

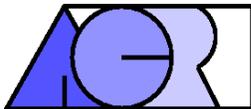
POR FLOW

a software tool for multiphase fluid flow, heat and mass transport in fractured porous media

USER'S MANUAL

VERSION 5.0

Rev:5



©ANALYTIC & COMPUTATIONAL RESEARCH, INC. 1985-2002

ALL RIGHTS RESERVED

NOTIFICATION OF COPYRIGHT

The **PORFLOW™** 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**.

March 25, 2002

© ACRi

IMPORTANT -- PLEASE READ CAREFULLY BEFORE USING THIS SOFTWARE

SOFTWARE LICENSE AGREEMENT

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

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 containing the SOFTWARE and accompanying documentation are free of defects in materials and workmanship for a period of 30 days from the date of purchase. In the event of notification of any physical defects in diskettes or documentation, ACRi will replace the defective diskettes and documentation.
10. ACRi disclaims all other warranties, expressed or implied, including but not limited to implied warranties of fitness or usefulness for a particular purpose, either with respect to defects in the diskettes, and documentation, or with respect to any defects of coding and logic in the SOFTWARE, even if such defects are known to ACRi or have been brought to ACRi's attention.
11. ACRi will not be liable to CUSTOMER, or any other party, for consequential loss or damages arising from the use, operation, or modification of the SOFTWARE by CUSTOMER. ACRi will not be responsible for any loss, damage, or claim resulting from reliance by CUSTOMER or any other party on results obtained by use of the SOFTWARE.
12. If any provision of this AGREEMENT shall be unlawful, void, or for any reason unenforceable, then

that provision shall be deemed severable from this AGREEMENT and shall not affect the validity and enforceability of the remaining provisions of this AGREEMENT.

WARRANTY

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

NOTICE

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

PREFACE

During the past 25 years, **PORFLOW™** has evolved from a simple mathematical model for flow and heat transfer into a comprehensive **software tool for analysis of a wide range of environmental applications in flow, heat and mass transport in geologic media**. It provides for coupled transport of flow, heat and multiple chemical species in complex three-dimensional geometry, transient or steady-state flow, confined or unconfined aquifers, fully or partially saturated media, single or multiple phase systems, and phase change between liquid and solid and liquid and gaseous phases. It has grown from a simple computer code with few modules and fewer options to a software package with well over 300 modules and a versatile set of options that can accommodate almost any user requirements

PORFLOW™ has been used to analyze problems as diverse as salinity intrusion into fresh water aquifers and remediation of hazardous waste sites. It has been used to evaluate pumping of an aquifer over a period of days, remediation of waste sites over a period of years, corrosion of waste canisters over tens of years, and transport of contaminants from nuclear waste over a time span of hundreds of thousands of years.

In the process, **PORFLOW™** 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 **PORFLOW™** 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 **PORFLOW™**, 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 **PORFLOW™** to emerge as leading software in its field of application.

PORFLOW™ is also distinguished from other computer models by the diversity of its users. Commercial, research and educational organizations in 15 countries are using the software. Among its users are: U.S. DOE, USGS, U.S.NRC, U.S.Army, Southwest Research Institute, Idaho National Engineering Laboratory, Oak Ridge National Laboratory, Savannah River Laboratory, Battelle Pacific Northwest Laboratory, ANDRA (France), SCK-CEN (Belgium), AECL (Canada), Westinghouse, Lockheed Martin, Fluor Daniel, Rockwell, and a large number of other commercial organizations. Over 100 publications and project reports on the benchmarking, verification and application of **PORFLOW™** are currently available.

PORFLOW™ has been extensively peer-reviewed. Idaho National Engineering Laboratory, Battelle Pacific Northwest, and Prof. Allan Freeze of the University of British Columbia have formally reviewed **PORFLOW™** or its derivatives. Additionally, it has been reviewed by ANDRA (France), BAe-SEMA (UK), British Petroleum (UK), Exxon Production Research, Failure Analysis Associates Inc., Fluor Daniel Inc., Gaz de France (France), SAIC, Shell Oil, SOHIO, and Westinghouse Hanford Company.

PORFLOW™ 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 **PORFLOW™** 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 enquiries, 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
March 25, 2002

ACKNOWLEDGEMENTS

The **PORFLOW™** 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

	Page
NOTIFICATION OF COPYRIGHT	2
SOFTWARE LICENSE AGREEMENT	3
WARRANTY	5
NOTICE	5
PREFACE.....	6
ACKNOWLEDGEMENTS.....	7
NOTATION	16
CHAPTER 1.....	21
1.1 OVERVIEW OF CAPABILITIES	22
1.2 VALIDATION AND PREVIOUS APPLICATIONS	24
1.3 STRUCTURE OF PORFLOW™	25
1.4 SCOPE AND LIMITATIONS	26
1.4.1 System of Equations.....	26
1.4.2 Spatial Dimensionality	26
1.4.3 Problem Geometry.....	26
1.4.4 Temporal Dependence	26
1.4.5 Spatial Dependence	26
1.4.6 Methods for Solving Governing Equations	27
1.4.7 Boundary Conditions	27
1.4.8 Sources and Sinks.....	27
1.4.9 Multiple Phases and Variable Saturation	27
1.4.10 Free Surface and Unconfined Flow	27
1.4.11 Phase Change	28
1.4.12 Pore Structure.....	28
1.4.13 Special Geologic Features	28
1.4.14 Operational and Output Control	28
CHAPTER 2.....	30
2.1 THE GENERAL TRANSPORT EQUATION	31
2.2 GOVERNING EQUATIONS FOR FLUID FLOW	32
2.2.1 Continuity or Mass Balance Equations	32
2.2.2 Momentum Balance Equation	33
2.2.3 Pressure Equations	34
2.3 GOVERNING EQUATIONS FOR HEAT TRANSFER.....	36
2.4 GOVERNING EQUATIONS FOR MASS TRANSPORT IN FLUID.....	38

2.5	GOVERNING EQUATIONS FOR MASS TRANSPORT IN SOLID.....	40
CHAPTER 3..... 42		
3.1	FLUID DENSITY	43
3.2	FLUID VISCOSITY	44
3.3	CHARACTERISTIC SATURATION RELATIONS.....	45
3.4	EFFECTIVE STORATIVITY	48
3.5	HYDRAULIC CONDUCTIVITY	49
3.6	FLUID LATENT AND SPECIFIC HEATS	51
3.7	VAPOR FORMATION EQUATION.....	52
3.8	EQUILIBRIUM VAPOR PRESSURE.....	53
3.9	HYDRODYNAMIC DISPERSION.....	54
3.10	FLUID, HEAT OR MASS SOURCES OR SINKS.....	55
3.11	BOUNDARY CONDITIONS	56
	3.11.1 General Boundary Condition	56
	3.11.2 Dirichlet Boundary Condition.....	56
	3.11.3 Neumann Boundary Condition.....	56
	3.11.4 Mixed Boundary Condition	56
3.12	INITIAL CONDITIONS.....	57
3.13	FIELD VARIABLE INTERRELATIONSHIPS	58
CHAPTER 4..... 60		
4.1	THE GRID ARRANGEMENT.....	61
4.2	DISCRETIZATION SCHEME.....	65
4.3	SOLUTION METHOD	67
4.4	NUMERICAL STABILITY AND ACCURACY	68
	4.4.1 General Considerations.....	68
	4.4.2 Time Scale of Pressure Propagation	68
	4.4.3 Time Scale of Diffusion.....	69
	4.4.4 Time Scale of Convection.....	69
	4.4.5 Other Pertinent Time Scales	69
	4.4.6 Grid Peclet Number	70
4.5	CONVERGENCE	71
CHAPTER 5..... 73		
5.1	SPECIFICATION OF PROBLEM GEOMETRY	74
5.2	SPECIFICATION OF SPATIAL AND TEMPORAL GRID	75
	5.2.1 Basic Considerations.....	75
	5.2.2 Computer Resources.....	75

5.2.3	Numerical Stability and Accuracy.....	75
5.2.4	Scale of Heterogeneity	75
5.2.5	Scale of Physical Resolution	75
5.2.6	Sources and Sinks.....	75
5.2.7	Boundary Conditions	76
5.2.8	Input and Output Requirements	76
5.2.9	Planar and Linear Features.....	75
CHAPTER 6.....		78
6.1	OVERVIEW OF KEYWORD COMMAND LANGUAGE.....	79
6.2	ORDER OF INPUT COMMANDS.....	83
6.3	SPECIFICATION OF GRID AND PROBLEM DIMENSIONALITY	84
6.4	INPUT AND OUTPUT FILE UNITS	85
6.5	UNITS OF PHYSICAL QUANTITIES.....	87
6.6	DIAGNOSTIC OUTPUT SPECIFICATION.....	88
6.7	DEPENDENT VARIABLES OF PORFLOW™	89
6.8	TABULAR OUTPUT OF FIELD VARIABLES.....	90
6.9	ARCHIVAL AND POST-PROCESSING OUTPUT.....	93
6.10	RESTART OUTPUT	94
6.11	MODIFICATION OF INPUT DATA DURING SIMULATIONS	95
6.12	DIMENSIONING PARAMETER.....	97
CHAPTER 7.....		99
7.1	KEYWORD COMMAND NOTATION	100
7.2	GENERAL INPUT FEATURES FOR FREEFORM™COMMANDS	101
7.2.1	Interactive or Run Time Input	101
7.2.2	File Name Specification	101
7.2.3	Subregion Specification	102
7.2.4	Subregion Identification	103
7.2.5	Boundary Identification	104
7.2.6	General Functional Relationships	106
7.2.7	Command Input of Functional Form	108
7.2.8	Examples of Functional Form of Input	111
7.2.9	User-Defined Input Options	114
7.2.10	Specification of Fluid Phase	114
7.2.11	Specification of Frequency for Output.....	114
7.3	DESCRIPTION OF KEYWORD COMMANDS	115
	ADAP	116
	ALLOCATE	117
	BANNER	119
	BOUNDARY	120

CAPILLARY 125

CLOSE 126

CONDUCTIVITY 127

CONNECTIVITY 133

CONVERGENCE 143

COORDINATE 149

CORRELATION 156

DATUM 159

DEBUG 160

DECAY 163

DEFINE 165

DENSITY 166

DIAGNOSTIC 168

DIFFUSION 170

DISABLE 171

DISTRIBUTION COEFFICIENT 173

EBU 178

ELSE 180

END 181

ENDIF 182

FILE 183

FIX 184

FLOW 187

FLUID 197

FLUX 202

FOR 209

GAS 210

GEOMETRY 213

GRAVITY 219

GRID 220

HISTORY 224

HYDRAULIC 229

IF-ELSE-ENDIF 230

INCLUDE 232

INITIAL 233

INLET 234

INTEGRATION 235

LIMIT 236

LOCATE 237

MATERIAL 257

MATRIX 265

MULTPHASE 274

OPEN 281

OPTION 282

OUTLET.....	283
OUTPUT.....	285
PAUSE.....	289
PERIODIC.....	290
PHASE.....	294
PRECIPITATION.....	295
PRINT.....	298
PROBLEM.....	308
PROPERTY.....	312
QUIT.....	313
REACTION.....	314
READ.....	320
REFERENCE.....	322
REGENERATION.....	324
RELAX.....	325
RENAME.....	326
RETARDATION.....	327
ROCK.....	328
SAVE.....	329
SCALE.....	335
SCREEN.....	336
SELECT.....	337
SET.....	338
SOLVE.....	350
SOURCE.....	355
STACK.....	371
STATISTICS.....	374
STORAGE.....	375
SWIRL.....	376
SYMMETRY.....	382
THERMAL.....	383
TIME.....	384
TITLE.....	385
TRACK.....	386
TRANSPORT.....	390
USER.....	391
VELOCITY.....	392
VISCOSITY.....	393
WRITE.....	396
ZONE.....	404
REFERENCES.....	406
APPENDIX A.....	407
PARTIAL LIST OF PUBLICATIONS.....	407

