{LOCate CIRCle} \quad \text{dia}=0.4, \text{ center coordinates} \ (2., \ 0.5) \text{ external BOUNdary only} \quad !2D \\
\text{LOCate CIRCle} \quad \text{dia}=0.4, \text{ center coordinates} \ (2., \ 0.5, \ 0.5) \\
\text{LOCate CIRCle} \quad \text{dia}=0.4, \text{ center coordinates} \ (2., \ 0.5, \ 0.5) \ \text{normals} \ (1.,1.0.) \ \text{45 degree in xy} \\
\text{LOCate CYLinder} \quad \text{dia}=0.4, \text{ center coordinates} \ (2., \ 0.5, \ 0.5) \ \text{normals} \ (1.,1.0.) \ \text{half length}=0.5 \\
\text{LOCate CYLinder} \quad \text{dia}=0.4, \text{ center coordinates} \ (2., \ 0.5, \ 0.5) \ \text{normals} \ (1.,1.0.) \ \text{half length}=0.5 \ \text{file `Cylinder.loc’} \\
\text{LOCate CYLinder} \quad \text{dia}=0.4, \text{ center IJK at} \ (11,5,7) \ \text{normals} \ (1.,1.0.) \ \text{half length}=0.5 \\
\text{LOCate CYLinder} \quad \text{dia}=0.4, \text{ center ELEMen} at \ (147) \ \text{normals} \ (1.,1.0.) \ \text{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 FIELD 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

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOCAte SPHEre radius=0.4, center coordinates (2., 0.5)</td>
<td>2D</td>
</tr>
<tr>
<td>LOCAte SPHEre radius=0.4, center coordinates (2., 0.5, 0.5)</td>
<td>FIEL node only</td>
</tr>
<tr>
<td>LOCAte SPHEre radius =0.4, center coordinates (2., 0.5, 0.5 )</td>
<td>FIEL node only</td>
</tr>
<tr>
<td>LOCAte SPHEre radius =0.4, center coordinates (2., 0.5, 0.5 ) output to file ‘SPHERE.LOC’</td>
<td>FIEL node only</td>
</tr>
</tbody>
</table>
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 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 MATERIAL (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**

LOCA te MATERial type 3 as the active subregion
LOCA te ZONE number 5 as subregion with ID=TYP5
LOCA te 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` 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 COORD LIST 500 as ID=BIG from file ‘XYZ.LST’ only FIELD`  ! Read file; only field
- `LOCAte COORD 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 EXTERIOR 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**

```plaintext
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 FIELD modifier is present. For unstructured grids this is the default mode of the command; however the modifier must be specified for a structured grid.

**idsub**

See Mode 1 specification.

**FIEL**

See Mode 1 Specification.

**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**

```plaintext
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

LOCATE { SEQU } [ID=idsub] [FIELD] { 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 FIELD modifier is present. For unstructured grids this is the default mode of the command; however the modifier must be specified for a structured grid.

idsub See Mode 1 specification.

FIELD 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=LISTSEQ 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. CORRLEATION) 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 LOCATE or SELECT on the command can make a generic reference to the specified subregion until the next LOCATE or SELECT 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

LOCAtte CORRrelated Elements from ID=REGION1 and ID=REGION2
LOCAtte CORRrelated 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

```
LOCAta (element, surface) PAIRs:  (35,1), (53,3), (77,2), (13,4)
LOCAta PAIR : (35,1), (53,3), (77,2), (13,4) as boundary ID=BNDRy
LOCAta PAIR : 500 sets as ID=BNDRy from file ‘BOUNDARY.BIG’
LOCAta PAIR : (35,1), (53,3), (77,2), (13,4) as boundary ID=BNDRy
LOCAta PAIR : (35,1), (53,3), (77,2), (13,4), (28,5), (33,3), (35,6)
LOCAta 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

<table>
<thead>
<tr>
<th>option</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANY</td>
<td>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.</td>
</tr>
<tr>
<td>ONE</td>
<td>A boundary element is included if at least one of its vertices is included.</td>
</tr>
<tr>
<td>TWO</td>
<td>A boundary element is included if at least two of its vertices are included.</td>
</tr>
<tr>
<td>THRE</td>
<td>A boundary element is included if at least two of its vertices are included.</td>
</tr>
<tr>
<td>FOUR</td>
<td>A boundary element is included if at least two of its vertices are included.</td>
</tr>
</tbody>
</table>

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 SEQUENCE modifier, these numbers identify the vertices that define the boundary of interest. In the presence of the SEQUENCE 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 MATCHED PAIR or CORRELATED type of element subdomains or on previous LOCATE commands defined with a FIELD modifier.

EXAMPLES

LOCAt: VERTices 1,3,7,8,10,20,55,99,203,105,77
LOCAt VERTEc: in SEQUENCE from 23 through 231 in steps of 3 as boundary ID=BNDRy
LOCAt VERTEc: 500 values as ID=BNDRy from file ‘BOUNDARY.BIG’
LOCAt VERTEc: 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 (LOCATE PAIR, LOCATE VERTEX, LOCATE COORDINATE BOUNDARY and LOCATE POLYGON BOUNDARY 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

```
LOCAt e COMPlimentary ID=DMN1 as ID=DMN2
LOCAt e 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:

\[ C_0 = C_1 x + C_2 y + C_3 x^2 + C_4 y^2 + C_5 xy \]

\[ C_0 = C_1 x + C_2 y + C_3 z + C_4 x^2 + C_5 y^2 + C_6 z^2 + C_7 xy + C_8 yz + C_9 z^2 \]

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 C0 through C5 (for 2D) or C0 through C9 (for 3D). Any trailing coefficients not specified are assumed to be zero. At a minimum C0 through C2 (for 2D) or C0 through C3 (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 TOLEnANCE 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
LOCAte external BOUNdary SURFACE 2 = 1. *x + 1. * y + 0. * z  ! 2D mode
LOCAte SURFACE 1 = -1. x -1. y -1. z + 1. xx + 1. yy +1. zz +0. xy -1. yz  ! 3D mode
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

```plaintext
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**

```
{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 LOCATE 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 LOCATE 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 INTERIOR 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**

```plaintext
LOCAt MATCH X-direction of ID=DMN1 and X+ direction of ID=DMN1 as ID=MATCHED
LOCAt MATCH X-direction of ID=DMN1 and X- of ID=DMN2 INTERnal surfaces as ID=MATCH2
LOCAt 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 UNION {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 INTERSECTION 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 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.

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=DMNNM12
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 LOCATE LIST command) or boundary type of subdomains (such as LOCATE LIST PAIR). If the idsub1 specifies a set of elements, and idsub2 is not present, then any boundary definition that was originally present is still retained. This implies that if a reference is made to a boundary of modified idsub1 (e.g. a BOUNDARY command with the dir modifier), then the boundary of the original idsub1 will be selected. If however idsub2 is specified, then it will contain the definition of the modified boundary of the subdomain unless the FIELD modifier is also specified on the command.

EXAMPLES

LOCa te SOLIed elements ID=DMN1
LOCa te FLUIed elements of ID=DMN1 as ID=DMN2
LOCa te UNIQue elements ID=DMN1
LOCa te 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 "CURTAIN mode" (see INJECTION FILM and INJECTION CURTAIN 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 \( \frac{\mathbf{AB}}{AB} \mathbf{\hat{g}} \).

EXAMPLES

LOCAte INJExction 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 INJExction 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 INJExction on entire domain and define ID=W CURTAIN
with injection from (2.0, 2.7, 0.0) to (5.0,2.7,2.0)

LOCAte INJExction on ID=WALLRIGHT and define ID=W FILM
with injection from (2.0, 2.7, 0.0) to (5.0,2.7,2.0)

LOCAte INJExction on SELEcted surface and define ID=W FILM
with injection from (2.0, 2.7, 0.0) to (5.0,2.7,2.0)

LOCAte INJExction 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] { NMat } [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.

NMat: 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 more than 100 material types are required, then the ALLOCATE MATERIAL 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 FIELD 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] \{ NMat \} [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.

**NMat**
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 more than 100 material types require to be defined, then the **ALLOCATE MATERIAL** 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 **COORDINATE** 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] { NMat } {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.

NMat

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:

\[ \text{Area} = \frac{\pi}{4} (D^2 - d^2) \]

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, $\rho_s (> 0)$, 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 MATERIAL (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 OVERRIDE is present, then no internal check is made.

N1 The effective (or flow) porosity, $\Theta_E$. The default value is 1.

N2 The total porosity, $\Theta_T$. If no value is specified, total porosity is set equal to the effective porosity.

N3 The diffusional porosity, $\Theta_D$. If no value is specified, diffusional porosity is set equal to the effective porosity.

APPLICABILITY This command is currently available only for the PORFLOW™ Software Tool.

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 MATERIAL (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, \( \rho_s \) (> 0). The default value is 1.

N2 The effective (or flow) porosity, \( \Theta_E \). The default value is 1.

N3 The total porosity, \( \Theta_T \). If no value is specified, total porosity is set equal to the effective porosity.

N4 The diffusional porosity, \( \Theta_D \). If no value is specified, diffusional porosity is set equal to the effective porosity.

APPLICABILITY

This command is currently available only for the PORFLOW™ Software Tool.

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 MATERIAL (Mode 1) and ZONE commands.

This mode of the MATERIAL 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 \( \tau_{ij} \) 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 \[\phi=N1, \phi=N2, \ldots, \phi=Nn\] [dir] [option] [auxiliary]

\(\phi\) One or more symbols that denote the variable(s) for which the \(N1, N2, \ldots, \) 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 (CONVERGENCE command). The specified number of sweeps is always carried out. A sweep is more efficient than a full inner iteration because no matrix residue norms are computed. On the other hand a large number of sweeps may prove wasteful because, the sweeps are forced irrespective of the state of the solution.
EXAMPLES

MATRix sweeps in X 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 EXPlcit SOR
MATRix for T USER option 5 ! Matrix Solver supplied by User
MODE 2: Number of Inner Matrix Iterations

SYNTAX MATR {ITER} { N1 | \( \Phi = N1 \), \( \Phi = N2 \), ..., \( \Phi = Nn \) }

ITER The input pertains to the maximum number of inner iterations for the matrix.

\( \Phi \) 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 (CONVERGENCE command) is satisfied. During this process the matrix coefficients stay constant even though the value of the computed variable is changing. This process of solution is here named the "Inner Iteration". Each inner iteration consists of one pass through the matrix of equations followed by a check on the matrix residue (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.

<table>
<thead>
<tr>
<th>precon</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>NEUM</td>
<td>Neumann matrix polynomial. This is the default option.</td>
</tr>
<tr>
<td>LEAS</td>
<td>Least squares matrix polynomial.</td>
</tr>
<tr>
<td>DEGR</td>
<td>Degree of NEUMann or LEASt squares polynomial (integer) default: 3</td>
</tr>
<tr>
<td>REDU</td>
<td>Reduced System preconditioner.</td>
</tr>
<tr>
<td>CHOL</td>
<td>Incomplete Cholesky Factorization.</td>
</tr>
<tr>
<td>JACO</td>
<td>The point Jacobi preconditioner.</td>
</tr>
<tr>
<td>SOR</td>
<td>The Successive Over-Relaxation preconditioner and accelerator</td>
</tr>
</tbody>
</table>

accel

Accelerator Component of the Matrix Solver

<table>
<thead>
<tr>
<th>accel</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONJ</td>
<td>Conjugate Gradient accelerator.</td>
</tr>
<tr>
<td>BCGS</td>
<td>The Biconjugate Gradient Squared accelerator.</td>
</tr>
<tr>
<td>GMRE</td>
<td>GMRES accelerator. This is the default option.</td>
</tr>
<tr>
<td>ORTH</td>
<td>ORTHOMIN accelerator.</td>
</tr>
<tr>
<td>CGNR</td>
<td>Conjugate Gradient applied to Normal Equations.</td>
</tr>
<tr>
<td>LANC</td>
<td>Lanczos with ORTHOMIN accelerator.</td>
</tr>
</tbody>
</table>

option

Modifier for the specification or matrix or solver method

<table>
<thead>
<tr>
<th>option</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>MODI</td>
<td>If CHOLESKY preconditioner is selected, then modified incomplete Cholesky decomposition is used; otherwise this input is ignored.</td>
</tr>
<tr>
<td>PERM</td>
<td>If CHOLESKY preconditioner is selected, then matrix is red-black permuted; otherwise this input is ignored.</td>
</tr>
</tbody>
</table>
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= NVector]


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.

<table>
<thead>
<tr>
<th>precon</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMG</td>
<td>Algebraic Multi-Grid Preconditioner. This is the default option.</td>
</tr>
<tr>
<td>CHOL</td>
<td>Incomplete Cholesky Factorization.</td>
</tr>
<tr>
<td>OIAG</td>
<td>Diagonal Scaling Preconditioner</td>
</tr>
<tr>
<td>SPAR</td>
<td>The Sparse Approximate Inverse Preconditioner.</td>
</tr>
</tbody>
</table>

accel Accelerator Component of the Matrix Solver

<table>
<thead>
<tr>
<th>accel</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMGA</td>
<td>The AMG accelerator.</td>
</tr>
<tr>
<td>BICG</td>
<td>The Bi-Conjugate Gradient accelerator</td>
</tr>
<tr>
<td>CONJ</td>
<td>Conjugate Gradient accelerator.</td>
</tr>
<tr>
<td>GMRE</td>
<td>GMRES accelerator. This is the default option.</td>
</tr>
</tbody>
</table>

option Modifier for the specification of matrix or solver method

<table>
<thead>
<tr>
<th>option</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYMM</td>
<td>By default the matrix is assumed to be non-symmetric. If this modifier is present, then the matrix is assumed to be symmetric.</td>
</tr>
<tr>
<td>INDE</td>
<td>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 INTEGRATION Command)</td>
</tr>
<tr>
<td>STAT</td>
<td>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 STATic modifier will lead to a more efficient set up strategy. This modifier is ignored if any preconditioner other than AMG is specified.</td>
</tr>
</tbody>
</table>

VECT This modifier defines the number of vectors in the Krylov space that are saved.

NVector 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 coefficients of the matrix of equations. The default value is 1.E-20.

ZERO
The machine zero parameter is specified. This parameter is used by the NSPCG and HYPRE matrix solver packages to perform various numerical tests. In general, it treats any real number smaller in magnitude than the specified value as equivalent to machine zero. It is useful to modify this parameter in the instances where the right hand side of the equation system is less than the default value. If the tests indicate that normalized values of the right side of the matrix are smaller than this number then the matrix system may return without solving the system.

V_{Zero}  The numerical value for the machine zero. The default values are given in the table below.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.192 x 10^{-7}</td>
<td>IEEE real with 32 bit precision.</td>
</tr>
<tr>
<td>2.22 x 10^{-16}</td>
<td>IEEE real with 64 bit precision. This is the default.</td>
</tr>
<tr>
<td>7.1 x 10^{-15}</td>
<td>Cray XMP.</td>
</tr>
<tr>
<td>1.49 x 10^{-8}</td>
<td>Dec 10 (single precision)</td>
</tr>
<tr>
<td>4.768 x 10^{-7}</td>
<td>IBM 370 / 158 (single precision)</td>
</tr>
</tbody>
</table>

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=1.0E-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 **HYPRE** 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 **CHOLESKY** preconditioner, the typical values are between 0.001 and 0.01 and the default is set at 0.01. For the **AMG** preconditioner, the default value is 0.25 for 2D problems and 0.5 for 3D problems.

**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 **CHOLESKY**, **SPAR** or **AMG** methods have been selected (that is, the same numerical value is used for these methods if several of them are selected for different variables in a given simulation).

**EXAMPLES**

- MATRix for P from HYPRE with AMG preconditioner and CG accelerator
- MATRix for U from HYPRE with SPAI preconditioner and GMREs accelerator
- 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} { NLevel | Φ=NLevel1, Φ=NLevel2, ..., Φ=NLevend }

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.

NLevel

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 NLevel 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 CHOLESKY Preconditioner of the HYPRE 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.

\Phi

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 CHOLESKY, 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 NSPCG solvers, by default the specified convergence tolerance (CONVERGENCE command) is normalized with the larger of the machine zero or a norm of the current matrix residue. If this command along with the ABSO modifier is present, then the specified convergence tolerance is used as an absolute measure.

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: 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 INITIAL, READ or SET commands and is maintained even if the fluid leaves through the boundary.

EXAMPLES:

OPEN at Y+ boundary at maximum y
OPEN at X+ boundary for most recently SELECTed subregion
OPEN at Y- boundary for subregion ID = OPEN
OPEN OFF at Y- boundary for subregion ID = OPEN
### Command: OPTION COMMAND

**Purpose:** To select or modify built-in default options.

**Syntax:**
```plaintext
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 (SOLID 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 (SOLID 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:

\[ \text{Ma} = \frac{V}{c} \]

Here \( V \) is the fluid speed and \( c \) is the speed of sound which can be specified with the GAS SONIC command. This is the default option unless COMPRESSIBLE or REACTION command is specified. This modifier is effective only in the presence of the MACH modifier.

**GAMM**
The Mach number is computed from the relation:

\[ \text{Ma} = \frac{V}{\sqrt{(\gamma - 1)C_pT}} \]

Here \( V \) is the fluid speed, \( \gamma \) is the ratio of specific heat at constant pressure to that at constant volume, \( C_p \) is the specific heat at constant pressure and, \( T \) is the temperature in absolute units. The \( \gamma \) can be specified by the GAS GAMMA command. This modifier is effective only in the presence of the MACH modifier.

**GAS**
The Mach number is computed from the relation:

\[ \text{Ma} = \frac{V}{\sqrt{\frac{R}{C_p - R}C_pT}} \]

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 COMPRESSIBLE or REACTION 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 OUTLET boundaries is corrected so that the total outflow equals the net global inflow into the domain. For incompressible flow, the convergence of the solution process may be adversely affected if this correction is not performed. For compressible flow, this
may or may not be necessary based on the specified boundary conditions for pressure and/or density. This option is available only for ANSWER™ Software Tool.

OFF The outflow correction indicated by the OUTLET modifier is suppressed. For incompressible flow, convergence of the solution process may be adversely affected and is strongly influenced by the boundary conditions for pressure and/or density. On the other hand, for compressible flow, this modifier may improve the convergence of the solution process. This modifier is effective only in the presence of the OUTLET modifier for the ANSWER™ Software Tool.

LOCA The mass balance at each of the OUTLET 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 OUTLET modifier for the ANSWER™ Software Tool. The use of this modifier is not generally recommended.

VELO By default the corrections relating to the OUTLET boundaries are made by directly modifying outgoing convective flux; the outlet velocity is not corrected to correspond to the local values of density and boundary orientation. The outlet velocity is controlled by the specified or default boundary conditions. If this modifier is specified, then the outlet velocity is corrected so that it corresponds to the outflow flux. This modifier is effective only in the presence of the OUTLET modifier for the ANSWER™ Software Tool. The use of this modifier is not generally recommended.

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**  

```plaintext
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:

\[
Q_i = Q \sum_i F_i / F_i
\]

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 OUTLET 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] [Vfrq|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 SOLVE command is encountered. The default set includes the variables for which equations are solved, and some important supporting variables based on the nature of the problem.

plane  One of the character strings: XY, XZ, YZ, YX, ZX, YX, RX, 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 OUTPUT command is activated by default. If the ADD modifier is present, then the specified command is added to the list of active OUTPUT commands. On the other hand, if the ADD modifier is not specified, then the default command is modified or replaced by user specified command.

STAN  The output is directed to the Standard Output Unit. This the default option. The default name for this file is the name of the Input Command file but with the extension changed to "OUT". For example if the input command file name is "MYCASE.INP" or "MYCASE.DAT", then the default output file name will be "MYCASE.OUT".

fname  The name of the file to which the output is directed. If a file name is specified on an OUTPUT command with an ADD modifier, then this file is uniquely attached to a unit for output from that command and any subsequent OUTPUT command that specifies the same file name. The file name specification is case sensitive and the case must be consistent, otherwise some operating systems may report an error. See Section 7.2.2 for additional information.

Output from all OUTPUT command without an ADD modifier is directed to a single common file (by default, the Standard Output Unit). If any such command specifies a file name, then that file becomes the common output file for all commands without the ADD modifier. If a new file name or the STANDARD modifier is subsequently specified, then the previous file is closed and all subsequent output from all OUTPUT commands without an ADD modifier is directed to the new file or the Standard Output Unit.
subrgn  The subregion for which the output is required. For the plane type of output, the specified subregion must be in the mode of a grid index window (Mode 1 of LOCATE command). If any other type of subregion is specified, then the TABLE mode (MODE 2) of the command is automatically invoked. For TABLE type of output, any type of subregion can be specified. If no subregion is specified, the entire computational domain is selected. See 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.

Vfrq  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, Vfrq is the frequency of output in terms of number of steps. If this modifier is present, then Vfrq is interpreted to be time interval between successive outputs.

IMME  The output is produced immediately. This modifier is necessary only before the 1st SOLVE command is given. It then enables the output of the current value of specified variable(s) immediately. After the 1st SOLVE command, the output is always produced as soon as the command is encountered and this modifier is redundant. The symbol(s) for the output variable(s) must be specified with this modifier since the default output symbols are not available till after the 1st SOLVE command.

NOW  If this modifier is specified before the 1st SOLVE 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 SOLVE command. The output is always produced as soon as the command is encountered; this modifier is redundant.

ONLY  If this modifier is specified then the output is produced only once in response to a NOW or IMMEDIATE modifier or at the frequency determined by Vfrq.

OFF  Output for any previous command(s) for the same subrgn and type (plane or TABLE) is subsequently suppressed.

COMMENTS

If no OUTPUT command is specified then a command is automatically invoked to produce output for active variables at the end of simulations. The active variables consist of all variables for which the equations are solved plus other variables that are of interest to the problem being solved. Successive commands may be used to accommodate changing output requirements.

Any modifiers or numerical input specified on an OUTPUT command without the ADD modifier stay active for all commands without the ADD modifier unless changed by a subsequent command. For example, any output frequency specified on one such command will stay in effect till it is replaced by new value on a subsequent command.
EXAMPLES

OUTPUT U, V, W in NARrow tabular format
OUTPUT IMMEDIATELY the current values of U and V
OUTPUT T, V, U and P in WIDE tabular format NOW with variable STATistics
OUTPUT U, V, for subregion ID=OUTDomain in WIDE tabular format NOW with STATistics
OUTPUT U, V, T, P, K by XZ planes in WIDE format NOW for SELECTed subregion
OUTPUT OFF for all variables
OUTPUT NOW for subregion specified by the most recent LOCAte command
OUTPUT ONLY at step number 50
OUTPUT U, V, W as 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] [Vfrq] [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.

Vfrq See Mode 1 specification.

TIME See Mode 1 specification.

IMME The output is produced at the start 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 the 1st 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 TABLE) and same combination of COND, DIFF, STOR, SOUR, MATR, and RESI modifiers is subsequently suppressed.

COMMENTS

This mode of the OUTPUT command provides the ability to monitor the coefficients and contribution of the various components of the transport equation. Any combination of the modifiers COND, DIFF, STOR, SOUR, MATR, and RESI may be specified. At least one of these must be specified to trigger this mode of the command.

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

\texttt{OUTP \{\Phi\} \{FLUX\} \{CONV\} \{DIFF\} \{TOTA\} \{AREA\} \{plane\} \{TABL\} \{ADD\} \{STAN \fname\} \{subrgn\} \{STAT\} \{NOST\} \{NARR\} \{WIDE\} \{Vfrq\} \{TIME\} \{IMME\} \{NOW\} \{ONLY\} \{OFF\}}

\texttt{\Phi} 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.

\texttt{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.

\texttt{CONV} The convective flux for each face of the elements of the computational domain is printed.

\texttt{DIFF} The diffusive flux for each face of the elements of the computational domain is printed.

\texttt{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 \texttt{CONV} or \texttt{DIFF} modifiers are not present.

\texttt{AREA} The computed flux is divided by the projected area of the face for output.

\texttt{plane} See Mode 1 specification.

\texttt{TABL} See Mode 1 specification.

\texttt{ADD} This modifier is assumed by default. Each command in this mode, unless an \texttt{OFF} modifier is present, is treated as an additional command that adds to the active commands.

\texttt{STAN} See Mode 1 specification.

\texttt{fname} See Mode 2 specification.

\texttt{plane} See Mode 1 specification.

\texttt{subrgn} See Mode 1 specification.

\texttt{NARR} See Mode 1 specification.

\texttt{WIDE} See Mode 1 specification.

\texttt{Vfrq} See Mode 1 specification.

\texttt{TIME} See Mode 1 specification.

\texttt{IMME} See Mode 2 specification.

\texttt{NOW} See Mode 1 specification.

\texttt{ONLY} See Mode 1 specification.

\texttt{OFF} Output for any previous command(s) for the same \texttt{subrgn}, same type (\texttt{plane} or \texttt{TABL}) and same combination of \texttt{FLUX, CONV, DIFF} and \texttt{TOTA} modifiers is subsequently suppressed.
COMMENTS

This mode of the OUTPUT command provides the output of the convective, diffusive and total fluxes at the faces of the elements. Any combination of the choice of fluxes may be specified. **At least one of these must be specified to trigger this mode of the command.**

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

**OUTP**ut: FLUX for T for each element at the final stage
**OUTP**ut: CONVective FLUX for T NOW (at the next step) to file ‘CFLUX.TMP’
**OUTP**ut: CONVective and DIFFusive FLUX for T at step number 50 ONLY
**OUTP**ut: CONVective, DIFFusive and TOTAL FLUX divided by AREA for T every 50 steps
**OUTP**ut: FLUX divided by AREA for T at step# 52 ONLY in NARRow mode for ID=SUBREGION
**OUTP**ut: in TABLE mode FLUX for T in NARRow mode for ID=SUBREGION at end of simulations
KEYWORD COMMANDS

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] [Vfrq]
[TIME] [IMME | NOW | ONLY | OFF]
```

GRAD

The tensor components of the gradients of velocity, \( \varphi_{ij} \) 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, \( \omega \), is obtained, where:

\[
\omega_1 = \frac{\partial u_3}{\partial x_2} - \frac{\partial u_2}{\partial x_3}
\]
\[
\omega_2 = \frac{\partial u_1}{\partial x_3} - \frac{\partial u_3}{\partial x_1}
\]
\[
\omega_3 = \frac{\partial u_2}{\partial x_1} - \frac{\partial u_1}{\partial x_2}
\]

There are 3 components of vorticity for 3D flow but only one for 2D flow.