APPENDIX B 415

ACRi FREEFORM™ COMMAND LANGUAGE 415

APPENDIX C 425

EXAMPLES OF PORFLOW™ INPUT AND OUTPUT 425

LIST OF TABLES

TABLE 3.8.1: VAPOR PRESSURE FOR WATER 53

TABLE 6.2.1: FUNCTIONAL UNITS OF PORFLOW™ COMMANDS..... 83

TABLE 6.4.1: I/O FILE UNITS AND THEIR DEFAULT ASSIGNMENT 85

TABLE 6.7.1: DEPENDENT VARIABLES OF PORFLOW™ 89

TABLE 6.8.1: STANDARD VARIABLES OF PORFLOW™ 91

TABLE 6.8.2: SUPPLEMENTARY VARIABLES OF PORFLOW™ 92

TABLE 6.11.1: ILLUSTRATION OF A TWO-SEGMENT SIMULATION SEQUENCE..... 96

TABLE 7.2.1: SUBREGION DEFINITION AND APPLICABILITY..... 102

TABLE 7.2.2: VALID SUBREGION IDENTIFICATION MODIFIERS 103

TABLE 7.2.3: BOUNDARY IDENTIFICATION INDEX..... 104

TABLE 7.2.5: VALID INDEPENDENT VARIABLES 110

TABLE 7.2.6: NUMERICAL INPUT FOR THE TABLE SPECIFICATION 110

TABLE 7.2.7: NUMERICAL INPUT FOR SIMPLE ANALYTIC FUNCTIONS..... 110

TABLE 7.2.8: NUMERICAL INPUT FOR THE SERIES FUNCTIONS 110

TABLE 7.2.9: NUMERICAL INPUT FOR USER SPECIFIED FUNCTIONS 110

TABLE 7.2.10: EXAMPLES OF SIMPLE ANALYTIC FUNCTIONS 111

TABLE 7.2.11: EXAMPLES OF ANALYTIC FUNCTION SERIES 112

TABLE 7.2.12: EXAMPLES OF TABULAR FUNCTIONS..... 113

TABLE 7.2.13: VALID MODIFIERS FOR PHASE DESIGNATION 114

LIST OF FIGURES

FIGURE 1.1.1: PROCESSES INCORPORATED INTO PORFLOW™23

FIGURE 4.1.1: CARTESIAN GRID ARRANGEMENT 62

FIGURE 4.1.2: CYLINDRICAL GRID ARRANGEMENT..... 62

FIGURE 4.1.3: TYPICAL GRID ELEMENT IN CARTESIAN COORDINATES..... 63

FIGURE 4.1.4: TYPICAL GRID ELEMENT IN CYLINDRICAL COORDINATES 63

FIGURE 4.1.5: TYPICAL GRID ELEMENT IN X-Y PLANE 64

FIGURE 7.2.1: BOUNDARY INDEX NOTATION 105

NOTATION

Symbol	MEANING	UNITS		
		GENERIC	SI	FPS
B	Buoyancy parameter	---	---	---
c	Specific heat	$L^2 t^{-2} T^{-1}$	J/(kg K)	BTU/(lbm°F)
C	Species concentration in fluid	$M L^{-3}$	kg/m ³	lbm/ft ³
D	Dispersion coefficient	$L^2 t^{-1}$	m ² /s	ft ² /s
D _M	Molecular diffusivity	$L^2 t^{-1}$	m ² /s	ft ² /s
F	General transport variable	various	various	various
g	Gravitational acceleration	$L t^{-2}$	m/s ²	ft/s ²
h	Enthalpy of a fluid phase	$L^2 t^{-2}$	J/kg	BTU/lbm
H	Volumetric enthalpy of fluid	$M L^{-1} t^{-2}$	J/m ³	BTU/ft ³
k	Intrinsic permeability	L^2	m ²	ft ²
k	Thermal conductivity of fluid	$M L t^{-3} T^{-1}$	W/(m K)	BTU/(ft s°F)
k _d	Partition coefficient	$M^{-1} L^3$	m ³ /kg	ft ³ /lbm
k _r	Relative hydraulic conductivity	---	---	---
K	Hydraulic conductivity	$L t^{-1}$	m/s	ft/s
m	Rate of injection of fluid	$M L^{-3} t^{-1}$	kg/(m ³ s)	lbm/(ft ³ s)
m	Mass fraction of a species	---	---	---
m _v	Volumetric mass injection rate	t^{-1}	m ³ /(m ³ s)	ft ³ /(ft ³ s)
M	Molecular weight	$M mol^{-1}$	kg/mol	lbm/mol
p	Thermodynamic pressure	$M L^{-1} t^{-2}$	N/m ²	lbf/ft ²
P	Hydraulic pressure head	L	m	ft
Q	Amount of fluid in storage	$M t^{-3}$	W/(m ²)	BTU/(ft ² s)
r	Radius of curvature	L	m	ft
R	Density ratio	---	---	---
R _d	Retardation factor	---	---	---
R _u	Universal gas constant	$L^2 t^{-2} T^{-1}$	J/(kg molK)	BTU/(lbm-mol R)
s _F	Rate constant for sink of property F	t^{-1}	1/s	1/s
S	Saturation fraction of fluid	---	---	---

..... NOTATION continued

NOTATION - Continued

Symbol	MEANING	UNITS		
		GENERIC	SI	FPS
S	Source for a dependent variable	various	various	various
S _C	Source for a species	M L ⁻³ t ⁻¹	kg/(m ³ s)	lbm/ft ³ s)
S _e	Effective storativity	L ⁻¹	1/m	1/ft
S _r	Residual phase saturation	---	---	---
S _T	Heat source	M L ⁻¹ t ⁻³	W/(m ³)	BTU/(ft ³ s)
t	Time	t	s	s
T	Thermodynamic Temperature	T	K	R
T _a	Base to convert temp. to absolute units	T	K	R
T _c	Critical temperature	T	K	R
T _F	Freezing temperature	T	K	R
U	Darcy velocity in x-direction	L t ⁻¹	m/s	ft/s
v	Pore or fluid particle velocity	L t ⁻¹	m/s	ft/s
V	Darcy velocity in y-direction	L t ⁻¹	m/s	ft/s
V _i	Velocity in i th direction	L t ⁻¹	m/s	ft/s
\underline{V}	Total velocity vector	L t ⁻¹	m/s	ft/s
W	Darcy velocity in z-direction	L t ⁻¹	m/s	ft/s
x	x-coordinate	L	m	ft
y	y-coordinate	L	m	ft
z	z-coordinate	L	m	ft

GREEK SYMBOLS

Symbol	MEANING	UNITS		
		GENERIC	SI	FPS
α_f	Fluid compressibility	$M^{-1} L t^2$	m^2/N	ft^2/lbf
α_L	Longitudinal dispersivity	L	m	ft
α_s	Compressibility of solid matrix	$M^{-1} L t^2$	m^2/N	ft^2/lbf
α_s	Pressure coefficient of phase saturation	$M^{-1} L t^2$	m^2/N	ft^2/lbf
α_T	Transverse dispersivity	L	m	ft
β_f	Expansion coefficient for fluid	T^{-1}	1/K	1/°F
β_s	Temperature coefficient of phase saturation	T^{-1}	1/K	1/°F
Γ	Diffusion coefficient	$M L^{-1} t^{-1}$	kg/(m s)	lbm/(ft s)
ϵ	A small quantity	---	---	---
Θ	Angular coordinate	radian	radian	radian
Θ	Porosity of the matrix	---	---	---
λ	Reaction rate for species	t^{-1}	1/s	1/s
μ	Fluid viscosity	$M L^{-1} t^{-1}$	kg/(m s)	lbm/(ft s)
ρ	Fluid density	$M L^{-3}$	kg/m ³	lbm/ft ³
σ	Generation fraction for species	---	---	---
τ	Tortuosity factor	---	---	---
φ	Volume fraction filled with a given fluid	---	---	---
ψ	Capillary pressure	L	m	ft

SUBSCRIPTS

SYMBOL	PERTAINING TO
D	Diffusional component of porosity or fluid filled volume fraction
E	Effective component of porosity or fluid filled volume fraction
e	Equivalent of effective value for the fluid-solid matrix
f	The fluid phase
i	The i^{th} coordinate direction
j	The j^{th} coordinate direction
j	The j^{th} chemical species
s	The solid phase
t	The time coordinate
T	Total porosity or fluid filled volume fraction
x	The x direction
y	The y or r direction
z	The z or Θ direction
1	The primary fluid
2	The secondary fluid
3	The third fluid

SUPERSCRIPTS

SYMBOL	PERTAINING TO
n	The n^{th} fluid or the n^{th} chemical species
k	The k^{th} time step
1	The primary fluid
2	The second fluid
3	The third fluid
*	A reference state of the system
~	A representative or average value
^	A non-dimensional value

(This page left intentionally blank.)

CHAPTER 1

OVERVIEW AND INTRODUCTION

PORFLOW™ is a software tool for solution of multi-phase fluid flow, heat transfer, and mass transport problems in variably saturated porous or fractured media. It is a highly flexible, modular, and user-oriented software package. Due to the modular nature of **PORFLOW™** the user may readily customize the software for specific needs.

The software employs the **FREEFORM™** command language and **acrSHELL™** GUI pre-processor to provide a flexible, simple to use, and format-free user-interface. It interfaces with the **acrVIZUALIZER™** 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 4.00, which can be operated on a broad range of micro, mini, main frame and super computers. An outline of the theory, a complete description of the input and output options, and examples of its use are given in the following chapters. The theory on which the software is based is described in more detail by publications listed in the **REFERENCES** section and **APPENDIX A** of this manual. This chapter describes the various capabilities and applications of **PORFLOW™**.

1.1 OVERVIEW OF CAPABILITIES

The **PORFLOW™** software package is a comprehensive mathematical model for simulation of multi-phase fluid flow, heat transfer, and mass transport processes in variably saturated porous and fractured media. 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 **PORFLOW™** easily without extensive training.

PORFLOW™ can be used to simulate transient or steady state problems in Cartesian or cylindrical geometry. The geologic medium may be anisotropic and heterogenous, and may contain distinct embedded elements such as, discrete fractures or boreholes within a porous matrix. It provides a unified theoretical treatment of concepts relevant to fluid flow and transport. The physical processes incorporated in the software are shown in Figure 1.1.1. As can be seen from this figure, various levels of interaction and coupling occur between the different components of the flow system. In the **PORFLOW™** software package, these components may be employed in either a coupled or uncoupled mode. Some of the important features are:

- *Cartesian or cylindrical, structured or unstructured geometry*
- *Transient or steady state simulation*
- *Option to add new variables and arbitrary number of transport equations*
- *Option to solve any or all of the governing equations*
- *Dynamic coupling between flow, heat and mass transport*
- *Powerful built-in library functions for input of physics of the problem*
- *Arbitrary user-defined functions for problem specification*
- *Horizontal or vertical cross-sections*
- *Saturated, Variably saturated, free surface multi-phase flow*
- *Dynamically active liquid-solid and liquid-vapor phase change*
- *Inhomogeneous, anisotropic and time-dependent properties*
- *Inherently mass-conservative numerical method*
- *Inclusion of buoyancy effects due to density variations*
- *Alternate algorithms for changes in the media properties*
- *Time-dependent options for physical input*
- *Format-free, conversational input with built-in default values*
- *Flexible operational and output control*
- *Choice of 'basis functions' for integration of equations*
- *Alternate methods for solution of the matrix of equations*

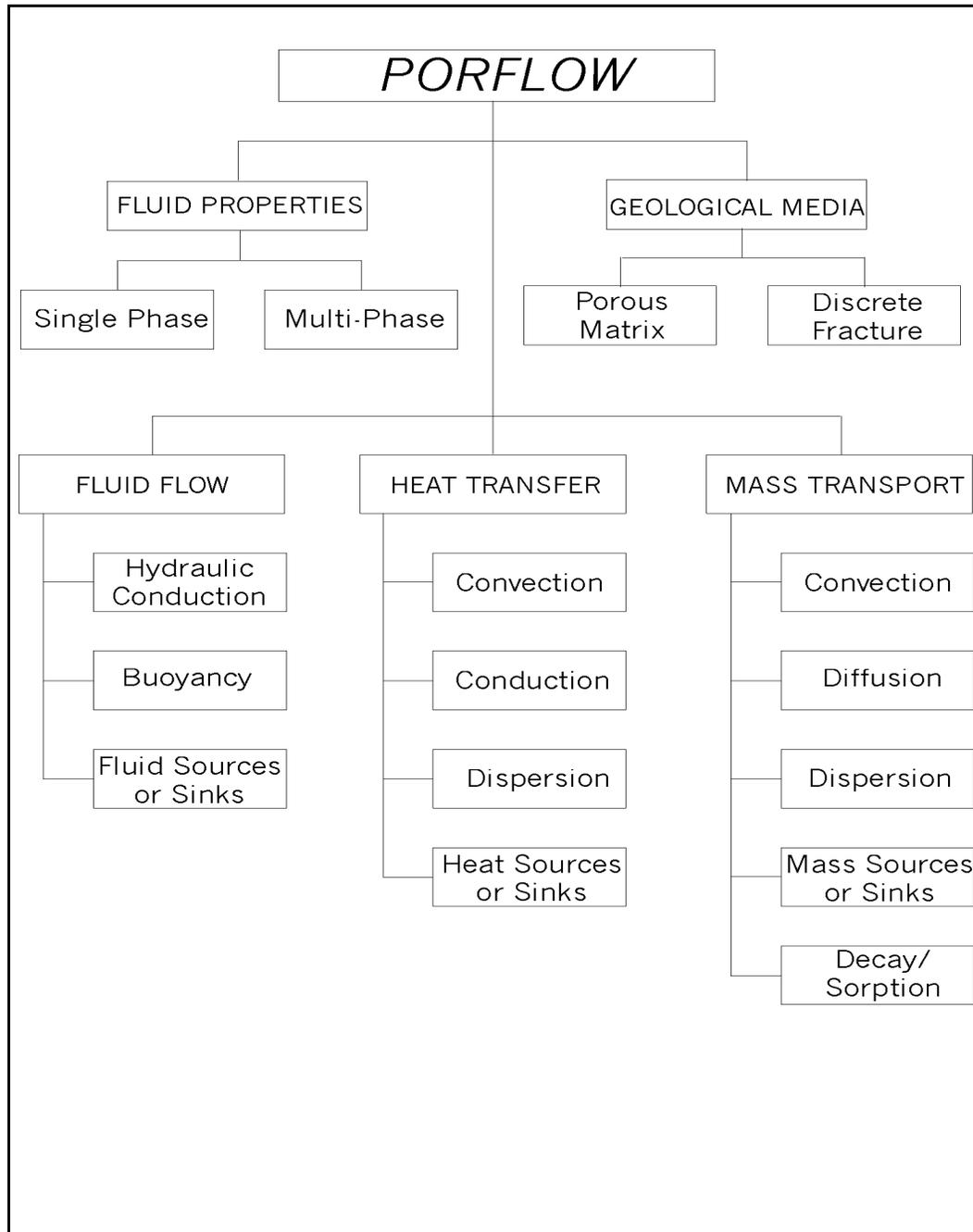


FIGURE 1.1.1: PROCESSES INCORPORATED INTO PORFLOW™

1.2 VALIDATION AND PREVIOUS APPLICATIONS

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

- **Analysis of high- and low-level nuclear waste repositories**
- **Pollution of ground water by organic and inorganic chemicals**
- **Organic contamination of ground water**
- **Ground water resource and pumping studies**
- **Flow, heat transfer and chemical reaction in porous geologic media**
- **Storage and withdrawal of hot water in deep aquifers**
- **Intrusion of seawater into coastal aquifers**
- **Thawing/freezing of ground due to buried oil and gas pipelines**
- **Propagation of freezing fronts in soils**
- **Analysis of under-sea oil pipelines**
- **Interaction of ground water systems with atmosphere**
- **Corrosion of waste canisters and liners**
- **Analysis of hydrologic effects of reverse-circulation drilling**
- **Performance enhancement of oil wells by optimization of casing perforation**
- **Dewatering of mines**
- **Consolidation of soils**
- **Analysis of lithification processes.**

1.3 STRUCTURE OF PORFLOW™

PORFLOW™ has been 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 nature of the geometry, itself, can be specified to be Cartesian, cylindrical or body-fitted. The grid may be structured or unstructured. The domain of interest is considered to be composed of several distinct 'material types'. These may differ from each other by virtue of differences in a physical property, such as hydraulic permeability, or the presence or absence of a feature such as fractures. Each material type may be represented by an arbitrary number of contiguous or non-contiguous grid 'elements'. The values of field variables, such as velocity, pressure, temperature and species concentration, may be specified for any grid element. Arbitrary sources or sinks may be located in the interior of the domain of interest. The physics of the problem is either inferred from the specified initial and boundary conditions or is easily selected by simple user commands.

The last two objectives are achieved by employing a highly modular structure for the software which, for the current version, consists of more than 300 individual modules. In general, each module is dedicated to a single function and can be replaced by an alternate module without significant change to the remaining modules or to the structure of **PORFLOW™**. The code provides options for alternate discretization schemes, solution methods, matrix solvers, fluid and porous media 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 **PORFLOW™** 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*

PORFLOW™ numerically solves a variable set of equations for general transport, multi-phase pressure, temperature and one or more chemical species. Constitutive equations, phase-change relations, equations of state, and initial and boundary conditions supplement the governing equations. The equations are coupled through convection, buoyancy, temperature, phase change, fluid density and viscosity 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 **PORFLOW™** permits simulation of flow systems with up to three phases. Examples of such systems are water-oil-vapor-air, water-steam-air, water-steam, water-ice or water-air systems.

1.4.2 *Spatial Dimensionality*

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

1.4.3 *Problem Geometry*

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

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

1.4.4 *Temporal Dependence*

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

1.4.5 *Spatial Dependence*

The values of most parameters are allowed to vary within the spatial grid. The model domain can be divided into zones, each differing from the others in one or more distinct way, such as in material properties. The material properties can also be either isotropic or anisotropic.

1.4.6 Methods for Solving Governing Equations

The method of Nodal Point Integration (NPI) is employed for integration of the governing differential equations by temporal and spatial discretization over each control volume (element) of the physical domain.

It leads to solutions that automatically conserve fluid, heat, and mass locally within every grid element, as well as for the entire flow domain. The storage terms are approximated by a modified Newton-Raphson method. The dependent variable or its change from the current state approximates the flux terms. The elements used to define the problem geometry can vary in size, but their shape is restricted to that of a quadrilateral, hexahedral or segment of a cylinder. Three distinct discretization schemes are available: the hybrid, CONDIF and QUICK schemes.

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

1.4.7 Boundary Conditions

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

1.4.8 Sources and Sinks

Several options are provided 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 by both.

1.4.9 Multiple Phases and Variable Saturation

Although the framework of **PORFLOW™** is sufficiently general to allow for any number of phases (or fluids), the current version allows for one (e.g., water), two (e.g., water and air) or three (e.g., water, oil and air) phase fluid flow. A single-phase system is also termed a "fully" saturated flow system, whereas a multi-phase system is referred to as a "variably" saturated flow system.

In the case of a multi-phase system, the software assumes that the phases are immiscible and that each phase has a distinct pressure that is, of course, a function of space and time. The pressure of each phase differs from that of the adjoining phase by a capillary pressure.

PORFLOW™ can be used to solve problems with variable saturation. In the partially saturated zone, liquids (water, oil, etc.) and gas (air, vapor, etc.) are assumed to co-exist. The degree of saturation of each phase is determined at each grid node as part of the solution.

1.4.10 Free Surface and Unconfined Flow

In its default mode, the software assumes that the flow domain is confined. The multi-phase option may be invoked to solve problems involving an unconfined flow domain. However, the multi-phase option is generally considerably more expensive to use than the saturated flow option. Consequently, if the vadose zone is not of interest, then **PORFLOW™** provides an alternative method to solve the equations governing unconfined flow. In this mode, only flow equations for the saturated region below the water table are solved.

1.4.11 Phase Change

PORFLOW™ addresses dynamic liquid-vapor (e.g., water-steam) and liquid-solid (e.g., water-ice) phase changes. The phase change algorithm is reversible and generalized, and provides for non-equilibrium thermodynamics. The user may arbitrarily change the phase-change parameters and fluid-property relations.

1.4.12 Pore Structure

PORFLOW™ assumes a pore structure that can be classified in terms of three types of pore space. The largest of these porosities is total porosity, which is the volume fraction of the total pore space to the total volume of the host matrix. However, all of these pores do not actively participate in the movement of fluid. Some pores are completely isolated from the interconnected flow pathways; others form dead-end pathways. Consequently, effective porosity is defined as the fraction of the pore space volume through which fluid flow occurs compared to the total matrix volume. A third porosity of interest is diffusional porosity, this porosity is defined as the ratio of the volume of the pores that participate in diffusion to the total matrix volume. It includes the pores through which fluid movement occurs plus the dead-end pores that are assumed not to contribute to fluid flow, but which are assumed to facilitate the diffusion of heat and mass. By definition, its numerical value is between the total and effective porosity values.

1.4.13 Special Geologic Features

In addition to the capability to consider heterogeneity and anisotropy of the porous media, **PORFLOW™** incorporates an option that permits the user to consider planar geologic features such as fractures, faults, clastic dikes and linear features such as boreholes. Their distinctively different length scales and properties distinguish these features from the parent media (soil and/or rock). For example, if one of the three dimensions of a fracture is small relative to the other two, then it behaves essentially as a two-dimensional (planar) element that is embedded in the three-dimensional domain. Similarly, boreholes or other small man-made excavations can be treated as essentially one-dimensional features.

It is possible to treat all of these features as distinct three-dimensional zones. However, due to the different length scales involved, this treatment may make excessive demands on computer resources. An alternate option, to consider features as two- or one-dimensional elements that are embedded within three-dimensional media, is available. The choice of this option greatly reduces the required grid sizes and computational time.

1.4.14 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 intentionally blank.)

CHAPTER 2

GOVERNING EQUATIONS

PORFLOW™ solves a set of coupled transport equations for fluid velocities, pressure, temperature, and concentration of chemical species in multi-phase or multi-fluid, variably saturated, fractured or porous media flow. The number of phases (or fluids) and number of species equations to be solved can easily be 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, except that the Darcy's equations are used in place of the Navier-Stokes 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:

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

where

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

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

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

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

2.2 GOVERNING EQUATIONS FOR FLUID FLOW

2.2.1 Continuity or Mass Balance Equations

The continuity equation for the conservation of mass of a compressible fluid in non-deforming media, in Cartesian tensor notation, is written as:

$$\frac{\partial}{\partial t}(\rho^n S^n \Theta_E) + \frac{\partial}{\partial x_i}(\rho^n V_i^n) = m^n \tag{2.2.1}$$

where:

- ρ^n is the mass density of the n^{th} phase,
- S^n is the saturation of the n^{th} phase,
- Θ_E is the effective porosity of the matrix defined as the component of the pore space, expressed as a fraction of the total matrix volume, through which flow occurs (see Section 1.4.12),
- V_i is the specific discharge or Darcy velocity component of the n^{th} phase in the i^{th} direction, and
- m^n is the rate of mass injection (> 0) or withdrawal (< 0) of the n^{th} phase per unit volume.

The sum of saturation of all phases is equal to unity; that is:

$$\sum_n S^n = 1 \tag{2.2.2}$$

Therefore, for a total of N phases, only N-1 partial differential equations of the form of Equation 2.2.1 need to be solved; saturation for the last phase can be obtained from Equation 2.2.2 as:

$$S^N = 1 - \sum_{n=1}^{N-1} S^n. \tag{2.2.3}$$

2.2.2 Momentum Balance Equation

Darcy's Law adequately expresses the momentum balance in a porous media. For the n^{th} fluid phase, the Darcy flux for variable density conditions, may be written as (Bear, 1972):

$$V_i^n = -K_{ij}^n \left(\frac{\partial P^n}{\partial x_j} - B_j^n \right) . \tag{2.2.4}$$

In this equation:

$$K_{ij}^n = k_{ij} k_r^n \rho^* g / \mu^n , \tag{2.2.5}$$

$$P^n = \frac{p - p^*}{\rho^* g} , \tag{2.2.6}$$

$$B_j^n = R^n \frac{g_j}{g} , \tag{2.2.7}$$

$$R^n = \rho^n / \rho^* , \tag{2.2.8}$$

$$g = \sqrt{\sum_j g_j^2} , \tag{2.2.9}$$

where the superscript n denotes the n^{th} phase and:

- K_{ij} is the hydraulic conductivity tensor,
- P is the hydraulic head at a reference density,
- B_j is the buoyancy vector,
- k_{ij} is the intrinsic permeability tensor,
- $*$ denotes a reference value of the corresponding quantity,
- g is the constant of gravitational acceleration,
- μ is the dynamic viscosity of the fluid,
- x_j is the coordinate distance along the j^{th} direction,
- g_j is the component of gravity in the j^{th} direction,
- p is the thermodynamic pressure, and
- R is the ratio of local fluid density to its reference value.

In general, the pressure head, P^n , can be defined from any arbitrary datum (p^*) because only the gradients of pressure appear in the momentum equation. For multi-phase or partially saturated flow, the pressure head often is defined with atmospheric pressure as the datum. In that case, pressure head is positive if fluid pressure is above atmospheric pressure and negative if below. The pressure head can be combined with the potential head of the fluid to define a total hydraulic head. For a constant-density fluid, the total head, H^n , for the n^{th} phase is defined as:

$$H^n = P^n + \sum_j (x_j^* - x_j) \frac{\rho^n g_j}{\rho^* g} , \tag{2.2.10}$$

where x_j^* are the coordinates of an arbitrary elevation datum.

2.2.3 Pressure Equations

The governing equation for fluid pressure of the n^{th} phase is obtained by substituting Equation 2.2.4 into 2.2.1. After some algebraic manipulation, this equation may be written as:

$$\frac{\partial Q^n}{\partial t} = \frac{\partial}{\partial x_i} [R^n K_{ij}^n (\frac{\partial P^n}{\partial x_j} - B_j^n)] + m_v^n, \quad (2.2.11)$$

where Q^n , the storage term, and m , the rate of addition or loss of fluid mass per unit volume, are respectively given by:

$$Q^n = R^n S^n \theta_E, \quad (2.2.12)$$

$$m_v^n = m^n / \rho^*. \quad (2.2.13)$$

PORFLOW™ permits any of four alternate formulations of the pressure equation to be selected by the user. The first of these is the classical "storage coefficient" formulation which is based on the assumptions that:

$$\theta_E = \theta_E(P^n), \quad (2.2.14)$$

$$\rho^n = \rho(P^n, T), \quad (2.2.15)$$

$$S^n = S^n(P^k, T); \quad k = 1, \dots, n, n+1 \quad (2.2.16)$$

Equation 2.1.11 then transforms into:

$$S_e^n \frac{\partial P^n}{\partial t} = \frac{\partial}{\partial x_i} [R^n K_{ij}^n (\frac{\partial P^n}{\partial x_j} - B_j^n)] + R^n \theta_E (S^n \beta_f^n + \beta_S^n) \frac{\partial T}{\partial t} + m_v^n, \quad (2.2.17)$$

with:

$$S_e^n = R^n S^n \alpha_s^n + R^n \theta_E (S^n \alpha_f^n + \alpha_S^n), \quad (2.2.18)$$

$$\alpha_s^n = \frac{\partial \theta_E}{\partial P^n}, \quad (2.2.19)$$

$$\alpha_f^n = \frac{1}{\rho^n} \left(\frac{\partial \rho^n}{\partial P^n} \right) \Big|_T, \quad (2.2.20)$$

$$\alpha_S^n = \sum_m \left(\frac{\partial S^n}{\partial P^m} \right) \left(\frac{\partial P^m}{\partial P^n} \right), \quad (2.2.21)$$

$$\beta_f^n = - \frac{1}{\rho^n} \left(\frac{\partial \rho^n}{\partial T} \right) \Big|_P, \quad (2.2.22)$$

$$\beta_S^n = \left(\frac{\partial S^n}{\partial T} \right) \Big|_P. \quad (2.2.23)$$

In these equations, for the n^{th} phase denoted by the superscript n :

- S_e is the effective storativity
- α_s is the compressibility of the host solid matrix divided by specific weight of the fluid,
- α_f is the fluid compressibility divided by specific weight of the fluid,
- α_s is the pressure coefficient of phase saturation,
- β_f is the fluid thermal expansion coefficient,
- β_s is the temperature coefficient of phase saturation,
- T is the fluid temperature, and
- m_v is the volumetric rate of mass addition (> 0) or loss (< 0) of phase per unit volume,

The above formulation is used primarily when the storage term in Equation 2.2.11 is small compared to the pressure and source terms. This is often the case for incompressible or slightly compressible flow and for single-phase flow when the speed of pressure propagation is much larger than the flow speed. However, for highly compressible multiphase flows, the storage term can be the dominant term in the equation. In such cases Equation 2.2.11 is replaced by a successive approximation formulation with a "modified storage coefficient" to ensure mass conservation. This formulation is expressed as:

$$S_e^n \frac{\partial \Delta}{\partial t} = \frac{\partial}{\partial x_i} [R^n K_{ij}^n (\frac{\partial P^n}{\partial x_j} - B_j^n)] + m_v^n - \frac{\partial Q^n}{\partial t} . \tag{2.2.24}$$

where Δ is the difference between the current estimate of pressure, Φ , and P^n , the pressure at the new state of the system; that is:

$$\Delta = P^n - \Phi . \tag{2.2.25}$$

Equations 2.2.11 and 2.2.25 are equivalent as $\Phi \rightarrow P^n$ (and, $\Delta \rightarrow 0$). The coefficient S now plays the role of a classical relaxation factor or pseudo-transient storativity coefficient. However, the choice of this coefficient does not affect the final solution so long as $\Phi \rightarrow P^n$. The role played by the storage coefficient in the conventional formulation of the pressure equation is now included implicitly in the storage term, the last term on the right-hand side of Equation 2.2.25. An appropriate choice of S is nonetheless important because it affects the convergence of Φ to P^n . It may be conjectured that the optimum choice for this quantity is that given by Equation 2.2.18.