STRUCTURE

Output of a “vorticity” structure variable, \( \Omega \), is obtained. It is defined as:

\[
\Omega = -\frac{\partial u_1}{\partial x_2} \frac{\partial u_2}{\partial x_1}
\]

for 2D

\[
\Omega = -\frac{\partial u_1}{\partial x_2} \frac{\partial u_2}{\partial x_1} - \frac{\partial u_2}{\partial x_3} \frac{\partial u_3}{\partial x_2} - \frac{\partial u_3}{\partial x_1} \frac{\partial u_1}{\partial x_3}
\]

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, \( \tau_{ij} \), is obtained, where:

\[
\tau_{ij} = \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \delta_{ij} \sum_k \frac{\partial u_k}{\partial x_k} \right) - \rho \delta_{ij} p
\]

Here \( \mu \) is viscosity and \( p \) is the pressure. There are 3 components for 2D and 6 for 3D flow.

LIGHTHILL

Output of the components of “lighthill” stress tensor, \( L_{ij} \), is obtained, where:

\[
L_{ij} = -\mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \delta_{ij} \sum_k \frac{\partial u_k}{\partial x_k} \right) + \rho u_i u_j
\]

Here \( \mu \) is viscosity and \( \rho \) is the density. There are 3 components for 2D and 6 for 3D flow.

plane  See Mode 1 specification.

TABL  See Mode 1 specification.

ADD  This modifier is assumed by default. Each command in this mode, unless an OFF modifier is present, is treated as an additional command that adds to the active commands.

STAN  See Mode 1 specification.

fname  See Mode 2 specification.
plane  See Mode 1 specification.
subrgn See Mode 1 specification.
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 TABLE) 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      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
PAUS and await operator action
**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**

PRIN 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=Vtol\} \{NORM\} \{BASE=Vbase\} \{MASS | AREA | VOLU\} \{OUTP | fname\} \{Vfrq\} \{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 R = sqrt(y² + z²) 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 – Vtol ≤ S < S_i + Vtol 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 Vtol will be set of 0.05. However if the modifier TOLE is present, then the tolerance is set to the user specified value Vtol.

**Vtol**

The user specified value of tolerance. This input must be specified only if the TOLE modifier is present and then it must immediately follow the modifier.

**STAT**

By default, if the dir modifier is present, then the computed values are the arithmetic, area-weighted and mass-flux-weighted mean values; otherwise the computed values are arithmetic, volume-weighted and mass-weighted mean values. If this modifier is present, then computed values include the arithmetic mean, minimum, maximum and standard deviation of the variable.

**NORM**

The output for the selected variable is normalized or non-dimensionalized as:

\[ \phi_{output} = \frac{\phi_{computed} - \phi_{base}}{\phi_{norm} - \phi_{base}} \]

where \( \phi_{base} \) and \( \phi_{norm} \) are normalizing values. By default these are set to the minimum and maximum values for the sub-domain selected by subrgn and dir modifiers. However other options are available as described below.

**BASE**

The base value (\( \phi_{base} \)) for non-dimensional output. This modifier will automatically select the normalized (NORM) form of the output. If this modifier exists, then \( \phi_{base} \) must be specified and, in this case, only one variable (Φ) must be specified on the command.

**Vbase**

The user specified value for \( \phi_{base} \). This input must be specified only if the BASE modifier is present and then it must immediately follow the modifier.
MASS  This modifier is significant only if the BASE modifier is specified. By default $\Phi_{\text{norm}}$ is taken to be arithmetic mean of the values for the sub-domain. If this modifier is present, then $\Phi_{\text{norm}}$ is the mass or mass-flux weighted mean of the variable.

AREA  This modifier is significant only if the BASE modifier is specified. By default $\Phi_{\text{norm}}$ is the arithmetic mean of the values for the sub-domain. If this modifier is present, then $\Phi_{\text{norm}}$ 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.

Vfrq  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 Vfrq 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, Vfrq is the frequency of output in terms of number of steps. If this modifier is present, then Vfrq 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 $\Phi$ should be the symbol T (for Temperature) and the base value, $V_{\text{base}}$, should be specified as the average inlet temperature (commonly called T3). The $\Phi_{\text{norm}}$ 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

\[ \text{PRIN} \{ \Phi \} \{ \text{AVER} \mid \text{PROF} \} \{ N1 \} \{ N2,..,Nn \} \{ \text{coordinate} \} \{ \text{subrgn} \} \{ \text{dir} \} \{ \text{TOLE} = V_{tol} \} \{ \text{NORM} \} \{ \text{BASE} = V_{base} \} \{ \text{MASS} \mid \text{AREA} \mid \text{VOLU} \} \{ \text{OUTP} \mid \text{fname} \} \} \{ V_{frq} \} \{ \text{TIME} \} \]

\( \Phi \) 
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.

\( \text{AVER} \)
The averages of the specified variable are computed at a number of specified locations.

\( N1 \)
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 \( \text{coordinate} \) for the \( \text{subrgn} \). There is no default value.

\( \text{coordinate} \)
Same as Mode 2 of the command.

\( \text{subrgn} \)
Same as Mode 2 of the command.

\( \text{dir} \)
Same as Mode 2 of the command.

\( \text{TOLE} \)
The coordinate tolerance, \( V_{tol} \), for inclusion of the elements in averaging at a location. With \( S_i \) as the \( \text{coordinate} \) 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.

\( \text{STAT} \)
Same as Mode 2 of the command.

\( \text{NORM} \)
Same as Mode 2 of the command.

\( \text{BASE} \)
Same as Mode 2 of the command.

\( V_{base} \)
Same as Mode 2 of the command.

\( \text{MASS} \)
Same as Mode 2 of the command.

\( \text{AREA} \)
Same as Mode 2 of the command.

\( \text{VOLU} \)
Same as the \( \text{AREA} \) modifier.

\( \text{fname} \)
Same as Mode 2 of the command.

\( V_{frq} \)
Same as Mode 2 of the command.

\( \text{TIME} \)
Same as Mode 2 of the command.

EXAMPLES

\text{PRIN} \text{t PROFile of U, V, T at 5 X locations 0.1, 0.3, 0.4, 0.5, 0.9}
\text{PRIN} \text{t PROFile of STATistics for T at 4 Y values 0.1, 0.3, 0.5, 0.8 for ID=OUTLET in Y+ direction}
\text{PRIN} \text{t PROFile of NORmalized T at 4 X stations 0.1, 0.3, 0.5, 0.8 for ID=OUTLET in X+ direction}
\text{PRIN} \text{t AVERage for T at 4 R values 0.1, 0.3, 0.5, 0.8 for ID=OUTLET. in X+ direction BASE value = 375}
\text{PRIN} \text{t AVERage for T at 4 R values 0.1, 0.3, 0.5, 0.8 ID=SECTION1 BASE value = 375 MASS weighted}
\text{PRIN} \text{t AVERage for T at 3 R values 0.2, 0.4, 0.8 TOLErance=0.01 SELEcted region in X+ direction}
\text{PRIN} \text{t 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}
\text{PRIN} \text{t 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] [Xᵢ] [Φₒ] [Vfreq] [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:

\[ Fᵢ = \int (Φ + Φₒ) nᵢ \, dA \]
\[ Mᵢ = Fᵢ \times (Xᵢ - Yᵢ) \]

\( Fᵢ \) is the force vector for the Φ variable,
\( Φₒ \) is a reference datum for the Φ variable,
\( nᵢ \) is the normal vector at the surface of the subregion,
\( A \) is the area of the surface,
\( Mᵢ \) is the moment vector for the Φ variable,
\( \times \) denotes the cross product of vectors,
\( Xᵢ \) is the location of the pivot point around which the moment is computed,
\( Yᵢ \) is the point of intersection of the vector from \( Xᵢ \) and the force vector such that the vector from \( Xᵢ \) is normal to force vector. This is automatically computed.

MOME Same as FORC 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 selected. In this case the computed force will be the net force on the selected body.

\( Xᵢ \) 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.

Φₒ 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.

Vfreq 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, Vfreq 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] [Vfrq] [TIME] [NOW] [OFF]

FLUX

The convective and diffusive fluxes for a 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 variable 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.

<table>
<thead>
<tr>
<th>option</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>INLE</td>
<td>Boundaries specified by the INLET command are selected.</td>
</tr>
<tr>
<td>OUTL</td>
<td>Boundaries specified by the OUTLET command are selected.</td>
</tr>
<tr>
<td>OPEN</td>
<td>Boundaries specified by the OPEN command are selected.</td>
</tr>
<tr>
<td>IO</td>
<td>All boundaries specified by INLET, OUTLET or OPEN command are selected.</td>
</tr>
<tr>
<td>WALL</td>
<td>Walls specified by WALL or BLOCK command are selected.</td>
</tr>
<tr>
<td>EXTE</td>
<td>All external (or outer) boundaries of the computational domain are selected.</td>
</tr>
<tr>
<td>ALL</td>
<td>All of the above boundaries are selected.</td>
</tr>
</tbody>
</table>

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.

Vfrq

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, Vfrq 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 Vfrq 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] [Vfrq] [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.

<table>
<thead>
<tr>
<th>option</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>INLE</td>
<td>Boundaries specified by the INLET command are selected.</td>
</tr>
<tr>
<td>OUTL</td>
<td>Boundaries specified by the OUTLET command are selected.</td>
</tr>
<tr>
<td>OPEN</td>
<td>Boundaries specified by the OPEN command are selected.</td>
</tr>
<tr>
<td>IO</td>
<td>All boundaries specified by INLET, OUTLET or OPEN command are selected.</td>
</tr>
<tr>
<td>WALL</td>
<td>Walls specified by WALL or BLOCK command are selected.</td>
</tr>
<tr>
<td>EXTE</td>
<td>All external (or outer) boundaries of the computational domain are selected.</td>
</tr>
<tr>
<td>ALL</td>
<td>All of the above boundaries are selected.</td>
</tr>
</tbody>
</table>

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.

Vfrq
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, Vfrq 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 Vfrq 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  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: 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 = ƒ]  [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 "FORMATTED" or "UNFORMATTED" that define the nature of the data in the restart file. If this specification is omitted, the file is assumed to be formatted (see Section 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.

ƒ: 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 SOLVE 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 `RESTART` modifier. If this restart option is used, then it is recommended that the input command file must be identical to the file that generated this particular restart file except for changes in, or presence of, `READ`, `SAVE`, `OUTPUT` and `SOLVE` commands. Any other changes in, say, boundary conditions or geometry may lead to unpredictable results.

**fname**

The name of the file from which the input is obtained. See Sections 6.4 and 7.2.2 for more information. The default file name is the output file name (see `OUTPUT` Command) but with the extension "_.RESTART.TMP" attached to the file name. For example if the standard output file name is "MYCASE" or "MYCASE.OUT", then the default file name will be "MYCASE_RESTART.TMP".

**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, Ta, 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;

\[
p^n = \frac{p - p^*}{p^* g}.
\]

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/m²) 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 (HYDRAULIC 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 \( (p^* g) \) 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/m²) 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 (HYDRAULIC 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, $\rho$ is the density and $V$ is the fluid speed. By default the location of the reference pressure is assumed to be the same as that of the datum (see NODE).

**NORM**

The normalizing pressure is specified. By default the normalizing pressure is unity except for PORFLOW™ it is the pressure head ($p^* g$) 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, $\rho$ is the density and $V$ is the fluid speed. By default the location of the reference pressure is assumed to be the same as that of the datum (see NODE).

N1, Nn
The location of the reference pressure in terms of grid indices for structured grid or element number for unstructured grid.