For multi-phase or partially saturated flow, phase saturations are often strongly non-linear functions of the phase pressures. The storage term can therefore be large in these circumstances. Equation 2.2.24 ensures that mass is conserved unconditionally for such systems irrespective of the nature of the relationship between the storage coefficient and the fluid pressure. Equation 2.2.17, on the other hand, assures mass conservation only if changes in the effective storativity are small compared to changes in the flow due to pressure differences.

2.3 GOVERNING EQUATIONS FOR HEAT TRANSFER

The governing equation for heat transfer is based on the assumption that thermal equilibrium exists between all of the fluid phases and the solid constituents of the porous matrix. Thus, at any given location the solid and fluid constituents are all at the same temperature. The equation for the conservation of thermal energy is then written as:

$$\frac{\partial}{\partial t}(H_e) + \frac{\partial}{\partial x_i}(V_i^e H) = \frac{\partial}{\partial x_i}(\Gamma_{ij}^T \frac{\partial T}{\partial x_j}) + S_T \quad , \quad (2.3.1)$$

where:

- H_e is the effective enthalpy per unit volume of the porous media,
- V is the effective specific discharge component for all the fluid phases in the i^{th} direction,
- H is the total enthalpy of all fluid phases per unit volume,
- Γ_{ij}^T is the effective diffusion (and dispersion) coefficient for heat, and
- S_T is the rate of heat generated (> 0) or dissipated (< 0) per unit volume.

The effective enthalpies of the geologic media and the fluid are given by:

$$H_e = (1 - \theta_T) \rho_s c_s T + \theta_T H \quad , \quad (2.3.2)$$

$$H = \sum_n \rho^n S^n h^n \quad , \quad (2.3.3)$$

where:

- h^n is the enthalpy of the n^{th} phase per unit mass,
- θ_T is the total porosity of the matrix, which is defined as the ratio of the total pore space to the total volume of the matrix (see Section 1.4.12),
- c_s is the specific heat of the solid in dry condition,
- ρ_s is the density of the solid in dry condition, and
- T is the temperature.

The effective enthalpy of the fluid and the porous matrix may be expressed as:

$$H_e = \alpha_h T, \quad (2.3.4)$$

$$H = \beta_h T, \quad (2.3.5)$$

so that:

$$\alpha_h = (1 - \Theta_T) \rho_s c_s + \Theta_T \beta_h, \quad (2.3.6)$$

$$\beta_h = \sum_n \rho^n S^n h^n / T. \quad (2.3.7)$$

The diffusional heat flux consists of three components: thermal conduction in the solid, thermal conduction in the fluid, and hydrodynamic dispersion. The effective coefficient for the total diffusional flux is given by:

$$\Gamma_{ij}^T = (1 - \Theta_T) k_s + \Theta_T \tau_{ij} \sum_n S^n k_f^n + \Theta_E D_{ij} \sum_n S^n \rho^n c_f^n, \quad (2.3.8)$$

where:

- k_s is the thermal conductivity of the solid,
- Γ_{ij} is a constrictivity or tortuosity factor tensor,
- k is the thermal conductivity of the n^{th} phase,
- D_{ij} is the hydrodynamic dispersion tensor (discussed further in Section 3.9) and
- c is the specific heat of the n^{th} phase.

Combination of the above equations leads to the final form of the governing equation for temperature:

$$\frac{\partial}{\partial t} (\alpha_h T) + \frac{\partial}{\partial x_i} (\beta_h V_i T) = \frac{\partial}{\partial x_i} (\Gamma_{ij}^T \frac{\partial T}{\partial x_j}) + S_T. \quad (2.3.9)$$

2.4 GOVERNING EQUATIONS FOR MASS TRANSPORT IN FLUID

To derive the governing mass transport equation, it is assumed that transport of chemical species is limited to only one phase, that is, the n^{th} phase of the fluid system. The equation for the conservation of mass is then written as:

$$\frac{\partial}{\partial t}(\rho^n C_e^k) + \frac{\partial}{\partial x_i}(\rho^n V_i C^k) = \frac{\partial}{\partial x_i}(\rho^n \Gamma_{ij} C \frac{\partial C^k}{\partial x_j}) + S_C^k - \rho^n \gamma^k, \quad (2.4.1)$$

where:

- k, n in superscript, denote the k^{th} species and the n^{th} fluid phase, respectively,
- C_e is the effective concentration of the species in the porous matrix per unit mass of the n^{th} phase fluid,
- C^k is the concentration of the k^{th} species in the fluid per unit mass of the n^{th} phase fluid,
- Γ_{ij} is the effective diffusion (and dispersion) coefficient for the species,
- S_C is the source of the k^{th} species, and
- γ^k is the net decay of the k^{th} species.

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

$$\gamma^k = \lambda^k C_e^k - \sum_m \sigma^{mk} \lambda^m C_e^m, \quad (2.4.2)$$

where λ^k is the decay rate for the k^{th} species, and σ^{mk} is the fraction of decay of the m^{th} species which generates the k^{th} species.

The effective mass concentration in the porous matrix is related to that in the fluid and solid phases by:

$$C_e^k = \varphi_E C^k + (S^n - \varphi_E) C_s^k, \quad (2.4.3)$$

$$\varphi_E = S^n \Theta_E, \quad (2.4.4)$$

where:

- φ_E is the effective saturated porosity of the host matrix for the n^{th} phase,
- C_s is the concentration of the transported species in the solid phase,
- S^n is the saturation of the n^{th} phase, and
- Θ_D is the diffusional porosity of the host matrix (see Section 1.4.12).

We now assume that a chemical equilibrium exists between the fluid and the solid constituents of the porous matrix and that the concentrations in the fluid and the solid are related through a linear Freundlich isotherm (Freeze and Cherry 1979). With k_d as the sorption or partition coefficient for the species, the concentrations in the solid and the fluid are related by:

$$C_s^k = \rho_s k_d^k C^k. \quad (2.4.5)$$

Equations 2.4.3 and 2.4.5 combine to yield:

$$C_e^k = \varphi_E R_D^k C^k, \quad (2.4.6)$$

$$R_D^k = 1 + \rho_s k_d^k \frac{1 - \Theta_E}{\Theta_E}, \quad (2.4.7)$$

where, R , is the retardation coefficient. The decay term can now be written as:

$$\gamma^k = \varphi_E R_D^k \lambda^k C^k - \sum_m \varphi_E R_D^m \sigma^{mk} \lambda^m C^m. \quad (2.4.8)$$

Two options are provided to compute the effective diffusion coefficient for mass. The first option is based on the assumption that Γ is composed of two components: that due to molecular diffusion and that due to hydrodynamic dispersion; consequently:

$$\Gamma_{ij}^C = \varphi_D \tau_{ij} D_M + \varphi_E D_{ij}, \quad (2.4.9)$$

$$\varphi_D = S^n \Theta_D, \quad (2.4.10)$$

where D_M is the molecular diffusion coefficient, τ_{ij} is the tortuosity factor, and φ_D is the diffusional saturated porosity for the n^{th} fluid phase. The second option, following Kemper and van Schaik (1966), provides for an exponential dependence of Γ_{ij} on φ_D in the form:

$$\Gamma_{ij}^C = D_M \alpha_1 \exp(\alpha_2 \varphi_D). \quad (2.4.11)$$

The final governing equation for chemical species is now obtained as:

$$\frac{\partial}{\partial t} (R^n \varphi_E R_D^k C^k) + \frac{\partial}{\partial x_i} (R^n V_i C^k) = \frac{\partial}{\partial x_i} (R^n \Gamma_{ij}^C \frac{\partial C^k}{\partial x_j}) + R^n (S_C^k - \gamma^k). \quad (2.4.12)$$

2.5 GOVERNING EQUATIONS FOR MASS TRANSPORT IN SOLID

The governing mass transport equation in the solid phase is similar to that in the fluid phase except that the convective term is zero and the accumulation term pertains only to the solid phase of the porous matrix. This equation for the conservation of mass is written as:

$$\frac{\partial}{\partial t} [(1 - \Theta_E) \rho_s C^k] = \frac{\partial}{\partial x_i} \left(\rho_s \Gamma_{ij} \frac{\partial C^k}{\partial x_j} \right) + S_C^k - (1 - \Theta_E) \rho_s \gamma^k \tag{2.5.1}$$

where:

- k in superscript, denotes the kth species,
- ρ_s is the density of the solid,
- Γ_{ij} is the effective diffusion coefficient for the species,
- S_C is the source of the kth species, and
- γ^k is the net decay of the kth species which is given by:

$$\gamma^k = \lambda^k C^k - \sum_m \sigma^{mk} \lambda^m C^m, \tag{2.5.2}$$

where γ^k is the decay rate for the kth species, and σ^{mk} is the fraction of decay of the mth species which generates the kth species.

(This page left intentionally blank.)

CHAPTER 3

AUXILIARY EQUATIONS

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

3.1 FLUID DENSITY

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

$$\rho = \rho^* \left[\frac{T_c - T}{T_c - T^*} \right]^{a_1} \quad (3.1.1)$$

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

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

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

where:

- ρ^* is the reference density for the fluid,
- T_c is the critical temperature of the fluid,
- T^* is the reference temperature,
- a_1, a_2, a_3 are empirical constants,
- C^* is the reference mass concentration,
- p^* is the reference datum for thermodynamic pressure,
- R_u is the universal gas constant,
- T_a is a constant to convert temperature to absolute units,
- m_j is mass fraction of the j^{th} chemical species in the gas, and
- M_j is the molecular weight of the j^{th} species.

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

3.2 FLUID VISCOSITY

The changes in fluid viscosity affect the hydraulic conductivity of the porous matrix. In general, the viscosity of a fluid is a much stronger function of temperature than of pressure. The effect of pressure is therefore ignored. Both analytic and arbitrary user-defined functions for calculating fluid viscosity can be used. The currently available analytic functions are:

$$\mu = \mu_0 \exp\left[\frac{a_1}{T + T_a}\right], \quad (3.2.1)$$

$$\mu = \mu^* [1 + a_1 (T^* - T) + a_2 (T^* - T)^2 + a_3 (T^* - T)^3], \quad (3.2.2)$$

where μ^* is the fluid viscosity at reference temperature, T^* , and a_1 through a_3 are empirically determined constants. For water, Perry and Chilton (1973, p.3-246) recommend Equation 3.2.1. With μ_0 and a_1 equal to 6.4×10^{-6} kg/(m s) and 1436 K, respectively, the calculated values differ from the reference values by less than 8% between 25 and 250 °C, by less than 15% between 20° and 350 °C and, by 31% at 0 °C.

In PORFLOW™, it is the viscosity ratio rather than the viscosity that is required to account for the influence of viscosity variation on hydraulic conductivity (see Equation 3.5.1 below). The in this ratio can be considerably decreased from the above values by selecting a reference temperature, T^* , that is toward the middle of the expected temperature range. Equation 3.2.1 can be written in terms of μ^* as:

$$\mu = \mu^* \exp\left[\frac{a_1}{T + T_a} - \frac{a_1}{T^* + T_a}\right]. \quad (3.2.3)$$

3.3 CHARACTERISTIC SATURATION RELATIONS

PORFLOW™ can simulate multi-phase flow systems. For example, the flow may consist of one (e.g., water or gas), two (e.g., water-air, water-gas, water-ice) or three (e.g., water-oil-gas) phases. The phase change relationships that determine S^n are based on the Clapeyron relation:

$$S^n = S^n(\psi), \tag{3.3.1}$$

$$\psi = A P + B T, \tag{3.3.2}$$

where ψ may be interpreted as a mean curvature of the phase interface, P and T are the characteristic pressure and temperature of the system, and A and B are empirical constants. However, the Clapeyron phase change relations are applicable in a more general sense and the coefficients need not be interpreted on the basis of surface curvature.

The characteristic saturation relationship, often termed the "soil moisture characteristic", is a basic soil (or rock) property for multi-phase flow and is usually based on empirical measurements. Furthermore, these relationships are generally expressed in terms of cumulative relative saturation, \hat{S}^n , rather than as the absolute value of the saturation, S^n .

For a three-phase system, cumulative relative saturations are given by:

$$\hat{S}^1 = \frac{S^1 - S_r}{1 - S_r - S_g}, \tag{3.3.3}$$

$$\hat{S}^2 = \frac{S^1 + S^2 - S_r}{1 - S_r - S_g}, \tag{3.3.4}$$

$$\hat{S}^3 = 1, \tag{3.3.5}$$

where S_r is the "residual" saturation of the primary fluid and S_g is the residual saturation of the gaseous phase. For a two-phase system, \hat{S}^1 is given by Equation 3.3.3 and \hat{S}^2 is unity.

Several methods for computing phase saturations are provided. Many practical applications, especially problems involving ground water hydrology and environmental pollution, involve isothermal or near-isothermal fluid systems. In such cases, empirical evidence suggests that phase saturation be determined primarily by the capillary pressure difference between adjoining phases. In such cases, S^n is specified by empirical relations between S^n and ψ^n , where ψ^n is the capillary pressure between the n^{th} phase and the adjoining $n+1^{\text{th}}$ phase; that is:

$$\psi^n = P^{n+1} - P^n. \quad (3.3.6)$$

The quantity ψ^n is always defined so that "n" refers to the fluid with higher "wettability" (e.g., water) compared to the fluid of lower "wettability", denoted by "n+1" (e.g., air). Therefore, if the capillary pressure, ψ^n , is positive, then both fluids are simultaneously present in that pore space (at a macroscopic level); otherwise, the pore space is saturated with the "wetter" fluid. The capillary pressure may also be modified by changes in temperature and the presence of chemical species. PORFLOW™ addresses these influences by means of the generic relation:

$$\psi = \psi_0 \left[\frac{T_c - T^*}{T_c - T} \right]^\alpha (1 + \alpha_c C + \beta_c C^2), \quad (3.3.7)$$

where ψ is the value actually employed in the constitutive relations, ψ_0 is the base value computed from the pressure equations, T_c and T^* are reference temperatures and C is the concentration of a chemical species.

For the (S- ψ) characteristic relations, several options are provided. The most general option is that of a tabular relation between S and ψ . Other options are based on specific analytic formulae. The framework of the software is sufficiently general that any functional relationship between S and ψ can be accommodated. The built-in options include the Brooks and Corey (1966), van Genuchten (1978), exponential, logarithmic and polynomial formulations. These are described below. For reasons of simplicity, the phase superscript, n , is omitted from these relations.

The Brooks and Corey (1966) and the van Genuchten (1978) formulations are respectively given by:

$$\begin{aligned} \hat{S} &= (\psi_a / \psi)^\lambda; \quad \psi > \psi_a; \\ &= 1; \quad \psi \leq \psi_a \end{aligned} \quad (3.3.8)$$

$$\begin{aligned} \hat{S} &= \frac{1}{[1 + (\alpha \psi)^N]^M}; \quad \psi > 0; \\ &= 1; \quad \psi \leq 0, \end{aligned} \quad (3.3.9)$$

where ψ_a , λ , α , M and N are empirical constants. The ψ_a is the so-called air-entry pressure head, and α of the van Genuchten formulation is equivalent to $1/\psi_a$ of the Brooks and Corey formulation. The two formulations become identical as $\psi \gg \psi_a$. In this case their exponents are related by:

$$\lambda = MN. \quad (3.3.10)$$

The exponential and the logarithmic relations are respectively given by:

$$\begin{aligned} \hat{S} &= \frac{C}{C + [\exp(\alpha \psi)]^{N-1}}; \quad \psi > 0; \\ &= 1; \quad \psi \leq 0, \end{aligned} \quad (3.3.11)$$

$$\hat{S} = \frac{C}{C + [\ln(\alpha \psi)]^N}; \psi > 1/\alpha;$$

$$= 1; \psi \leq 1/\alpha,$$
(3.3.12)

where N, α and C are empirical constants.

In the above relations, S is expressed as an explicit function of ψ . The polynomial relation given below is an inverse relation, with ψ expressed as an explicit function of S:

$$\psi = A + B \hat{S} + C \hat{S}^2 + D \hat{S}^3 + E \hat{S}^4,$$
(3.3.13)

where A, B, C, D and E are empirical constants.

In contrast to capillary-pressure-driven systems, the temperature, rather than pressure, primarily governs the phase saturation in water-ice systems. The characteristic built into PORFLOW™ for phase saturation employs the constitutive relation given by Wheeler (1973) that is suitable for medium-to-coarse grained soils. In this characteristic relation:

$$\hat{S} = \frac{1}{[1 + (T_F - T)/T_D]^\lambda}; T \leq T_F;$$

$$= 0; T > T_F,$$
(3.3.14)

where:

- \hat{S} is the relative saturation of the liquid phase,
- T_F is the freezing temperature,
- T_D is a characteristic delay temperature for freezing, and
- λ is an empirical power-law exponent.

3.4 EFFECTIVE STORATIVITY

The effective storativity, S_e of Equation 2.2.18, is given by:

$$S_e^n = R^n S^n \alpha_s^n + R^n \Theta_E (S^n \alpha_f^n + \alpha_s^n). \quad (3.4.1)$$

For a porous matrix, α_s^n , the compressibility of the matrix is assumed to be a constant that is specified from empirical data. R^n and α_f^n , are obtained from the fluid density relations. For a liquid, fluid compressibility, α_f^n , is generally quite small and is specified by the user as input. For a gas, the fluid compressibility can be calculated from the perfect gas law as:

$$\alpha_f^n = 1 / P^n. \quad (3.4.2)$$

For a confined one-phase flow system, saturation is unity; therefore, α_s is uniformly zero. For unconfined flow, α is equal to unity when dry soil becomes saturated or wet soil drains; Otherwise, it is zero. For a multi-phase system, α is determined from the saturation characteristic according to Equation 2.2.21. For tabular input of moisture characteristic data, this coefficient is computed numerically; otherwise, it is obtained by formal differentiation of the specified analytic formula.

3.5 HYDRAULIC CONDUCTIVITY

The hydraulic conductivity tensor, K_{ij} of Equation 2.2.5 may be written as:

$$K_{ij}^n = K_{ij}^* k_r^n \mu^* / \mu^n, \quad (3.5.1)$$

$$K_{ij}^* = k_{ij} \rho^* g / \mu^* \quad (3.5.2)$$

where k is the relative permeability for the n^{th} phase, and K_j are the reference values of the hydraulic conductivity tensor for a reference fluid (usually water) of density ρ^* and viscosity μ^* . The K_j are specified from field data for a given soil or rock formation, or are calculated from the intrinsic permeability of the porous matrix according to Equation 3.5.2. The viscosity ratio is calculated from one of the viscosity relations given earlier.

For a single-phase system, k_r is equal to unity. For a multi-phase system, k_r may be either specified from empirical data or it may be obtained from the saturation characteristic relations by one of the available predictive models. The the resulting expression for the Mualem (1976) or Burdine (1953) predictive models, for the van Genuchten relation, is given by:

$$k_r^n = [\hat{S}^n - \hat{S}^{n-1}]^A \{ [1 - (\hat{S}^{n-1})^{1/M}]^M - [1 - (\hat{S}^n)^{1/M}]^M \}^B \quad (3.5.3)$$

.For the Brooks and Corey (1966) relation, this expression is:

$$k_r^n = [\hat{S}^n - \hat{S}^{n-1}]^A [(\hat{S}^{n-1})^{B+2/\lambda} - (\hat{S}^n)^{B+2/\lambda}], \quad (3.5.4)$$

Exponents A and B are, respectively, 1/2 and 2 for the Mualem equation (1976) and, 2 and 1 for the Burdine (1953) equation. The superscript n takes the values of 1, 2, and 3, with:

$$\hat{S}^0 = 0. \quad (3.5.5)$$

In addition to these predictive models, several analytic relations to compute the relative conductivity from either phase saturations or capillary pressure are provided. The k_r versus S formulations, for any of the fluid phases, are:

$$k_r = A [B + \hat{S}]^N + C, \quad (3.5.6)$$

$$k_r = A + B \hat{S} + C \hat{S}^2 + D \hat{S}^3 + E \hat{S}^4. \quad (3.5.7)$$

The built-in relations for k_r , in terms of ψ include exponential, logarithmic and inverse- power-law formulations. These are respectively given by:

$$k_r = \frac{C}{C + [\exp(\alpha \psi)]^N - 1}; \quad \psi > 0 ;$$

$$= 1; \quad \psi \leq 0, \quad (3.5.8)$$

$$k_r = \frac{C}{C + [\ln(\alpha \psi)]^N}; \quad \psi > 1/\alpha;$$

$$= 1 \quad ; \quad \psi \leq 1/\alpha, \quad (3.5.9)$$

$$k_r = \frac{C}{C + [\alpha \psi]^N}; \psi > 0; \quad (3.5.10)$$

$$= 1 \quad ; \psi \leq 0.$$

For two-phase flow, the relative hydraulic conductivity relations can be used in a complimentary, mode where the k_r of the 2nd phase is given in terms of the k_r of the 1st phase by:

$$k_r^2 = 1 - k_r^1. \quad (3.5.11)$$

For example, if Equation 3.5.6 is used for k_r of the 1st phase, then:

$$k_r^2 = 1 - A [B + \hat{S}^1]^N - C. \quad (3.5.12)$$

Furthermore, from Equation 2.2.3, this equation can be written as:

$$k_r^2 = 1 - A [B + (1 - \hat{S}^2)]^N - C. \quad (3.5.13)$$

3.6 FLUID LATENT AND SPECIFIC HEATS

The enthalpy of the n^{th} fluid phase, with h as its enthalpy of formation and c as its specific heat, is written as:

$$h^n = h_0^n + \int_T c_p^n dT. \quad (3.6.1)$$

The heat of formation for all of the fluid phases, except for the component that changes phase, is taken as zero. It is also assumed that for all components in solid or liquid form, the specific heat is constant. Thus, the enthalpy for these components is given by:

$$h^n = c_p^n T. \quad (3.6.2)$$

For the component that changes phase, the enthalpy-temperature relation for the vapor phase is defined by:

$$h_v = c_0 + c_1 T + c_2 T^2 + c_3 T^3, \quad (3.6.3)$$

where c_0 through c_3 are empirical constants and T is in absolute units. The default set of values are those given by Bixler et al. (1985) for evaporation of water, with $c_0 = 3.9202 \times 10^6$, $c_1 = -2441.00$ and $c_2 = c_3 = 0$. Another set of values which are valid for a wider temperature range are (Vargaftik, 1975) $c_0 = 3.9202 \times 10^6$, $c_1 = 3681.00$, $c_2 = -11.76$ and, $c_3 = 0.01772$. For these constants, T is in K and h_v in J/kg.

3.7 VAPOR FORMATION EQUATION

PORFLOW™ provides both equilibrium and non-equilibrium options for vapor formation. In the equilibrium mode, the rate of vapor formation is automatically computed from the equilibrium value of the vapor pressure and the thermal energy available for heat of formation of the vapor. In the non-equilibrium mode, the vapor formation equation is based on mass-transfer equilibrium principal similar to that implemented by Bixler et al. (1985). The net transfer rate is given by:

$$m_v = C_m \Theta_E S^\alpha \frac{p_s - p_v}{p_g}, \quad (3.7.1)$$

where:

- m_v is the rate of evaporation or condensation,
- C_m is an empirical constant that defines the characteristic frequency of phase change,
- S is the saturation fraction of the fluid,
- α is an empirical exponent,
- p_s is the saturation pressure of the vapor at equilibrium,
- p_v is the actual thermodynamic pressure of the vapor, and
- p_g is the thermodynamic pressure of the gas phase.

3.8 EQUILIBRIUM VAPOR PRESSURE

The built-in equilibrium vapor-pressure formula is:

$$p_s = a_0 \exp \left[a_1 - \frac{a_2}{T - a_3} \right], \tag{3.8.1}$$

where a_0 through a_3 are empirical constants. With $a_0 = 131.57894$, $a_1 = 18.3443$, $a_2 = 3841.1954$, $a_3 = 45$, and T is in Kelvin, the values of saturation pressure for water are in good agreement with the experimental data in the 0°C to 350°C, range as shown in Table 3.8.1.

TABLE 3.8.1: VAPOR PRESSURE FOR WATER

T		p_s (pascals x 10^5)	
°C	K	Equation	Steam Tables
0	273.15	0.00594	0.00611
20	293.15	0.02309	0.02337
50	323.15	0.12260	0.12335
100	393.15	1.00539	1.0133
150	425.15	4.72619	4.7597
170	443.15	7.87248	7.9203
200	473.15	15.47723	15.5506
250	523.15	39.54853	39.776
300	573.15	84.61033	85.92
350	623.15	158.7037	165.37

3.9 HYDRODYNAMIC DISPERSION

The general form of the hydrodynamic dispersion tensor as given by Scheidegger (1961) is:

$$D_{ij} = \alpha_T \delta_{ij} \underline{v} + (\alpha_L - \alpha_T) \frac{|v_i v_j|}{\underline{v}}, \quad (3.9.1)$$

$$\underline{v} = (v_i v_i)^{1/2}, \quad (3.9.2)$$

where α_L and α_T are, respectively, the longitudinal and transverse dispersivities for the porous matrix, δ_{ij} is the Kronecker delta function and, \underline{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 by:

$$v_i = V_i / \theta_E. \quad (3.9.3)$$

The current versions of **PORFLOW™** assume that the principal axes of the dispersion tensor are aligned with the coordinate directions. Consequently, all off-diagonal components are zero and the diagonal components, in (x, y, z) notation, are given by:

$$D_x = [\alpha_L U^2 + \alpha_T (V^2 + W^2)] / (\theta_E \underline{v}), \quad (3.9.4)$$

$$D_y = [\alpha_L V^2 + \alpha_T (W^2 + U^2)] / (\theta_E \underline{v}), \quad (3.9.5)$$

$$D_z = [\alpha_L W^2 + \alpha_T (U^2 + V^2)] / (\theta_E \underline{v}), \quad (3.9.6)$$

and

$$\underline{v} = (U^2 + V^2 + W^2)^{1/2}. \quad (3.9.7)$$

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. Thus, for example, if fluid is being injected into a region, the source term for that region is positive. On the other hand, if fluid is being 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:

$$S_F = S_F(\xi), \quad (3.10.1)$$

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

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

$$S_F = q_m F_{inj}, \quad (3.10.2)$$

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

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

$$S_C^k = f_S(C_S^k - C^k), \quad (3.10.3)$$

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

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

$$\frac{\partial C}{\partial t} = -\lambda_k C, \quad (3.10.4)$$

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

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

A fraction of the decay of the k^{th} species may generate the next species in the chain; this fraction is denoted by σ^{mk} and, along with $t_{1/2}$, 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:

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

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

3.11.2 Dirichlet Boundary Condition

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

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

3.11.3 Neumann Boundary Condition

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

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

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

3.11.4 Mixed Boundary Condition

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

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

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

3.12 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.

By default, initial values of all variables, except saturation of the first fluid phase, are taken to be zero.

3.13 FIELD VARIABLE INTERRELATIONSHIPS

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

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

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

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 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 intervals, (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 employed in **PORFLOW™** are functions of the discretization process. These numerical features are discussed in this chapter.