EXAMPLES

- Reference NODE for P at (I=5, J=10, K=3)
- Reference location for P COEFficient is (I=1, J=5, K=2)
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 CONVERGENCE command) are used at each time step. A more complete discussion of the role of the relaxation parameter is given in standard textbooks (for example, Varga, 1962).

EXAMPLES:
- RELAxation factor for P = 0.7
- RELAxation factors: T = 1.2, C = 0.9
- RELAxation factors: T = 0.7, C = 0.9, C2 = 0.7
- RELAxation factors: P=0.1, P2=0.1, T=0.1, S = 0.5
- RELAxation 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] [Vfrq] [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 SOLVE command is encountered. The default set includes the variables for which equations are solved, and some important supporting variables based on the nature of the problem.

BLOC The output for each variable is written in a block format. One record is written for each variable. The variables are written in the order in which they appear on the command. If the grid is structured then the variable is written in the manner of a FORTRAN DO loop (over the grid indices I, J, K). If the grid is unstructured, then the record for the variable is sequential over element numbers starting with the 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 OUTPUT Command) but with the extension changed to “SAV” for the BLOC and “_TABLE.SAV” for the TABL 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 "FORMATTED" or "UNFORMATTED", which defines the nature of the data in the archive file. By default, the file is formatted.
GEOM By default the archive file contains problem geometry and grid connectivity variables plus variables specified by the user or active default variables (see $\phi$ above). If GEOMETRY modifier is present, then only the problem geometry and grid connectivity information is written to the file; the $\phi$ variables are omitted.

COMP If this modifier is present, then the problem geometry and grid connectivity information is not written to the file; only the $\phi$ variables are written. If both modifiers are specified, then COMPACT modifier takes precedence over the GEOMETRY modifier.

DATA By default standard ACRi file header containing information about the problem and nature of data appears at the start of the file. If DATA modifier is present then the file header information is omitted.

REPL If multiple records are written due to the $V_{frq}$ 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 TABLE mode of the command. It specifies the subregion for which the output is required. If no subregion is specified, then entire computational domain is selected. See 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 SOLVE command is given. It then enables the output of the current value of specified variable(s) immediately. After the 1st SOLVE command, the output is always produced as soon as the command is encountered and this modifier is redundant. The symbol(s) for the output variable(s) must be specified with this modifier since the default output symbols are not available till after the 1st SOLVE command.

NOW If this modifier is specified before the 1st SOLVE 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 SOLVE command. The output is always produced as soon as the command is encountered; this modifier is redundant.

ONLY If this modifier is specified then the output is produced only once in response to a NOW or IMMEDIATE modifier or at the frequency determined by $V_{frq}$.

OFF Output for any previous command(s) for the same subrgn and type (BLOCK or TABLE) 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 a 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] [Vfrq] [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 \((B_i - A_{ij} X_j)\) 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.

Vfrq 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] [Vfrq] [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.

Vfrq 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 or TABLE) and same combination of FLUX, CONV, DIFF and TOTAL 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]
[Vfrq] [TIME] [IMME] [NOW] [ONLY] [OFF]
```

GRAD

The tensor components of the gradients of velocity, \( \phi_{ij} \) are output, where:

\[
\phi_{ij} = \frac{\partial u_i}{\partial x_j}
\]

Here \( u_i \) is the \( i^{th} \) 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, \( \omega \), is obtained, where:

\[
\omega_1 = \frac{\partial u_3}{\partial x_2} - \frac{\partial u_2}{\partial x_3}; \quad \omega_2 = \frac{\partial u_1}{\partial x_3} - \frac{\partial u_3}{\partial x_1}; \quad \omega_3 = \frac{\partial u_2}{\partial x_1} - \frac{\partial u_1}{\partial x_2}
\]

There are 3 components of vorticity for 3D flow but only one for 2D flow.

STRUCTURE

Output of a “vorticity” structure variable, \( \Omega \), is obtained. It is defined as:

\[
\Omega = -\frac{\partial u_1}{\partial x_2} \frac{\partial u_2}{\partial x_1} \text{ for 2D}
\]
\[
= -\frac{\partial u_1}{\partial x_2} \frac{\partial u_2}{\partial x_1} - \frac{\partial u_2}{\partial x_3} \frac{\partial u_3}{\partial x_2} - \frac{\partial u_3}{\partial x_1} \frac{\partial u_1}{\partial x_3} \text{ for 3D}
\]

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, \( \tau_{ij} \), is obtained, where:

\[
\tau_{ij} = \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \delta_{ij} \sum_k \frac{\partial u_k}{\partial x_k} \right) - \delta_{ij} p
\]

Here \( \mu \) is viscosity and \( p \) is the pressure. There are 3 components for 2D and 6 for 3D flow.

LIGHTHILL

Output of the components of “lighthill” stress tensor, \( L_{ij} \), is obtained, where:

\[
L_{ij} = -\mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \delta_{ij} \sum_k \frac{\partial u_k}{\partial x_k} \right) + \rho u_i u_j
\]

Here \( \mu \) is viscosity and \( \rho \) is the density. There are 3 components for 2D and 6 for 3D flow.

BLOC See Mode 1 specification.

TABL See Mode 1 specification.

ADD This modifier is assumed by default. Each command in this mode, unless an OFF modifier is present, is treated as an additional command that adds to the active commands.
fname The file name to which the output is directed. Since the ADD modifier is assumed by default, the filename, if specified, is considered unique for the command and any subsequent SAVE command that specifies the same file name. By default, the output is directed to the same file that is used for Mode 1 output of SAVE commands without the ADD modifier. See Section 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.

Vfrq 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 TABLE) 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 ] [ Vfrq ] [ 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 RESTART modifier to restart the computations for the same identical problem. Only the SOLVE command may be modified for the new simulations; all other input data must be identical to the run that generated the file. RESTART file should not be used to start a different problem or to restart the same problem with different input commands. In these latter cases, the required variables should be explicitly saved by Mode 1 of the SAVE command. If multiple commands are given, and a new file name is not specified, then any previous data on the file is replaced by the new data.

fname
The name of the file to which the output is directed. See Section 7.2.2 for additional information. The default file name is the output file name (see OUTPUT Command) but with the extension "_RESTART.TMP" attached to the file name. For example if the standard output file name is "MYCASE" or "MYCASE.OUT", then the default file name will be "MYCASE_RESTART.TMP".

Vfrq
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 Vfrq is interpreted to be the frequency in terms of number of steps. If this modifier is present, then Vfrq is interpreted to be time interval between successive outputs.

NOW
A restart file is immediately written. If Vfrq modifier is not present, then the file is written only once, otherwise the file is replaced at the specified frequency.

EXAMPLES

SAVE REST  ! acr_RESTART.TMP file will be generated at end of simulations
SAVE RESTart file
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:

\[ Q_{\text{in}} = a_1 \times Q + a_2 , \]

where \( Q_{\text{in}} \) 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 COORDINATE, SET and SOURCE commands.

EXAMPLES  
SCALe multiply by 3.3 and add 10.
SCALe multiply by 0.3048
COMMAND SCREEN

PURPOSE To control the echo of diagnostic output obtained from the DIAGNOSTIC command to the CRT device.

SYNTAX SCREEN [ OFF | ON ] [ WIDE ]

OFF The diagnostic output to the CRT device is turned off.

ON The diagnostic output to the CRT device is turned on. This is the default option. This option may also be used to turn on the CRT output which was previously turned off.

WIDE By default a maximum of 5 selected diagnostic variables (DIAGNOSTIC command) are echoed to the CRT device. If this modifier is present, then the maximum is set to 9.

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  \( \Phi \)  [func \( \xi \)]  [option]  [mod]  [ALWA]  [subrgn]  [FIEL]  [dir]  \( N1 \)  [fname | \( N2 .., Nn \)]  [STAC]

\( \Phi \)  One, and only one, symbol that denotes the variable for which the values are specified. The valid symbols include those listed in Table 6.8.1 plus the user defined variables (ALLOCATE command) and the real type variables defined in the initialization file (xxxINIT.ACR). There is no default value; a symbol must be specified.

func  One of the modifiers listed in Table 7.2.3, which denotes the functional form of the dependent variable. If no function is specified, the value is assumed to be constant.

\( \xi \)  One of the independent variables listed in Table 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.

<table>
<thead>
<tr>
<th>option</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>REPL</td>
<td>The computed value replaces the existing value. This is the default option.</td>
</tr>
<tr>
<td>ADD</td>
<td>The computed value is added to the existing value.</td>
</tr>
<tr>
<td>SUBT</td>
<td>The computed value is subtracted from the existing value.</td>
</tr>
<tr>
<td>MULT</td>
<td>The computed value multiplies the existing value.</td>
</tr>
<tr>
<td>DIVI</td>
<td>The computed value divides the existing value. A value of 1.E-30 is added to the divisor if it is zero.</td>
</tr>
</tbody>
</table>

mod  The modifier for function evaluation.

<table>
<thead>
<tr>
<th>mod</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABS</td>
<td>The absolute value for the computed function is taken.</td>
</tr>
<tr>
<td>ABSO</td>
<td>or</td>
</tr>
<tr>
<td>POSI</td>
<td>The negative values for the computed function are set of zero</td>
</tr>
<tr>
<td>NEGA</td>
<td>The positive values for the computed function are set of zero</td>
</tr>
</tbody>
</table>

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 LOCATE command with the FIELD 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. This modifier is ignored if the LOCATE 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 `BOUNDARY` 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, $\phi$ is computed from the value of $\xi$ at the same location. If this modifier is present, then $\phi$ is computed from $\xi$ at a location previously specified by the `STACK LOCATE` command.

---

**EXAMPLES**

Generic examples for this command are given in Section 7.2.8. The keyword `SET` must replace the keyword used in these examples.

SYNTAX

SET $\{\Phi\} \{\ \text{NODE} \mid \text{STRU} \mid \text{UNST} \mid \text{fname}\} \{V_1,..V_m\} \{\text{subrgn}\} \{\text{FIEL}\} \{\text{dir}\}$

$\Phi$  
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 STRUCTURED mode if the subrgn was defined with a LOCATE command in terms of grid indices or in the UNSTRUCTURED mode otherwise.

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

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

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

UNST  
The input is read in the manner of a simple implied FORTRAN DO loop:

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

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

fname  
The name of the file for numerical values unless the values are directly specified by $V_1,..V_m$ 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 SOLVE OFF command) to make sure that the input has been properly interpreted.

EXAMPLES

SET $T$ from file 'ALLVAL' input for the entire domain of computation
SET $T$ for SELECTed region from 'ACTIVAL' input for active subregion
SET $T$ in ID=RGN1 region 'RGN1VAL' input for identified subregion
SET FC – a supplementary flux variable for the entire field from file: 'VALUES.FC'
SET MTYP – a supplementary material type index (integer) for the entire field from file: 'VALUES.IZ'
MODE 3: Specification of Multiple Variables on a Node-by-Node Basis from a Table

SYNTAX

\[ \text{SET } \{ \Phi_1, \ldots, \Phi_n \} \{ \text{NODE | STRU | UNST | fname} \} \{ V_{1..V_{nm}} \} \{ \text{subrgn} | \text{FIEL} \} \{ \text{dir} \} \]

\( \Phi_1, \ldots, \Phi_n \)
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 \( \Phi_n \) 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 STRUCTURED mode if the \( \text{subrgn} \) was defined with a LOCATE command in terms of grid indices or in the UNSTUCTURED mode otherwise.

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

\[
\text{Read (NUNIT, *) (((VAR(L,I,J,K), L=1,n), I=ILO,IHI), J=JLO,JHI), K=KLO,KHI)}
\]

Where \( \text{NUNIT} \) is an internally assigned unit number, \( \text{VAR} \) is the set of variables denoted by \( \Phi \)'s, \( n \) denotes the number of variables, and \( \text{ILO,IHI, JLO,JHI,KLO,KHI} \) define, respectively, the starting and ending grid index values for the \( \text{subrgn} \) for a structured grid. For 2D grid, \( \text{KLO and KHI are set to unity.} \)

**UNST**
The input is read in the manner of a simple implied FORTRAN DO loop:

\[
\text{Read (NUNIT, *) (VAR(L,M), L=1,n), M = MLO,MHI )}
\]

Where \( \text{NUNIT} \) is an internally assigned unit number, \( \text{VAR} \) is the set of variables denoted by \( \Phi \)'s, \( n \) denotes the number of variables, and \( \text{MLO and MIHI} \) are the starting and ending element numbers for the \( \text{subrgn} \). The order of the elements is the same as the one specified (or implied) by the corresponding LOCATE command which can be examined by including a file name on the LOCATE command.

**fname**
The name of the file (see Section 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 \( \text{subrgn} \) and \( \text{dir} \) 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  
<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>-5</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td>-10</td>
<td>+5</td>
<td>150</td>
<td></td>
</tr>
<tr>
<td>-10</td>
<td>+10</td>
<td>200</td>
<td></td>
</tr>
</tbody>
</table>

**SET** by NODE following variables for ID=SUBRGN1  
<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2</td>
<td>0.2</td>
<td>10</td>
<td>-5</td>
<td>10.20</td>
<td>100</td>
</tr>
<tr>
<td>3.5</td>
<td>0.6</td>
<td>-10</td>
<td>+5</td>
<td>12.50</td>
<td>150</td>
</tr>
<tr>
<td>10</td>
<td>1.0</td>
<td>-10</td>
<td>+10</td>
<td>2.00</td>
<td>200</td>
</tr>
</tbody>
</table>

**SET** by NODE for ID=MIDDLE at X- boundary  
<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>(0.2, 0.2, 0.2)</td>
<td>10</td>
<td>-5</td>
<td>100</td>
</tr>
<tr>
<td>(3.5, 0.6, 1.0)</td>
<td>-10</td>
<td>+5</td>
<td>150</td>
</tr>
<tr>
<td>(10., 1.0, 2.0)</td>
<td>-10</td>
<td>+10</td>
<td>200</td>
</tr>
</tbody>
</table>

**SET** by NODE ID=MIDDLE T, U, V SKIP from file 'SPATIAL'  
<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

! There are 3 nodes in this region)
MODE 4: Specification of a Variable as a Linear Sum of other Variables

SYNTAX

SET \{ \Phi \} \{ \text{SUM | LINE } \} \{ \text{MASS} \} \{ \xi_1, \ldots, \xi_n \} \{ a_1, \ldots, a_n \} \{ a_0 \} \{ \text{option} \} \{ \text{mod} \} \{ \text{ALWA} \} \{ \text{subrgn} \} \{ \text{FIEL} \}

\Phi

See Mode 1 specification.

SUM

The variable \( \Phi \) is computed from:

\[ \Phi = \sum_n \xi_n \]

LINE

The variable \( \Phi \) is computed from:

\[ \Phi = \sum_n a_n \xi_n + a_0 \]

MASS

By default, the \( \xi \)'s (see definition below) on the right hand side of the defining equation are assumed to be the field variables. If this modifier is present, then \( \xi \)'s are replaced by the total mass of the corresponding \( \xi \) variable. The total mass is equal to the quantity that appears in the accumulation term of the governing differential equation for \( \xi \) (see Chapter 2) multiplied by the volume of the element. In the presence of this modifier only those \( \xi \)'s can appear on the right hand side for which differential equations are solved.

\( \xi_1, \ldots, \xi_n \)

Symbols to denote the variables, or their masses, which contribute to the linear sum as in the above equation. The valid symbols include those listed in Table 6.8.1 plus the variables defined by the user (ALLOCATE command) or the real type variables defined in the initialization file (xxxINIT.ACR). There is no default value; appropriate symbols must be specified. No more than 10 symbols can be specified.

\( a_1, \ldots, a_n \)

The numerical constants and coefficients for the linear function. These values must be specified if the LINEAR modifier is present. With the SUM modifier these values must not be specified. There is not default value; appropriate number of values must be specified.

\( a_0 \)

The datum for the linear function if LINEAR modifier is specified. The default value is 0.

option

See Mode 1 specification.

mod

See Mode 1 specification.

ALWA

See Mode 1 specification.

subrgn

See Mode 1 specification.

FIEL

See Mode 1 specification.

EXAMPLES

SET \( T \) as SUM of \( X + Y + U \)
SET \( T \) as SUM of \( X + Y + U \) ALWAYS for \( \text{ID} = \text{T1DOMAIN} \)
SET \( T \) as LINEAR function \( 1. \ast X + 2. \ast Y - 0.5 \ast U + 5. \) ALWAYS for \( \text{ID} = \text{T1DOMAIN} \)
**MODE 5:** Specification of a Variable as a Square or Square Root Sum of other Variables

**SYNTAX**