4.1 THE GRID ARRANGEMENT

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

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

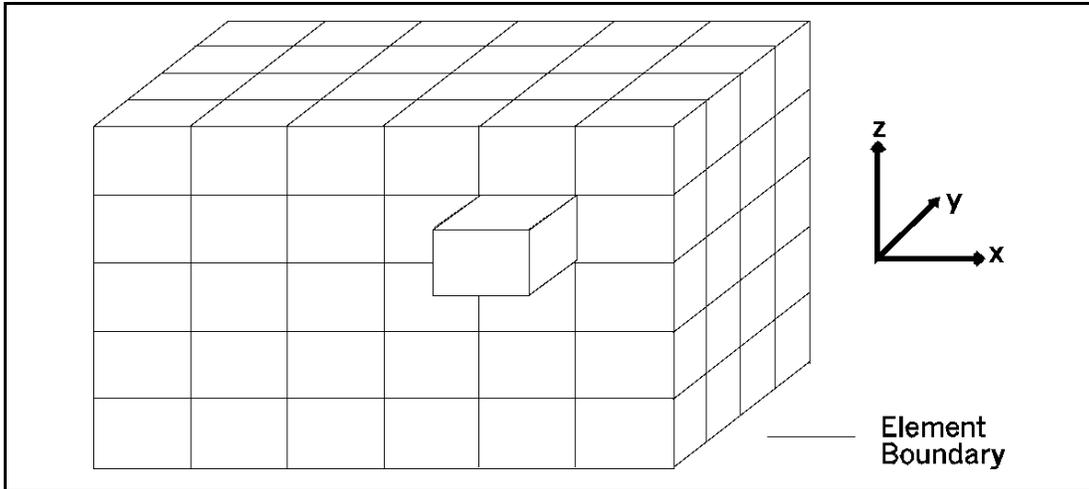


FIGURE 4.1.1: CARTESIAN GRID ARRANGEMENT

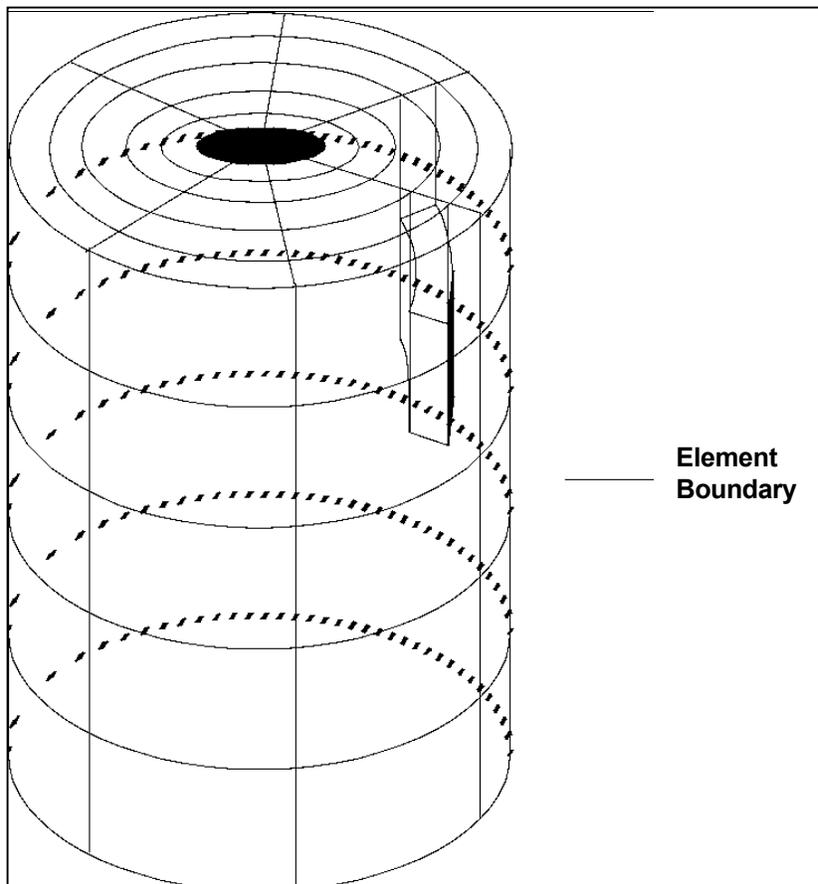


FIGURE 4.1.2: CYLINDRICAL GRID ARRANGEMENT

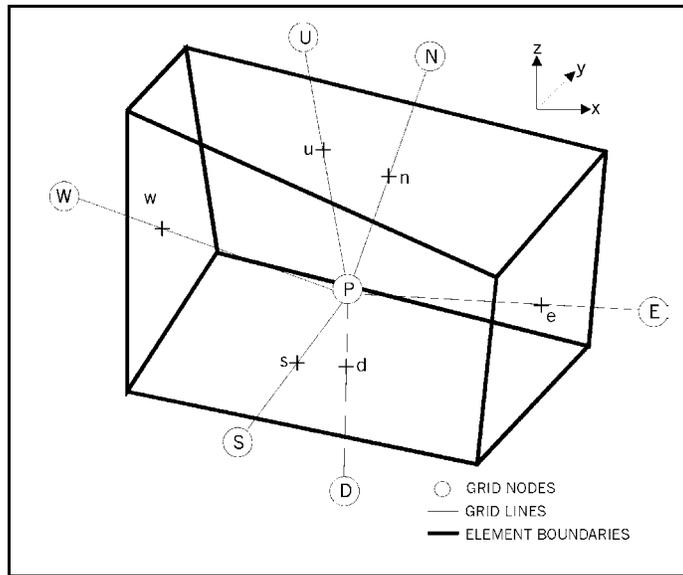


FIGURE 4.1.3: TYPICAL GRID ELEMENT IN CARTESIAN COORDINATES

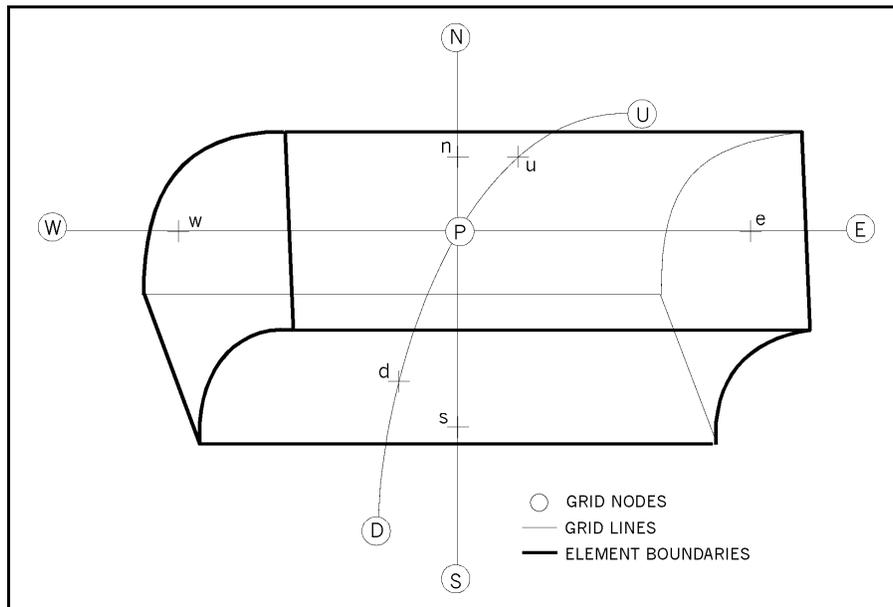


FIGURE 4.1.4: TYPICAL GRID ELEMENT IN CYLINDRICAL COORDINATES

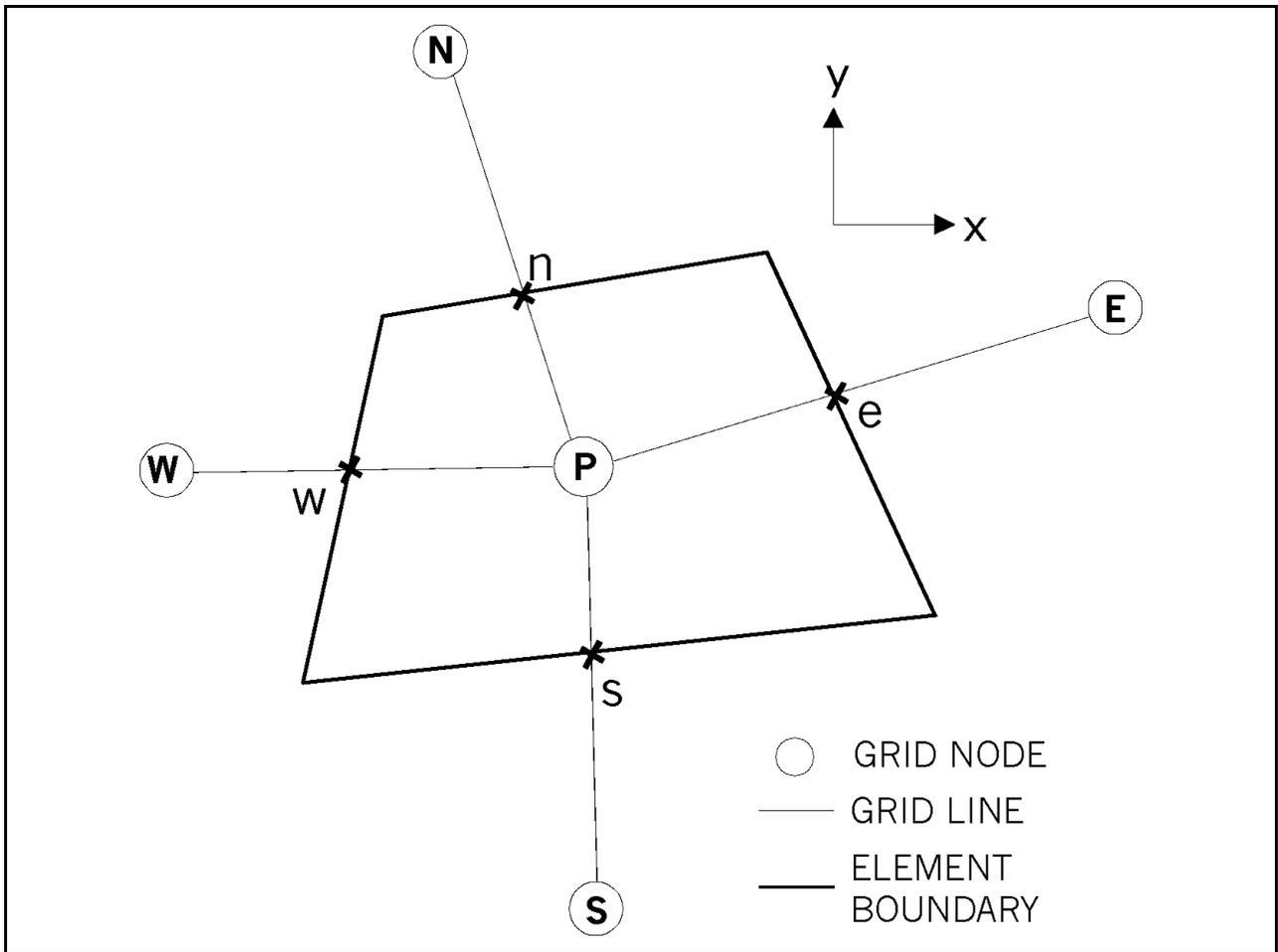


FIGURE 4.1.5: TYPICAL GRID ELEMENT IN X-Y PLANE

4.2 DISCRETIZATION SCHEME

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

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

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

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

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:

$$\begin{aligned}
 & A_P (F_P^{k+1} - F_P^k) + \sum_K A_{PK} [\Theta (F_P^{k+1} - F_K^{k+1}) + (1-\Theta) (F_P^{k+1} - F_K^k)] \\
 & = \delta V S_{FP} - s_{FP} [\Theta F_P^{k+1} + (1-\Theta) F_P^k]
 \end{aligned}
 \tag{4.2.2}$$

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

4.3 SOLUTION METHOD

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

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

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

4.4 NUMERICAL STABILITY AND ACCURACY

4.4.1 General Considerations

The selected numerical schemes and solution methods represent the best options currently available for solving a wide variety of flow and transport problems. In general, they provide accurate, economical and numerically stable solutions. However, as in all numerical 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 PORFLOW™ imply the existence of several characteristic temporal and spatial scales. For efficient, accurate and stable numerical solution, these scales must be duly considered in selecting the grid distribution and time step. The influence of the time step is, of course, limited to transient solution process; it plays no role in steady state simulations. These issues are discussed in more detail below.

4.4.2 Time Scale of Pressure Propagation

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

$$\delta t_{\text{press}} = S_e \delta L^2 / (2K), \quad (4.4.1)$$

where S_e , δL , and K are, respectively, the representative values of effective storativity, grid interval and hydraulic conductivity.

This characteristic time scale is important for transient flow problems. All pressure disturbances propagate across a grid interval of length δL 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.

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

4.4.3 Time Scale of Diffusion

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

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

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

This time scale is similar to that for pressure propagation except that α_e and Γ_e play the roles of the S_e and K , respectively. The thermal and mass transport equations contain diffusion terms. The general implications of this time scale are similar to those of the pressure propagation scale. The time step that is employed should be kept within an order-of-magnitude of the value given by Equation 4.4.2. For the explicit solution method, the time step employed must not, for reasons of stability, exceed the value given by this equation.

4.4.4 Time Scale of Convection

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

$$\delta t_{\text{conv}} = \delta L / U. \quad (4.4.3)$$

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

$$Co = \delta t / \delta t_{\text{conv}} = U \delta t / \delta L, \quad (4.4.4)$$

where δt is a representative time step for numerical solution.

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

4.4.5 Other Pertinent Time Scales

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

4.4.6 Grid Peclet Number

The grid Peclet number (Equation 4.2.1), plays an important role in the numerical stability and accuracy of a numerical scheme if both the convective and diffusive terms are not zero. Because the pressure equation contains no convective terms, the Peclet number is important only for the temperature and concentration equations. 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 PORFLOW™ provides three alternatives. These 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 solution of the temperature equation. The characteristic velocity for this equation is the Darcy velocity, which is typically small and allows large time steps according to the criterion above. For the concentration equation on the other hand, the characteristic velocity is the pore velocity. Therefore, for the concentration equation, it may be advisable to use one of the other options.

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:

$$A_{ij} \phi_{ij} = b_j \quad (4.5.1)$$

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

$$\phi_i^k = T_{ij} \phi_i^{k-1} + c_i \quad (4.5.2)$$

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

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

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

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

$$R_3 = \left| \phi_i^k - \phi_i^{k-1} \right| \quad (4.5.5)$$

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

$$R_{\max} = \frac{1}{N} \sum_{n=1}^N (R_n) \leq \varepsilon \quad (4.5.6)$$

$$R_{\max} = \max (R_n) \leq \varepsilon, \quad i = 1, 2, 3, \dots, N, \quad (4.5.7)$$

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

(This page left intentionally blank.)

CHAPTER 5

SPECIFICATION OF 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 geometric elements termed "material types" or "sub-domains". For complex problems, much of the input specification is made by sub-dividing the domain of interest into several material types or sub-domains that differ from the others in some respect. This chapter outlines the manner in which the computational domain is defined and later used to accommodate the disparate components of the geometry, physics and the computational requirements of a problem.

5.1 SPECIFICATION OF PROBLEM GEOMETRY

The problem geometry is defined by a set of Cartesian (x,y,z) or cylindrical (x,r,θ) coordinates for the vertices of the elements or of the grid nodes. The domain of interest is specified in terms of material "types" or "zones". Each material type has distinctive physical properties or other characteristics that distinguish it from other sub-regions. Distinguishing features include for example, different rock or soil types or layers, fractures, or boreholes..

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

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

5.2 SPECIFICATION OF SPATIAL AND TEMPORAL GRID

5.2.1 *Basic Considerations*

Considerations that influence the design of a computational grid include:

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

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

5.2.2 *Computer Resources*

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

5.2.3 *Numerical Stability and Accuracy*

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

5.2.4 *Scale of Heterogeneity*

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

5.2.5 *Scale of Physical Resolution*

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

5.2.6 *Sources and Sinks*

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

5.2.7 Boundary Conditions

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

5.2.8 Input and Output Requirements

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

5.2.9 Planar and Linear Features

Geologic or man-made features such as fractures and clastic dikes, and bore holes, wells, and tunnels, are distinguishable from the geologic continuum by distinctive contrasts in their physical properties and scales. They may be represented as distinct zones if their geometric dimensions can be adequately resolved by the grid elements of PORFLOW™. However, the fracture width or the borehole diameter is often too small to be economically resolved by grid elements. Fractures and clastic dikes can be considered to be planar features because the dimension orthogonal to the plane defined by their dip and strike is considerably smaller than their dimension in that plane. Consequently, fluid flow, heat transfer, and mass transport can be assumed to occur in only two dimensions in these features. Similarly, only the longitudinal dimension needs to be considered for boreholes, wells, and tunnels.

The geometry and properties of the planar and linear features are identified independently of the remaining input. A restriction is placed on the geometry of these discrete structures; they must begin and end at the nodes of the grid. These planar and linear features are then automatically embedded in the porous media. Although these features are embedded within the porous media and communicate with it by means of the applicable governing equations, PORFLOW™ tracks the flow and transport in them separately from that in the porous media.

(This page left intentionally blank.)

CHAPTER 6

INPUT AND OUTPUT OPTIONS

PORFLOW™ 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 **PORFLOW™**.

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 PORFLOW™ AND THEIR FUNCTIONS

NO.	KEYWORD	INPUT FUNCTION
1	ADAP tive	Refine the specified grid by to Adapt to the computational solution
2	ALLO cate	Allocate memory for field variables and storage arrays for user input
3	BAN NeR	Print user & program identification to output file
4	BOUN dary	Override built-in boundary conditions
5	CAP illary	Effect of temperature and chemical species on capillary pressure
6	CLO se	Close specified output device immediately
7	CON duction	Specify conduction coefficient and options
8	CON nectivity	To specify element and vertex connectivity for unstructured grid
9	CON vergence	Specify convergence criterion
10	COOR dinate	Specify grid coordinates
11	COR relation	Computation of correlations between variables and locations
12	DATU m	Specify reference datum level for hydraulic head
13	DEBU g	Specify debug options
14	DECA y	Specify decay constants for chemical species
15	DEF ine	Specify value of a symbolic variable
16	DENS ity	Select fluid mass density options
17	DIAG nostic	Diagnostic output options
18	DIFF usion	Specify diffusion coefficient and options
19	DIS able	Disable certain built-in default options
20	DIST ribution	Specify distribution coefficient and options
21	EBU	Select Eddy-break-up limiting options for chemical reactions
22	ELSE	Define the “ELSE” part of the IF-ELSE-ENDIF construct
23	END	End of a problem
24	ENDIF	Terminate the IF-ELSE-ENDIF construct
25	FILE	Open or close selected I/O device
26	FIX	Fixed pressure, temperature or concentration nodes
27	FLOW	Specify mass flux entering at boundary
28	FLUI d	Specify thermal and transport properties of the fluid
29	FLUX	Compute flux of fluid, heat or chemical species crossing a sub-domain
30	FOR	Specify zone designation for property input
31	GAS	Specify gas constant and parameters
32	GEOM etry	Specify or modify problem geometry

..... Table 6.1.1 continued

TABLE 6.1.1: KEYWORDS OF PORFLOW™ AND THEIR FUNCTIONS (Continued)

NO.	KEYWORD	INPUT FUNCTION
33	GRAVity	Specify constants of gravitational acceleration
34	GRID	Number of grid nodes in the x, y and z directions
35	HISTory	Provide time history output for selected nodes or sub-domains
36	HYDRaulic	Hydraulic properties of porous matrix
37	IF	To initiate the IF-ELSE-ENDIF construct
38	INCLude	To include a file at run time in the input data stream
39	INITial	Initial conditions for state variables
40	INLEt	To specify an inflow boundary for the domain of computation
41	INTEgration	Index for selection of integration profile
42	LIMIT	Set upper and lower limits for dependent variables
43	LOCAt	Specify location of sub-regions or boundaries
44	MATERial	Specify material types and properties
45	MATRix	Specify Options for solution of matrix of equations
46	MULTiphase	Specify multi-phase or multi-fluid hydraulic characteristics
47	OPEN	To specify an open boundary through which fluid may enter or leave
48	OPTIon	To select alternative options for computation
49	OUTLet	To specify an outflow boundary for the domain of computation
50	OUTPut	Frequency and extent of tabular output
51	PAUSe	Cause a temporary pause in processing
52	PERIodic	Specify periodic boundary conditions
53	PHASe	Specify phase change mode and constants
54	PRECipitate	Specify precipitation reaction from fluid to solid phase
55	PRINT	Print flow rate, sources and statistical measures of variables
56	PROBLEM	Specify general nature and type of problem
57	PROPerty	Option for mode of property specification
58	QUIT	Terminate solution process
59	REACTION	To specify Arrhenius reaction rate kinetics for chemical species
60	READ	Read initial conditions from archive file
61	REFERence	Reference values for key variables
62	REGeneration	Specify regeneration rate for chemical decay chain
63	RELAX	Relaxation factors for governing variables
64	RENAME	Allows renaming of output variables listed in Table 6.8.1

..... Table 6.1.1 continued

TABLE 6.1.1: KEYWORDS OF PORFLOW™ AND THEIR FUNCTIONS (Continued)

NO.	KEYWORD	INPUT FUNCTION
65	RETArdation	Specify retardation coefficient for a transport variable
66	ROCK	Material properties of solid component of matrix
67	SAVE	Frequency of output to archive file
68	SCALE	Internal scaling of specified input
69	SCREEn	Echo some of the diagnostic output to screen
70	SELEct	Specify location of sub-regions or boundaries
71	SET	Set value of a variable as a function of space and time
72	SOIL	Material properties of solid component of matrix
73	SOLVE	Start of solution of equations
74	SOURCE	Specify source, injection or withdrawal terms
75	STACK	To park variables or transformations on stack for later use
76	STATistics	Obtain output of statistical parameters of variables
77	STORAge	Specify storage coefficient for a governing transport equation
78	SWIRL	specify cylindrically swirling flow in the context of Cartesian coordinates
79	SYMMetry	To identify an external boundary as a symmetry plane or axis
80	THERmal	Thermal properties of solid or porous matrix
81	TIME	Set initial time for simulations
82	TITLE	Problem title specification
83	TRACk	Compute Particle tracks and corresponding elapsed time in flow field
84	TRANsport	Transport properties of porous matrix
85	USER	User identification for input and output files
86	VELOcity	Select method of computation of Darcy velocity
87	VISCosity	Select fluid viscosity options
88	WRIT	Generate output of variables
89	ZONE	Specify host media zones

6.2 ORDER OF INPUT COMMANDS

The **FREEFORM™** command language allows an order-free input format except for the constraints imposed by common sense. For example, the command that initiates the solution of the equations must follow complete specification of the geometry and physics of the problem. Similarly, the command that signifies the end of calculations must be the last input record for a given problem. Table 6.2.1 lists the relationship of the keyword commands to various functional aspects of problem specification. Although it is not necessary, it may be convenient to follow this functional order. The **SOLVe**¹ command initiates solution of the governing equations and must, therefore, follow complete specification of the problem. The **END** command terminates the execution of a solution and, therefore, must be the last command of an input sequence for a problem.

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

TABLE 6.2.1: FUNCTIONAL UNITS OF PORFLOW™ COMMANDS

ORDER	FUNCTION	RELATED KEYWORD COMMAND
1	Identification	BAN Ner, TIT le, US er
2	Geometry specification	AD APtive, CON nectivity, COOR dinate, GEOM etry, GR ID, SCAL e
3	Problem definition	GRA avity, INCL ude, INLE t, LOC At, MATER ial, OP en, OUTLE t, PHAS e, PROB lem, REN ame, SELE ct, SWIR l, SYMM etry, ZON E
4	Initial and boundary conditions	BOUN dary, DATU m, FIX , FLOW , INIT ial, INLE t, OP en, OUTLE t, PER iodic, REA d, SE T, STAC k, SYMM etry, TIM E
5	Fluid Properties	DENS ity, FLUI d, GAS , REFE rence, VISC osity
6	Soil/rock matrix properties	CON ductivity, CAP illary, DIFF usivity, DIST ribution, FOR , HYDR aulic, MATER ial, MULT iphase, PROP erty, RET ardation, SOIL , STOR age, ROCK , THER mal, TRAN sport
7	Source and sink specifications	EBU , DECA y, PREC ipitate, REA ction, REGE neration, SOUR ce
8	Solution options	DIS able, INTE gration, LIMIT , MATR ix, OPT ion, REL ax
9	Output control	CLOS e, COR relation, DEBU g, DIAG nostic, FLUX , HIST ory, PRIN T, OUTP ut, SAVE , SCRE en, STAT istics, TRAC k, VELO city, WRIT e
10	Operational control	ALLO cate, CON vergence, DEF ine, ELSE , END , ENDIF , FILE , IF , PAUS e, QUIT , SOLV e

¹The 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.

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 all files, except those marked with a double asterisk (**) may be changed by the corresponding user commands listed.. 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. All these units are named with the prefix “acr_” and the file extension ”TMP”. For example, unit automatically opened in response to the **CORRelation** command is named ‘acr_CORxxxx.TMP’ where “xxxx” is an alphanumeric string.

TABLE 6.4.1: I/O FILE UNITS AND THEIR DEFAULT ASSIGNMENT

UNIT #	DEFAULT FILE NAME	FUNCTION OF THE FILE
15	Console	Read user input commands from standard input device.
16	Printer	Write to standard output device.
81	xxxINIT.ACR**	ACRi Initialization file; xxx is replaced by user specific string
Auto	xxx_LCNS**	ACRi user License file; xxx is replaced by user specific string
Auto	acr_DEBUG.TMP	User specific diagnostic and debug output (DEBUG).
Auto	acr_GRID_QUALITY.TM P	Grid quality information (DEBUG GEOMetry).
Auto	acr_MTRXxxxx.TMP**	Solution Matrix Coefficients (DEBUG MATRix).
Auto	acr_FLUX.TMP	Write data for fluxes crossing a sub-region boundary (FLUX).
Auto	acr_GRID_XYZ.TMP**	Internally generated ACRi format grid file (GRID PLOT3d).
Auto	acr_HISTORY.TMP	Time-history data file (HISTORY).
Auto	acr_HISTORY_XYZ.TMP	Time-history at specified locations(HISTORY COORdinate).
Auto	acr_SOURCE.TMP	Time-history of source of variables(HISTORY SOURce).
Auto	acr_STORAGE.TMP	Time-history of storage of variables(HISTORY STORAge).
Auto	acr_REDBLACK.TMP**	Red/Black matrix split information (MATRIX REDBlack).
Auto	acr_NOZZLE.TMP	Nozzle spray trajectory information (NOZZLE TRAjectory).
Auto	acr_PRINT_FORCE.TMP	Time-history of forces or moments on a surface (PRINT FORCEe).
11	acr_SAVE.TMP	Write data file for restart, archiving and post-processing (SAVE).
Auto	acr_RESTART.TMP	ACRi restart file (SAVE REStart, READ REStart).
12	acr_SAVE_TABLE.TMP	Write archive data file in tabular form (SAVE TABLE).
Auto	acr_STATS.TMP	Statistics information pertaining to variables (STATISTICS).
Auto	acr_TRACK.TMP	Particle Track information for the flow field (TRACK).
Auto	acr_XYZ_CRNR.TMP**	Corner coordinates for each element (WRITE CORNers).
Auto	acr_VRTX_MAP.TMP**	Vertex Mapping information automatically generated if needed.

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 PORFLOW™

PORFLOW™ 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. It should be noted that the user has both the flexibility to define new transport variables with the **ALLOcate** command as well as the ability to rename some of these variables with the **REName** command..