```
SET {Φ} {SQUA | ROOT} {ξ₁, ..., ξₙ} [a₁, ..., aₙ] [a₀] [option] [mod] [ALWA] [subrgn] [FIEL]
```

Φ

See Mode 1 specification.

SQUA

The variable Φ is computed from:

\[ Φ = \sum_{n} n^{2} \]  or  \[ Φ = \sum_{n} a_{n} ξ_{n}^{2} + a_{0} \]

ROOT

The variable Φ is computed from:

\[ Φ = \sqrt{\sum_{n} n^{2}} \]  or  \[ Φ = \sqrt{\sum_{n} a_{n} ξ_{n}^{2} + a_{0}} \]

ξ₁, ..., ξₙ

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 (ALLOCATE command) or the real type variables defined in the initialization file (xxxINIT.ACR). There is no default value; appropriate symbols must be specified. No more than 10 symbols can be specified.

a₁, ..., aₙ

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.

a₀

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] {Φ₁, ..., Φₙ} {Nsets} {V₁...Vₘ | fname} [option] [mod] [ALWA] [subrgn] [FIEL]

DIST

The Φᵱ variables are computed from a inverse distance based interpolation function (here ξᵢ is iᵗʰ value in the specified table for the corresponding variable and xᵢ,k are its coordinates):

\[ \Phi = \sum_i \xi_i / r_i \quad \text{or} \quad \Phi = \sum_i \xi_i / r_i^2 \quad ; \quad r_i^2 = \sum_k (x_k - x_{k,i})^2 \]

SQUA

The inverse distance squared interpolation is used. This is the default option.

LINE

The inverse distance based interpolation is used.

Φ₁, ..., Φₙ

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.

V₁…Vₘ

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 Φ₁, through Φₙ. The number of rows of the table must equal Nsets.

fname

The name of the file (see Section 7.2.2) from which V₁ through Vₙ 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)

\[
\begin{array}{ccc}
(X, \ Y) & U & V & T \\
(0.2, 0.2) & 10 & -5 & 100 \\
(3.5, 0.6) & -10 & +5 & 150 \\
(10, 1.0) & -10 & +10 & 200 \\
\end{array}
\]

SET by LINEAR DISTANCE interpolation for ID=MIDDLE from values at 3 stations (Three-dimensional)

\[
\begin{array}{ccc}
(X, \ Y, \ Z) & U & V & T \\
(0.2, 0.2, 0.2) & 10 & -5 & 100 \\
(3.5, 0.6, 1.0) & -10 & +5 & 150 \\
(10., 1.0, 2.0) & -10 & +10 & 200 \\
\end{array}
\]

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:

\[
Q = \sum_{\text{subrgn}} \alpha_{i} \Phi_{i} \delta V_{i}
\]

Here Q the subscript “i” denotes the values for an element, α is the accumulation coefficient for the transport equation, and δV is the volume of the element. The summation is taken over all elements of the subregion.

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

**Q**
The specified inventory (> 0) of the property, Φ:

**VOLU**
The inventory is distributed so that the amount added to each element is proportional to the volume of the element. This is the default option. The new value is computed as:

\[
\Phi_{i} = \theta \Phi_{i}^{o} + \frac{1}{\sum_{\text{subrgn}} \alpha_{i} \delta V_{i}} \frac{Q}{\alpha_{i} \delta V_{i}}
\]

Here $\Phi^{o}$ is the existing value of variable, and $\theta = 0$ 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:

\[
\Phi_{i} = \theta \Phi_{i}^{o} + \frac{Q}{\sum_{\text{subrgn}} \alpha_{i} \delta V_{i}}
\]

**SCAL**
The inventory is distributed proportional to the existing value of Φ:

\[
\Phi_{i} = \theta \Phi_{i}^{o} + \frac{Q \Phi_{i}^{o}}{\sum_{\text{subrgn}} \alpha_{i} \Phi_{i}^{o} \delta V_{i}}
\]

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 $\theta = 0$ in the above equation. If this modifier is present, then the inventory is added to the existing value; that is $\theta = 1$. 
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 UNIFomrly 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

\[
\text{SET } \{ \Phi \} \{ \text{INVE} \} \{ \text{ID=subrgn1} \} \{ \text{ID=subrgn2} \} \{ \text{VOLU} \} \{ \text{UNIF} \} \{ \text{SCAL} \} \{ \beta \} \{ \text{FRAC} \} \{ \text{TIME} \} \{ \text{ALWA} \} \{ \text{OFF} \}
\]

\(\Phi\)
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 equation is solved. There is no default value; a symbol must be specified.

\(\text{INVE}\)
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.

\(\text{ID=subrgn1}\)
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:

\[
Q = \lambda \sum_{i=1}^{\text{subrgn1}} \alpha_i \Phi_i^o \delta V_j
\]

Here superscript 'o' denotes the existing value of the variable \(\Phi\), subscript “i” denotes the value for an element, \(\alpha\) is the accumulation coefficient for the transport equation, \(\delta V\) is the volume of the element, and \(\lambda\) is a constant. The summation is taken over all elements of the donor subregion. The constant, \(\lambda\), is determined as follows:

\[
\lambda = 1 \text{ by default if } \beta \text{ is not specified.}
\]
\[
\lambda = \min(\beta, 1) \text{ if } \beta \text{ is specified but } \text{TIME} \text{ is not specified}
\]
\[
\lambda = \min(\beta \delta t, 1) \text{ where } \delta t \text{ is the time step if } \beta \text{ and } \text{TIME} \text{ are specified}
\]

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:

\[
\Phi_i = (1-\lambda) \Phi_i^o
\]

\(\text{ID=subrgn2}\)
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.

\(\text{VOLU}\)
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:

\[
\Phi_i = \Phi_i^o + \frac{1}{\alpha_i} \sum_{\text{subrgn2}}^Q \frac{Q}{\delta V_j}
\]
UNIF  The inventory is distributed so that the increment in $\Phi$ is uniform and constant:

$$\Phi_i = \Phi_i^0 + \frac{Q}{\sum_{\text{subrgn2}} \alpha_i \delta V_i}$$

(Receptor)

SCAL  The inventory is distributed proportional to the existing value of $\Phi$:

$$\Phi_i = \Phi_i^0 + \frac{Q \Phi_i^0}{\sum_{\text{subrgn2}} \alpha_i \Phi_i^0 \delta V_i}$$

(Receptor)

Since the old value of $\Phi$ appears in the denominator, this modifier should be used only if the variable is positive definite; that is all values are greater than zero.

$\beta$  A constant ($\beta \geq 0$) that determines how much of the current inventory is transferred from the donor to the receptor region. The specified value must not be negative. If no value is specified, then the total amount is transferred.

FRAC  $\beta$ specifies the fraction of the current inventory in the donor region that is transferred. Limits are imposed so that ($0 \leq \beta \leq 1$); that is the minimum transferred amount is zero and the maximum is equal to the total amount in the donor region at any time. This is the default interpretation of $\beta$.

TIME  $\beta$ specifies the fraction per unit time of the current inventory that is transferred. The actual fraction transferred is equal to $\beta \delta t$ where $\delta t$ is the time step. Limits are imposed dynamically so that ($0 \leq \beta \delta t \leq 1$).

ALWA  By default, the specified amount is transferred only once, that is, the first time the command is encountered. If the TIME modifier is present then the specified amount is transferred at each step starting with the first time the command is encountered.

OFF  If this modifier is present then any previous SET INVE command that was specified by the ALWAys modifier for the same $\Phi$, 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:

\[
Ψ = \frac{\partial Φ}{\partial ξ}
\]

or

\[
Ψ = \text{function}\left(\frac{\partial Φ}{\partial ξ}\right)
\]

The function, if any, is defined by the *fnc* modifier as given below.

**ψ**

The symbol for the variable where the computed gradient values are stored. This must be a valid symbol for one of the default field variables of ACRi Software or it must be allocated through the `ALLOCATE` command.

**Φ**

Symbol for one of the previously defined field variables listed in Table 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

<table>
<thead>
<tr>
<th>fnc</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABS or ABSO</td>
<td>The absolute value for the computed gradient is taken.</td>
</tr>
<tr>
<td>POSI</td>
<td>The negative values for the computed gradient are set to zero</td>
</tr>
<tr>
<td>NEGA</td>
<td>The positive values for the computed gradient are set to zero</td>
</tr>
<tr>
<td>SQUA</td>
<td>The function is computed as the square of the gradient</td>
</tr>
<tr>
<td>ROOT</td>
<td>The function is computed as the square root of the gradient; negative values are set to 0.</td>
</tr>
</tbody>
</table>

**BLOC**

If this modifier is present then the value of the gradient at the blocked nodes (see `BLOCK` command) is set to zero. Otherwise the computed value is retained.

**FIEL**

By default the gradient is computed throughout the computational domain including the field and the boundary nodes. If this modifier is present, then only the field values are computed.

**ALWA**

By default the `SET` command is implemented *immediately and only once* – as soon as the command is encountered. If this modifier is present then the command is implemented immediately as well as repeatedly at the beginning of every time step (or iterative step in steady state mode) of the solution procedure.

**OFF**

Any previous `SET` command with `GRAD` modifier for the specified variable is deactivated.

**EXAMPLES**

```plaintext
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 } { Φ₁, …, Φₙ } { V₁,...Vₙ+1m | fname} [ subrgn ] [FIEL] { dir | BOUN }

VERT

The Φₙ 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.

Φ₁, …, Φₙ

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.

V₁,…Vₙm

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 V₁ through Vₙ+1m 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

<table>
<thead>
<tr>
<th>#</th>
<th>SKIP</th>
<th>P</th>
<th>T</th>
<th>Φ₁</th>
<th>Φ₂</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.2</td>
<td>1.0</td>
<td>-5</td>
<td>10.20</td>
<td>100</td>
</tr>
<tr>
<td>2</td>
<td>0.6</td>
<td>0.0</td>
<td>+5</td>
<td>12.50</td>
<td>150</td>
</tr>
<tr>
<td>3</td>
<td>1.0</td>
<td>-1.0</td>
<td>+10</td>
<td>2.00</td>
<td>200</td>
</tr>
<tr>
<td>4</td>
<td>0.2</td>
<td>1.0</td>
<td>-5</td>
<td>10.20</td>
<td>100</td>
</tr>
<tr>
<td>5</td>
<td>0.6</td>
<td>0.0</td>
<td>+5</td>
<td>12.50</td>
<td>150</td>
</tr>
<tr>
<td>6</td>
<td>1.0</td>
<td>-1.0</td>
<td>+10</td>
<td>2.00</td>
<td>200</td>
</tr>
<tr>
<td>7</td>
<td>0.2</td>
<td>1.0</td>
<td>-5</td>
<td>10.20</td>
<td>100</td>
</tr>
<tr>
<td>8</td>
<td>0.6</td>
<td>0.0</td>
<td>+5</td>
<td>12.50</td>
<td>150</td>
</tr>
<tr>
<td>9</td>
<td>1.0</td>
<td>-1.0</td>
<td>+10</td>
<td>2.00</td>
<td>200</td>
</tr>
<tr>
<td>10</td>
<td>0.2</td>
<td>1.0</td>
<td>-5</td>
<td>10.20</td>
<td>100</td>
</tr>
<tr>
<td>11</td>
<td>0.6</td>
<td>0.0</td>
<td>+5</td>
<td>12.50</td>
<td>150</td>
</tr>
<tr>
<td>12</td>
<td>1.0</td>
<td>-1.0</td>
<td>+10</td>
<td>2.00</td>
<td>200</td>
</tr>
</tbody>
</table>

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

\[
\text{SET} \quad \{\Phi\} \quad \{\text{OFF}\} \quad [\text{subrgn}]
\]

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, \(\Phi\).

EXAMPLES

- SET commands for T for OFF for currently SELECTed subregion
- SET commands OFF for T for ID=MI DDle
KEYWORD COMMANDS

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 SOLVE 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 1.6E6 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  \[^{\text{!Previously an AUTO mode was used.}}\]
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 CONVERGENCE command with FLOW modifier). The time step is increased if the number of iterations to converge is less than 1/2 of the maximum number of iterations specified on the CONVERGENCE command with FLOW modifier and decreased if it is larger than 4/5th 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 CONVERGENCE command. The default value is 2.

N7

The maximum number of time steps. The default value is 99,999,999.

COMMENTS

The AUTOmatic solution mode is currently available only with the PORFLOW™ Software Tool. It works quite efficiently for single phase (saturated mode) flow problems. However, with multiphase problems, this mode may lead to excessive computation time unless the residual convergence criterion 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

\[ \text{SOLV } \Phi \{ \text{CFL} \} \{ N1, N2, N3, N4, N5, N6, N7, N8 \} \]

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:

\[ R = \max \left[ \frac{\max |A_{ij}|}{A_{jj}} \right] \leq R_{\text{max}} \]

over all \( i \)

where \( A_{ij} \) are the coefficients of the solution matrix: \( A_{ij} X_i = B_j \)

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, \( R_{\text{max}} \), for the CFL parameter. If the computed value of \( R \) exceeds the threshold value then new time step is computed from:

\[ \frac{dt_{\text{new}}}{dt_{\text{fac}}} = \frac{dt_{\text{old}}}{dt_{\text{fac}}} ; \quad dt_{\text{fac}} = \max \left( \frac{R}{N6}, N7 \right) \]

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\}  \[\Phi\]  [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.

\(\Phi\)  
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 SOLVE command immediately initiate solution of the governing differential equations. Therefore, these commands should be specified only after input has been supplied that is complete to initiate solution. However, the sequence of calculations may be sub-divided into as many segments as desired and a SOLVE command specified for each segment. See Section 6.10 for more information.

EXAMPLES

SOLVe OFF
SOLVe OFF now to obtain initial output for problem verification.
**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], [ρB]
```

- **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 VOLUME 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.

<table>
<thead>
<tr>
<th>option</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>VOLU</td>
<td>In the absence of the TOTAL modifier, the source for each element is computed as: ( Q = q \delta V ). Here ( q ) is the amount specified by the user and ( \delta V ) is the volume of the element. The ( q ), in turn, is computed from ( func(ξ) ) and ( N1 ) through ( Nn ). If the TOTAL modifier is present, the amount for each element is computed as: ( Q = q \delta V / V ), where ( V ) is the volume of the total subrgn.</td>
</tr>
<tr>
<td>AREA</td>
<td>In the absence of the TOTAL modifier, the source for each element is computed as: ( Q = q \delta A ), where ( \delta A ) is the area of the element boundary indicated by <code>dir</code>. If the TOTAL modifier is present, the source for each element is computed as: ( Q = q \delta A / A ), where ( A ) is the total area of the subrgn in the dir direction.</td>
</tr>
<tr>
<td>INTE</td>
<td>By default, if <code>dir</code> 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 INTERNAL modifier is present then the wall treatment and wall diffusive flux are retained.</td>
</tr>
</tbody>
</table>
| NORM   | In the absence of the TOTAL modifier, the source, \( Q_i \), is computed as:  
\[
Q = q \sum A_i \cdot V_i
\]
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 TOTAL 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, \( ρB \), 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 NORMAL 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 **NORMAL** 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, Vi, if the **NORMAL** modifier is present. Two values must be specified for 2D and 3 for 3D flows. There are no default values for this input.

**ρ**  
The density value that multiplies the computed source. It can only be specified if the **DENSITY** modifier is present and then it must be the last value on the command. If no value is specified but the modifier **DENSITY** is present, then default value is the boundary value at the node indicated by the **dir** direction if the **AREA** or **NORMAL** modifier is present, otherwise the default value is the local density for the element.

**EXAMPLES**

Generic examples for this command are given in Section 7.2.8. The command keyword (**SOURCE**) must replace the keyword used in these examples. Some additional examples that illustrate the use of the attributes specific to this command are given below.

**SOURCE** for T: = 10 W/per unit by VOLUme for SELECTed region  
**SOURCE** T is TABLE per unit AREA in X- direction: 3 sets (TIME, value) (0., 0.01), (100., 0.10 ), (200, -0.20)  
**SOURCE** for T 10 per unit area in X- direction for SELECTed region  
**SOURCE** for T 10 per unit area in X- direction with NORMALized of 0., 1.5, 2.5 ID=VSOURce  
**SOURCE** for T 10 in X- dir with NORMALized vel 1., 1.5, 2.5 and DENSity for ID=VSOURce  
**SOURCE** for T 10 X- dir INTERNAL with NORMALized vel 1., 1.5, 2.5 and DENSity = 5 for ID=VSOURce  
**SOURCE** for T 10 in per unit VOLUme multiply by DENSity = 5 for SELECTed region
MODE 2: Flow Injection with Fixed Variable Values

SYNTAX SOUR {FLOW} [TOTA] [ func [ξ ] ] [option] [subrgn] [dir] {N1} [fname | N2, ..,Nn] [Φ=Nn+1, .., Φ=Nm] [Nm+1, Nk] [ρB]

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, Vi, if the NORMAL modifier is present. Two values must be specified for 2D and 3 for 3D flows. There are no default values for this input.

ρB See Mode 1 specification.

COMMENTS

The SOURCE FLOW command is designed to allow injection of fluid through a solid boundary (or blocked region) inside the domain of computation. Its use is most appropriate when the scale of injection is smaller than the grid size. If the scale of injection is comparable to the grid size, then the FLOW command may be more appropriate. The injection at an exterior domain boundary is often better accounted for by the INLET command.

In implementing this command it is assumed that at the point of injection, the boundary wall is essentially removed. The computational treatment is akin to that for inflow such as through an INLET boundary. If the injection occurs through a blocked element or at an exterior boundary of the domain, then the injected values are also assumed to be the boundary values of the variables. If the wall needs to be retained, such as for flow percolating through a solid matrix or through holes much smaller than the boundary area of the inflow element, then the INTERNAL modifier should be used with the command.
KEYWORD COMMANDS

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**: amount = 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] [ρB]

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, \( \rho B \) is the density, \( A \) is the area of the element boundary specified by the \( \text{dir} \) modifier, and \( n_j \) is a normalizing vector. If the \( \text{NORMal} \) modifier is present, then \( n_j \) is obtained from the user input, otherwise the area unit vector (A_j/A; where A_j is the component in the \( j^{th} \) 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 \( \text{NORMal} \) modifier is present, it is assumed that the \( \text{AREA} \) modifier is in effect. Any specification of the \( \text{VOLu} \)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 \( \text{NORMal} \) modifier is present. Two values must be specified for 2D and 3 for 3D flows. There are no default values for this input.
ρB The density value to compute the velocity components. It can only be specified if the \( \text{DENSity} \) modifier is present and then it must be the last value on the command. If no value is specified, then boundary value at the node indicated by the \( \text{dir} \) is used.

COMMENTS

The \( \text{SOURCE MOMETUM} \) command is similar to the \( \text{SOURCE FLOW} \) command except that the momentum (or velocity) of the incoming flow is computed on the assumption that the direction of injected mass is normal to the boundary and that the mass is uniformly distributed at the boundary. All other details of its implementation are identical to those for the \( \text{SOURCE FLOW} \) command.
All the examples cited for Mode 2, except those with VOLUme modifier, are applicable provided that the modifier MOMEntum is added. Some illustrative examples specific to this mode are given below.

**SOURce with MOMEntum q=10 X- direction with ID=VSOURce; T=100, K=0.05, L=0.02**

**SOURce MOMEntum q=10 X- direction with ID=VSOURce; T=100, K=0.05, L=0.02, DENSity 5**

**SOURce MOMEntum q=10 X- direction with ID=VSOURce; T=100, K=0.05, L=0.02, NORMal 1, –1, 0**

**SOURce MOMEntum q=10 X- ID=VSOURce; T=100, K=0.05, L=0.02, NORMal 1, –1, 0, DENSity=5.**

**SOURce MOMEntum: TABLe 2 sets SELECT (0, 0), (100, 1) T=10, K=0.001, L=1**

**SOURce MOMEntum EXPOnential SERIes TIME 7 sets 'SOURCE' T=100, C=0. NORMal 1, –1 SELECTe**

**SOURce MOMEntum q=10 X- dir INTErnal for ID=VSOURce: injected variables: T=100, K=0.03, L=0.5 NORMalized vel 1., 1.5, -0.7 and DENSity as exists**
MODE 4: Flow Injection at Fixed Spherical Angles with Computed Momentum Components

SYNTAX

SOUR \{ ANGL \} [ func ] [ TOTA ] [ subrgn ] \{ dir \} \{ N1 \} [ fname | N2,..,Nn ]
[ Φ=Nn+1,..,Φ=Nm ] [ Nm+1, ..., Nm+5 ] [ DENS | SPEE ] [ ρB | VS ]

ANGL

Fluid is injected or withdrawn. The amount of the property of the injected (or withdrawn) fluid acts as the source or sink for each of the relevant properties. If the flow is injected, then the velocity component of the injected flow, \( U_i \) in the \( i^{th} \) direction, is either computed from the source flow rate, \( Q \), or determined from a specified injection speed, \( V_S \), as:

\[
U_i = \frac{Q}{\rho_B A} n_i \quad \text{or} \quad U_i = V_S n_i
\]

where \( \rho_B \) is the density at the neighboring element and \( A \) is the area of the element boundary specified by the \( \text{dir} \) 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, \( \Theta \) in degrees measured as the deflection of the injection vector from the face normal. The magnitude of the angle must be less than 180 degrees.

Nm+2

The spherical angle, \( \Phi \) in degrees measured as the rotation of the injection vector with respect to the projection of the axis vector onto the face plane.

Nm+3, Nm+5

The direction cosines of the axis vector used to measure the angle \( \Phi \); 3 values must be specified since this option is only available for 3D flows.

DENS

The density, \( \rho_B \), is specified as the last value on the command.

ρB

See Mode 3 specification.

SPEE

The injection speed, \( V_S \), is specified as the last value on the command.

VS

The value \( V_S \) if the SPEE 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.

SOUR ce q=10., X- direction with ID=VSOURce; T=100, K=0.05, L=0.02, Theta=45, phi=20, axis=1., 0, 0
SOUR ce q=10., X- ID=VSOURce; T=100, K=0.05, L=0.02, Theta=45, phi=20, axis=1., 0, 0, SPEEd=120
SOUR ce q=10., X- ID=VSOURce; T=100, K=0.05, L=0.02, Theta=45, phi=20, axis=1., 0, 0, DENSity=2.5
MODE 5: Flow Injection with Multiple Tabular Functions

SYNTAX

SOUR \{FLOW\} {\{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

<table>
<thead>
<tr>
<th>Time (s)</th>
<th>Flow (m/s)</th>
<th>U (m/s)</th>
<th>V (m/s)</th>
<th>T (K)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.001</td>
<td>1.00</td>
<td>0.02</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td>0.002</td>
<td>0.50</td>
<td>0.01</td>
<td>200</td>
<td></td>
</tr>
<tr>
<td>0.004</td>
<td>2.00</td>
<td>0.01</td>
<td>500</td>
<td></td>
</tr>
<tr>
<td>0.010</td>
<td>5.00</td>
<td>0.02</td>
<td>600</td>
<td></td>
</tr>
</tbody>
</table>

SOURce with FLOW MULTiple TABLe: 4 sets per unit VOLUME

<table>
<thead>
<tr>
<th>Time (s)</th>
<th>Flow (m/s)</th>
<th>U (m/s)</th>
<th>V (m/s)</th>
<th>T (K)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.001</td>
<td>1.00</td>
<td>0.02</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td>0.002</td>
<td>0.50</td>
<td>0.01</td>
<td>200</td>
<td></td>
</tr>
<tr>
<td>0.004</td>
<td>2.00</td>
<td>0.01</td>
<td>500</td>
<td></td>
</tr>
<tr>
<td>0.010</td>
<td>5.00</td>
<td>0.02</td>
<td>600</td>
<td></td>
</tr>
</tbody>
</table>
MODE 6: Solubility-Limited Source for a Chemical Species

SYNTAX SOUR \( \{ \Phi \} \) [SOLU \( \{ [\text{func} \{ \xi \}] \} \) [subrgn] [fname] \{N1, ..., Nn\} \{Nn+1\} \{Nn+2\}

\( \Phi \) A symbol that denotes the dependent variable for one of the chemical species. A symbol must be specified.

SOLU The source is solubility-limited. That is, the source is specified as the total initial mass of a chemical species and the solubility of the species in the fluid phase is limited to a maximum saturation value that is defined by Equation 3.10.3.

func The function that specifies the solubility limit, \( C_s \), for the species. See Mode 1 specification.

\( \xi \) 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: \( C_s=0.05 \), \( S=100 \) kg, \( t=0 \) for selected zone.

SOURce for FU: SOLUbility: 75 sets file 'SOURCE.DAT', \( S=75 \), \( t_{\text{start}} = 20 \).
MODE 7: Radiation Type of Source For a Variable

SYNTAX SOUR {Φ} {RADI} [COEF] [VARI] {H, Φeq} [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:

\[ Q = H \left( \Phi_{eq} - \Phi \right) \]

In the above equation, Q is the source for Φ in appropriate units, H is a transfer coefficient and Φeq is an equilibrium value of Φ. The H and Φeq 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.

Φeq 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. NORMal and DENSity 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 HCOE & 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, \( \lambda \) is the decay rate and \( \Psi \) 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, \( \sigma \):
       \[ \lambda = -\frac{\log(0.5)}{\sigma} \]

subrgn  See Mode 1 specification.

N1  The decay rate, \( \lambda \), or half-life, \( \lambda \), 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 REACTION 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} {VALU|DIFF|GRAD} {AREA | NORM | VOLU} {dir} [subrgn] [func [ξ]] {N1} [fname | N2, ..,Nn] [Nn+1, Nk] [DENS] [ρB]

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 \( F_\phi \), \( F_V \) and \( F_\rho \) 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 \( \phi_D \) is the value of \( \phi \) in the donor cell (that lies upstream with respect to direction of the computed Q). The source acts essentially as a convective flux across the interface between the donor and the receptor cells. This type of source can be used to implement processes such as settling of particulate material or droplets due to body forces. This is the default option.

DIFF \( F_\phi = (\phi_B - \phi_F) \), where \( \phi_B \) and \( \phi_F \) are the values of \( \phi \) in the “boundary” and “field” cells, respectively. The field cell is that defined by the 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 \( F_\rho \) is equal to \( \rho_B \). In the absence of this modifier, \( F_\rho \) is equal to unity.
\( \rho_b \)  

The density value that multiplies the computed source. It can only be specified if the \texttt{DENSITY} modifier is present and then it must be the last value on the command. If no value is specified but the modifier \texttt{DENSITY} is present, then default value is the boundary value at the node indicated by the \texttt{dir} direction if the \texttt{AREA} or \texttt{NORMAL} modifier is present, otherwise the default value is the local density for the element.

**EXAMPLES**

All examples cited for Mode 1 are applicable provided that the modifier \texttt{FLUX} is added. Some illustrative examples of the use of attributes specific to this mode are given below.

\begin{itemize}
  \item \texttt{SOURCE FLUX} type for T: \( q=0.1 \) for X- direction of currently active subregion.
  \item \texttt{SOURCE FLUX} type for C1: \( q=1.5 \) multiply by AREA of X- direction interface for ID=RGN1.
  \item \texttt{SOURCE FLUX} for C2: \( q=1.5 \) NORMALized velocities 0., 0., 0.25 for X- direction of ID=RGN1.
  \item \texttt{SOURCE FLUX} C2: \( q=1.5 \) NORM 0.12., 0., 0.25 multiply by DENSity; X- direction ID=RGN1.
  \item \texttt{SOURCE FLUX} C2: DIFFerence. \( q=1.5 \) NORM 0.12., 0., 0.25 DENSity = 5; X- ID=RGN1.
  \item \texttt{SOURCE FLUX} C2: GRADient. \( q=1.5 \) VOLUmetric DENS; X- ID=RGN1.
\end{itemize}
MODE 11: Disable Previously Specified SOURce commands

SYNTAX SOUR {OFF} {Φ} {subrgn}

Φ See Mode 1 specification.

OFF Previous SOURCE commands for the identified subregion are deactivated for the specified variable. A new specification may follow.

subrgn See Mode 1 specification.

EXAMPLES

SOURce OFF for T for most recently 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 ALWAYS 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 SPECIFIC 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 (SPECIFIC) 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


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] {N1, ..,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 \( x_i = T_{ij} X_j \). The index j varies the fastest. Each row of the transformation matrix is interpreted as a unit vector directed along the \( x_i \) 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 \( x_1 \) to be the unit normal to the plane, \( x_2 \) is computed by intersecting the plane with the bounding box of the domain, and \( x_3 \) is computed as the cross product of \( x_4 \) and \( x_2 \). The user has no control over the directions of \( x_1 \) and \( x_2 \). 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 2\textsuperscript{nd} point and the normal points from the 1\textsuperscript{st} to the 2\textsuperscript{nd} point. The transformation is computed in the same way as the previous option.

N1,..,Nn
The 1\textsuperscript{st} through 3\textsuperscript{rd} components of the transformation matrix if the MATRIX 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 PLANE modifier is present and the coordinates of the 1\textsuperscript{st} point if the POINT modifier is present. There is no default value.

Nn+1,..,Nm
The 4\textsuperscript{th} through 6\textsuperscript{th} components of the transformation matrix if the MATRIX modifier is present, a vector (or direction cosines) to specify the normal to the plane of transformation if the PLANE modifier is present and the coordinates of the 2\textsuperscript{nd} point (on the plane of transformation) if the POINT modifier is present. There is no default value.

N7,..,N9
The 7\textsuperscript{th} through 9\textsuperscript{th} components of the transformation matrix if the MATRIX modifier is present; otherwise this input is ignored. There is no default value.

EXAMPLES

STACk TRANSformation between local & global grid is:

\[
1 \quad 0 \quad 0 \\
0 \quad 0 \quad -1 \\
0 \quad 1 \quad 0
\]

STACk TRANSformation between local & global grid is:

\[
1.0 \quad 0.0 \quad 0.0 \\
0.0 \quad 0.9950040 \quad -0.0998334 \\
0.0 \quad 0.0998334 \quad 0.9950040
\]

STACk TRANSformation for PLANE at (0.12, 0.55) normal vector as: (1, -1)

STACk TRANSformation for PLANE at (0.12, 0.55, 0.012) normal vector as: (1, -1, 0)

STACk TRANSformation between POINTS (0.12, 0.55, 0.012) (0.15, 0.47, 0.012)
MODE 3:  Store Scaling Factors to Normalize Output of Variables

SYNTAX  STAC { WRIT } {Φ | OFF } { a } [ b ]

WRIT  The output produced by the WRITE command for the specified variable is scaled by a and b according to:

\[ Φ_{out} = a \ Φ + b \]

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 (ALLOCATE command) and the real type variables defined in the initialization file (xxxINIT.ACR). There is no default value; a symbol must be specified.

OFF  Any previous STACK WRITE commands are deactivated; new commands may follow.

a  The multiplier factor for the variable. There is no default value.

b  The additive constant for scaling the variable. The default value is 0.

COMMENTS  Up to 10 STACK WRITE commands each with its own Φ may be active at any given time.

EXAMPLES  STACk WRITE T by scaling with 1.8 and add 32 to convert deg Celsius to Fahrenheit.
STACk WRITE P by scaling with 1.00E5 and add 0.
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] [Vfrq] [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.

Vfrq         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, Vfrq is interpreted to be the frequency of computations in terms of number
             of steps. If this modifier is present, then Vfrq 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

STATististics for U for the entire domain
STATistics for T for SELEceted region every 20 steps
STATistics for T for SELEceted 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 SELEceted region at TIME interval of 0.4
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  TIDE

PURPOSE  To specify the height of the tide at an open boundary as a function of time and space.

MODE 1:  Tide as a Cosine or Sine Function

SYNTAX  TIDE  [dir]  { COSI | SINE }  {N}  {τn, an, Φn}  {τj, aj, Φj}

dir  One of the character strings:  EAST, WEST, NORTH, or SOUTH. It denotes the boundary of the domain of interest where the water elevation is specified as a function of time and space. If no direction is specified, then the tidal boundary is assumed to be at the west.

COSI  The tide is taken to be a cosine function as given by:

\[ \eta = \sum_{n=1}^{N} a_n \cos \left( \frac{2\pi}{\tau_n} t + \phi_n \right) \]

SINE  The tide is taken to be a sine function analogous to the sine function given by:

\[ \eta = \sum_{n=1}^{N} a_n \sin \left( \frac{2\pi}{\tau_n} t + \phi_n \right) \]

N  Number of sets of harmonic tidal components that follow. The default value is 0.

τn  The time period of the \( n^{th} \) component of the tide. The default value is 0.

an  The amplitude of the \( n^{th} \) component of the tide. The default value is 0.

Φn  The phase of the \( n^{th} \) component of the tide. The default value is 0.

τj, aj, Φj  Triplets of period, amplitude, phase lag for the 2\(^{nd}\) through \( N^{th} \) component of the tide.

COMMENTS  This command is implemented only for the TIDAL™ Software Tool.

EXAMPLES  
TIDE is COSine function, 1 set: period=24 hrs, amplitude=30 cm, phase=12 hrs  
TIDE SOUTH is the sum of SINE functions, 2 sets: (24,30,0) (676,10,0)
MODE 2: Tide specified as a table of Time and Height

SYNTAX  TIDE [dir] {N1} {N2, N3} [N4, ..., Nn]

dir  One of the character strings: EAST, WEST, NORTH, or SOUTH. It denotes the boundary of the domain of interest where the water elevation is specified as a function of time and space. If no direction is specified, then the tidal boundary is assumed to be at the west.

N1  Number of sets of tide values which follow. The default value is 0.

N2  The time at which the first tide height is specified. The default value is 0.

N3  The tide height at time N2. The default value is 0.

N4, ..., Nn  Pairs of time and tide height for one complete tidal cycle

EXAMPLES

TIDE WEST: (time, height) table, 5 sets: (0, 10) (3, 15) (9, 20) (12, 25) (24, 10)
MODE 3: Tide as a Linear Function of Distance and Time

SYNTAX

TIDE { LINE } [dir] {N1} {N2, N3, N4} [N5, ..., Nn] { Nn+1, Nn+2 }

LINE

The tidal boundary condition at the specified boundary is taken to be a linear function of both time and distance along the boundary.

dir

One of the character strings: EAST, WEST, NORTH, or SOUTH. It denotes the boundary of the domain of interest where the water elevation is specified as a function of time and space. If no direction is specified, then the tidal boundary is assumed to be at the west.

N1

Number of sets of tide values which follow. There is no default value; a value must be specified.

N2

The time at which the first tide height is specified.

N3

The value of the tide at the first location (Nn+1 below) on the specified boundary.

N4

The value of the tide at the second location (Nn+2 below) on the specified boundary.

N5, ..., Nn

Triplets of time, and tide at the first and second locations in the manner of N2 through N4.

Nn+1, ..., Nn+2

Grid coordinates of the starting and ending locations on the boundary for linear interpolation of the tide.

EXAMPLES

TIDE NORth LINEar 3 sets: (0,1,0) (5.E4,5,10) (1.E5,0,1) at X: 0, 1000
MODE 4: Tide Height Determined by a Connected Reservoir

SYNTAX

\[ \text{TIDE } \{ \text{RESE} \} \{ \text{dir} \} \{ Q_R \} \]

RESE

The tidal boundary condition at the specified boundary is given by:

\[ \frac{\partial \eta}{\partial t} = \pm \frac{Q}{Q_R} \]

dir

One of the character strings: EAST, WEST, NORTH, or SOUTH. It denotes the boundary of the domain of interest where the water elevation is specified as a function of time and space. If no direction is specified, then the tidal boundary is assumed to be at the west.

Q_R

The reservoir capacity factor. The value must be greater than or equal to zero. The default value is 0.

EXAMPLES

TIDE EAST boundary: RESErvoir, capacity factor = 1.0E10
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>PURPOSE</td>
<td>To set the initial simulation time for a problem.</td>
</tr>
<tr>
<td>SYNTAX</td>
<td>TIME { N1 }</td>
</tr>
<tr>
<td>N1</td>
<td>The starting time (≥0) for simulations. The default value is 0.</td>
</tr>
</tbody>
</table>

**EXAMPLES**

```
TIME = 50 years at start of simulations
```
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>TITLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>PURPOSE</td>
<td>To specify the problem title.</td>
</tr>
<tr>
<td>SYNTAX</td>
<td>TITL followed by character information</td>
</tr>
</tbody>
</table>

**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.

<table>
<thead>
<tr>
<th>option</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>Particle tracking stops if its x-coordinate exceeds the Nn+1\textsuperscript{th} numerical value.</td>
</tr>
<tr>
<td>Y or R</td>
<td>Particle tracking stops if its y-coordinate exceeds the Nn+1\textsuperscript{th} numerical value.</td>
</tr>
<tr>
<td>Z or THETA</td>
<td>Particle tracking stops if its z-coordinate exceeds the Nn+1\textsuperscript{th} numerical value.</td>
</tr>
<tr>
<td>DIST</td>
<td>Particle tracking stops when its distance from the point of release exceeds the Nn+1\textsuperscript{th} numerical value. This is default option if STOP modifier is specified.</td>
</tr>
<tr>
<td>TIME</td>
<td>Particle tracking stops when the time exceeds the Nn+1\textsuperscript{th} numerical value.</td>
</tr>
<tr>
<td>ELAP</td>
<td>Particle tracking stops when the elapsed time from its moment of release exceeds the Nn+1\textsuperscript{th} numerical value.</td>
</tr>
<tr>
<td>FREE</td>
<td>Particle tracking stops if it reaches a free surface or zone of saturation.</td>
</tr>
</tbody>
</table>

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 (\geq 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  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**  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 STACK WRITE 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 STACK WRITE 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 OUTPUT command. The output is obtained in the form of a list. Where possible the list is organized by element or node number. Output is provided as soon as the command is encountered.

**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.

<table>
<thead>
<tr>
<th>option</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>INLE</td>
<td>Only the boundaries specified by the INLET command are selected.</td>
</tr>
<tr>
<td>OUTL</td>
<td>Only the boundaries specified by the OUTLET command are selected.</td>
</tr>
<tr>
<td>OPEN</td>
<td>Only the boundaries specified by the OPEN command are selected.</td>
</tr>
<tr>
<td>IO</td>
<td>All boundaries specified by INLET, OUTLET or OPEN command are selected.</td>
</tr>
<tr>
<td>BOUN</td>
<td>All external boundaries of the specified (or default) subregion are selected.</td>
</tr>
<tr>
<td>SYMM</td>
<td>Only the boundaries specified by the SYMMETRY command are selected.</td>
</tr>
<tr>
<td>WALL</td>
<td>Only the walls are selected.</td>
</tr>
</tbody>
</table>

walltype
The type of wall to be selected for output if the WALL modifier is present.

<table>
<thead>
<tr>
<th>walltype</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALL</td>
<td>All walls are selected. This is the default option.</td>
</tr>
<tr>
<td>EXTE</td>
<td>Only the exterior walls of the computational domain are selected.</td>
</tr>
<tr>
<td>INTE</td>
<td>Only the walls located in the interior of the computational domain are selected.</td>
</tr>
<tr>
<td>BLOC</td>
<td>Only the walls of the blocks (BLOCK command) are selected.</td>
</tr>
</tbody>
</table>

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

\[ \text{WRIT} \{ \Phi \} \text{ [INTE] [method] [STAC] [subrgn] [fname]} \]

\[ \Phi \]

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.

\[ \text{INTE} \]

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.

\[ \text{method} \]

The method of interpolation for computing values at the specified (x, y, z) locations. Only one method may be selected for each command.

<table>
<thead>
<tr>
<th>method</th>
<th>INTERPRETATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINE</td>
<td>The values are computed by inverse linear distance interpolation from the computed values at nearest neighbors. This is the default option.</td>
</tr>
<tr>
<td>SQUA</td>
<td>The values are computed by inverse distance squared interpolation from the computed values at nearest neighbors.</td>
</tr>
<tr>
<td>AVER</td>
<td>The values are computed by arithmetic average of computed values at nearest neighbors.</td>
</tr>
<tr>
<td>NEAR</td>
<td>The value is set equal to that at the nearest neighbor.</td>
</tr>
<tr>
<td>X</td>
<td>Same as LINE except that the distance is set equal to the separation in the x-coordinate.</td>
</tr>
<tr>
<td>Y</td>
<td>Same as LINE except that the distance is set equal to the separation in the y-coordinate.</td>
</tr>
<tr>
<td>Z</td>
<td>Same as LINE except that the distance is equal to the separation in the z-coordinate.</td>
</tr>
</tbody>
</table>

\[ \text{STAC} \]

The output variable is scaled by coefficients a and b specified on a previous STACK WRITE command according to:

\[ \Phi_{\text{out}} = a \Phi_{\text{computed}} + b \]

The STAC modifier is effective only for real variables. It is ignored if no previous STACK WRITE command was specified.

\[ \text{subrgn} \]

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.

\[ \text{fname} \]

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=LPROFILE
WRITe U, V, P and T by SQUAred INTErpolation for ID=LPROFILE
WRITe U, V, P and T by INTErpolation for ID=LPROFILE
WRITe U by AVERAGE INTErpolation for ID=LPROFILE on file: ‘PROFILE.U’
WRITe U, V, P and T by X direction INTErpolation for ID=LPROFILE on file: ‘PROFILEX.VAR’
MODE 4: Write Selected Variables to User-Specified File in Block Mode

SYNTAX  
\texttt{WRIT} \{\Phi}\} \{BLOC\} \{HEAD\} \{FIEL\} \{fname\}

\Phi \hspace{1cm} 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 \hspace{1cm} 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 \hspace{1cm} By default only the numeric values for the selected variable(s) are written to the file without any header information. If the \texttt{HEADER} modifier is present, then a two line header in the standard ACRi SAVE file format appears before each set of variable values.

FIEL \hspace{1cm} 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 \texttt{BLOCK} modifier is simultaneously present. By default both the field and the boundary elements are written to the record.

fname \hspace{1cm} 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

\texttt{WRITe} T in BLOCK format to file named 'value.T'
\texttt{WRITe} U, V and W in BLOCK format to file 'value.UVW'
\texttt{WRITe} U, V, W, T, C, FU in BLOCK format to file named 'mixed.val'
\texttt{WRITe} FIELd values of U and X in BLOCK format 'UandX.val'
\texttt{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 mth 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 CONNECTIVITY 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


(This page left intentionally blank)
APPENDIX A

PARTIAL LIST OF PUBLICATIONS

TIDAL™ has been extensively used over the last 30 years. More than 100 publications and project reports on the benchmarking, verification and application of TIDAL™ are currently available. This appendix presents a partial list of these publications.


Phanikumar, M.S., M.D. Raghunath, and K.S. Yajnik, 1992. Modelling and Simulation of Pollutant Dispersion in Thane Creek. CSIR Center for Mathematical Modelling and Computer Simulation, National Aeronautical Laboratory, Bangalore 560 017, INDIA.


**Lake Manzala, Egypt.** Design of the outfall from a wastewater treatment plant and analysis of its environmental impact on Lake Manzala and the Mediterranean Sea. ACRi Project # 003-10-01 for Black & Veatch International and James M. Montgomery. March 1990.


**Kahe Offshore, Hawaii.** Three dimensional modeling of thermal discharge from the proposed GE design of an OTEC Power Plant offshore of Kahe Point. ACRi Job# 003-07 for Dames & Moore, Honolulu, and GE, Albany, NY. Sep. 1982.


**Coquimbo Bay, Chile.** Water quality and hydrodynamic modeling of Coquimbo Bay for design of effluent discharge outfall. ACRi Job # 003-02 for Ingenieros Civiles Consultores, Ltd., Santiago, Chile. November 1980.


**Kaneohe Bay, Oahu, Hawaii.** Computer modeling of Kaneohe Bay for urban water resources and impact of various land use scenarios. Dames & Moore Job # 4401-034-11 for U.S. Army Engineer District, Honolulu, HA. July 1977.


St. Rosalie, Louisiana. Hurricane surge analysis, proposed St. Rosalie Generating Station, Units 1 and 2, St. Rosalie, Louisiana. Dames & Moore Job # 3499-003-02 for Louisiana Power and Light Co. February 1975.
(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 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 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 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 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 **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 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 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 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 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>
<thead>
<tr>
<th>Number</th>
<th>Character</th>
<th>Description</th>
<th>ASCII Sequence #</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><code> </code></td>
<td>Space or Blank</td>
<td>32</td>
</tr>
<tr>
<td>2</td>
<td>#</td>
<td>Number or pound sign</td>
<td>35</td>
</tr>
<tr>
<td>3</td>
<td>(</td>
<td>Left parenthesis</td>
<td>40</td>
</tr>
<tr>
<td>4</td>
<td>)</td>
<td>Right parenthesis</td>
<td>41</td>
</tr>
<tr>
<td>5</td>
<td>,</td>
<td>Comma</td>
<td>44</td>
</tr>
<tr>
<td>6</td>
<td>:</td>
<td>Colon</td>
<td>58</td>
</tr>
<tr>
<td>7</td>
<td>;</td>
<td>Semicolon</td>
<td>59</td>
</tr>
<tr>
<td>8</td>
<td>=</td>
<td>Equal sign</td>
<td>61</td>
</tr>
<tr>
<td>9</td>
<td>^</td>
<td>Caret or Circumflex</td>
<td>94</td>
</tr>
</tbody>
</table>

**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 Terminator Character

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

Structure

♦ The dollar ($) and exclamation mark (!) characters are the only valid terminators.

♦ The terminator terminates the input for the keyword or continuation record on which it occurs; input associated with that particular keyword may continue on a continuation record which follows.

♦ The terminator may appear anywhere in a record.

♦ Any characters following the terminator on that input record are not processed but are treated as user comments and are merely written to the output file.

EXAMPLES

The character sequences:

XYZ $comments now
! any comments here
123.456 !789.123

are all examples of sequences with embedded terminators. In the first sequence, XYZ will be treated as valid character data (either keyword or modifier, depending on its starting position on the input record) whereas the characters following $ will be ignored. In the second example, the complete sequence will be treated as comments. In the third example, the numerical field 789.123 will be ignored, whereas 123.456 will be treated as numeric data.
B.2.6  Comment Field

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

Structure
♦ A comment field may be in the form of an embedded comment or a comment record.
♦ An embedded comment field is one which occurs on a keyword or continuation record. It must begin with a terminator ($ or !) character. Any combination of characters may follow the terminator. The comment field is terminated at the end of the 256-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 256-th 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 Prompt Character

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

Structure

♦ The user may insert a '?' in place of either an entire input record or numerical or character string in the input itself. The command interpreter will pause and prompt the user for input at this stage.

♦ The input obtained by the prompt may be comprised of one or more of the elements of an input record described in Sections 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 TIDAL™ INPUT AND OUTPUT

Illustrative examples of TIDAL™ input command files, and output obtained from them, are supplied under separate cover.