TABLE 6.7.1: DEPENDENT VARIABLES OF PORFLOW™

PORFLOW™ Name	Mathematical Symbol	Description of the Variable
P	P^1	Pressure head at reference density for the first (primary) fluid phase
P2	P^2	Pressure head at reference density for the second fluid phase
P3	P^3	Pressure head at reference density for the third fluid phase
T	T	Temperature
C	C	Mass concentration of 1 st species in fluid
C2,C3, ...,Ck	C^k	Mass concentration of 2 nd through k th species in fluid
User defined	none	Variables defined by the user with the ALLOcate command or one of the above variables renamed by the user with the REName command.

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 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 (**CONN**ectivity command) determines the number of vertices.

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

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

Table 6.8.1 lists the node-based variables for which output can be obtained. These are termed “standard” output variables. The output is always written to the Standard Output device (unit 16). The extent and frequency of this output is controlled by the **OUTPut** command. The **OUTPut** command specifies the variables to be written to the output device, their order, frequency and mode of output, and the sub-domain for which output is required.

Table 6.8.2 lists some of the other variables for which output can be obtained. These are collectively termed “supplementary” variables. The output for these can be directed to the Standard Output device or a user-specified file. The output for these is controlled by the **WRITe** command.

TABLE 6.8.1: STANDARD VARIABLES OF PORFLOW™

PORFLOW™ Name	Mathematical Symbol	Description of the Variable
U	U	Velocity component in x-direction
V	V	Velocity component in y or r-direction
W	W	Velocity component in z or l-direction
P	P ¹	Pressure head at reference density for the 1 st (primary) fluid phase
P2	P ²	Pressure head at reference density for 2 nd fluid phase
P3	P ³	Pressure head at reference density for 3 rd fluid phase
T	T	Temperature
C	C	Mass concentration of 1 st species in fluid
C2,C3, ...,Ck	C ^k	Mass concentration of 2 nd through k th species in fluid
S	S ¹	Saturation fraction for the first fluid phase
S2	S ²	Saturation fraction for the second fluid phase
S3	S ³	Saturation fraction for the third fluid phase
RHO	R ¹	Normalized density of the first fluid phase
RHO2	R ²	Normalized density of the second fluid phase
RHO3	R ³	Normalized density of for the third fluid phase
H	H ¹	Total hydraulic head for the first fluid phase
H2	H ²	Total hydraulic head for the second fluid phase
H3	H ³	Total hydraulic head for the third fluid phase
MOIS	φ _E	Effective moisture in the first fluid phase
User defined	none	Variables defined by the user with the ALLOcate command or one of the above variables renamed by the user with the RENAme command.

TABLE 6.8.2: SUPPLEMENTARY VARIABLES OF PORFLOW™

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

6.9 ARCHIVAL AND POST-PROCESSING OUTPUT

The archival and post-processing outputs may be obtained in three distinct modes. These files 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. Both modes of output are controlled by the **SAVE** command which selects the variables to be archived, the frequency of output, the file name, and the nature of data records (formatted or unformatted). To invoke the second mode the modifier **TABLE** must be present with the **SAVE** command; otherwise the first mode is invoked. In either case, the output consists of several records for each data set. Whenever these records are written on the 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.

In the 1st mode, the variables are listed individually and separately for whole of the domain of computation. 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 to file unit 11 in either unformatted or formatted records, depending on user specification. By default, the file connected to this unit is given the name 'acr_SAVE.TMP' and the data records are formatted.

In the 2nd mode, the variables are listed at each node in a tabular manner for 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. The information is written to file unit 12 and, unless explicitly named by the user, the file is named 'acr_SAVE_TABLE.TMP'.

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

```

TITLe EXAMPLE OF A TWO SEGMENT CALCULATION WITH OUTPUT OPTION CHANGES
////////////////////////////////////////////////////
GRID 11 BY 12
COORDinate X: RANGE = 100, grid spacing increment ratio = 1.1
COORDinate Y: RANGE = 40, grid spacing increment ratio = 1.
/
MATERial TYPE is 1 from (1,1) to (12,12)
/
INITial P = 1. from (2,2) to (4,4)
INITial T = 100 from (2,2) to (4,4)
SET T = 0 everywhere
/
SOIL DENSity = 1
SOIL POROSity = 0.4
HYDRaulic properties: storativity 0.1; conductivity x = 1., y = 1.
OUTPut for P (pressure) at this stage ! Print initial values of P
/
//////////////// Comment. Start of first segment of calculations
SOLVE for 5 years in time step of 0.1 yr
/
OUTPut for variables P and T
SAVE variables U, V, P
/
//////////////// Comment. Start of second segment of calculations
SOLVE for 15 more years in time step of 0.5 yr
/
OUTPut for variables U, P and T
SAVE variables U, V, W, P and T
/
//////////////// Comment. Start of third segment of calculations
SOLVE for another 30 years in time step of 1 yr
/
OUTPut for variables U, V, W, P and T
SAVE variables U, V, W, P and T
/
END
    
```

6.12 DIMENSIONING PARAMETER

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

(This page left intentionally blank.)

CHAPTER 7

PORFLOW™ KEYWORD COMMANDS

This chapter describes the keyword commands that comprise the user's interface with the PORFLOW™ 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 PORFLOW™ 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 typeface denote modifiers of **FREEFORM™** modifiers 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 nth numeric field in an input command denoted by a keyword.

fname The name of a file or device; see Section 7.2.2.

subrgn The subregion for the applicable input; see Section 7.2.3 and 7.2.4.

idsub A unique identity for a subregion of the domain of computation; see Section 7.2.4.

func An analytic or tabular function for input of a numerical quantity; see Section 7.2.6.

phase Fluid phase to which the input applies; see Section 7.2.10.

V_{frq} Frequency of output; see Section 7.2.11.

7.2 GENERAL INPUT FEATURES FOR FREEFORM™ COMMANDS

7.2.1 *Interactive or Run Time Input*

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

7.2.2 *File Name Specification*

Many input commands allow part or all of the input data to be read from a user-specified file (or device) or output to be directed to an output file. The name of the file, which is denoted by 'fname' in this manual, is specified as a character expression that **must be enclosed in single or double quotes**. The file or device name may be up to 64 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 77.

In general, if a file name is specified then it is **by default assumed to be in "FORMATTED" mode**. For some input commands, the "UNFORMATTED" file mode is also allowed. This 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. However, if the modifier "DEFER" is specified, then the file stays closed. In this case, user must explicitly open the file or an error will occur if an attempt is made to access the file.

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 **LOCAt**e or **SELEct** command. Six modes of subregion specification are available which are listed in Table 7.2.1.

TABLE 7.2.1: SUBREGION DEFINITION AND APPLICABILITY

MODE	METHOD OF SPECIFICATION
1	A rectangular subregion is specified by its lower-left and upper-right corners in terms of grid indices (I,J,K). The subregion is specified by 4 numerical values in the 2D mode and 6 in the 3D mode. In this mode, all the nodes within the specified window may be selected or nodes at a specified interval within the window may be selected. The node interval may be specified by 2 (for 2D) or 3 (for 3D) numerical values which follow those specifying the window. This mode of input can be used only for structured grids. It applies to all commands that operate by defining a subregion.
2	One or two rectangular subregions are specified by their lower-left and upper-right corners in terms of grid coordinates (x, y, z). The elements that belong to the window(s) are those with their node points located within the window(s). For each window, the input is specified by 4 numerical values in the 2D mode and by 6 in the 3D mode. Some set operations can be performed on the windows. This mode of input can be used for both structured and unstructured grids. It applies to all commands that operate by defining a subregion.
3	A previously defined MATER ial TYPE is specified as a distinct subregion by a single numerical value. In this mode, the subregion may be non-rectangular and non-contiguous. This mode of input can be used for both structured and unstructured grids. It applies to all commands that operate by defining a subregion.
4	The subregion is specified by a list of elements. The list may be ordered as a sequence of consecutive element numbers with specified interval or it may be a list of element numbers in arbitrary order. This mode of input can be used for both structured and unstructured grids. It applies to all commands that operate by defining a subregion.
5	The subregion is specified by a list of elements and surface pairs in arbitrary order. This mode of input can be used for both structured and unstructured grids. It applies only to those commands that operate on the boundary of a subregion.
6	A point (termed STAT ion) is specified by 3 numerical values that denote its (x, y, z) grid coordinates. This option operates in a different manner than the other options. For the other options, a numerical value is directly (and, if appropriate, immediately) assigned to the subregion as a constant or a functional relationship. However, for the station option, the user assigns a value for the variable at a specified number of stations. The field values for the variable are computed throughout the computational domain by interpolation. This mode of input can be used for both structured and unstructured grids. It applies only to the SET command.

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 **LOCAt**e 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 **LOCAt**e or **SELEct** command. The choices available for the 'subrgn' modifier are summarized in Table 7.2.2.

TABLE 7.2.2: VALID SUBREGION IDENTIFICATION MODIFIERS

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

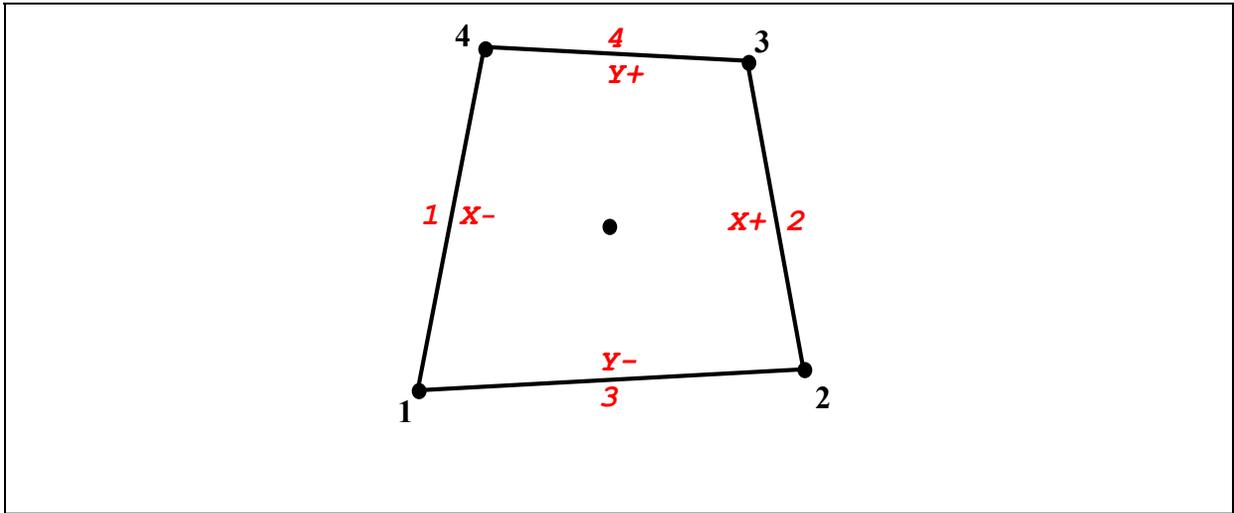
7.2.5 Boundary Identification

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

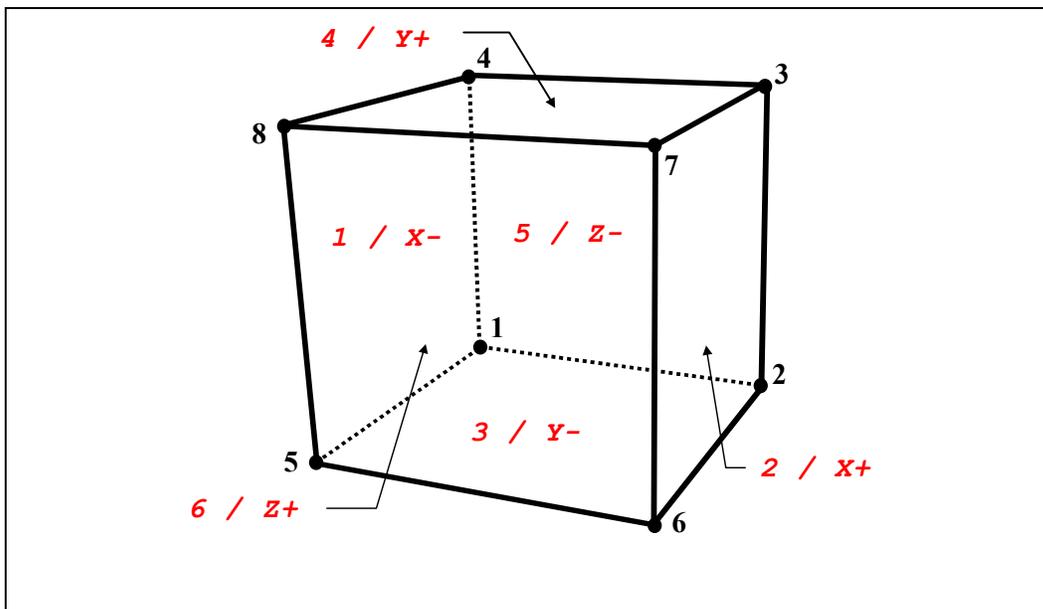
The identification of the boundary orientation by a direction modifier is illustrated in Table 7.2.3. The modifier consists of a character followed by a sign that represents the direction of the outward normal at the boundary. It should be noted that this direction is a nominal direction with respect to the local orientation of the grid element (ξ, η, ζ for a curvilinear grid). This orientation may or may not be related to the global framework in which the (x, y, z) coordinates of the element vertices are specified. The local orientation for an element is uniquely determined by the manner in which its vertices are numbered. The notation is illustrated in Figure 7.2.1 for both 2D and 3D elements. The vertices are numbered in an anti-clockwise manner in the xy -plane. Briefly, the local x (ξ) vector points from vertex 1 to vertex 2, the local y (η) vector from vertex 1 to vertex 4 and, for 3D elements, the local z (ζ) vector from vertex 1 to vertex 5. For a structured grid the local and global framework are consistent with each other. However, for an unstructured grid, the local framework may change from element to element since the vertex numbering may be completely arbitrary. In such a case a more general description is used which consists of the specification of a paired list of element and surface numbers (see **LOCate** 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 7.2.3: BOUNDARY IDENTIFICATION INDEX

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



Vertex and Face Number Nomenclature for a Quadrilateral Element



Vertex and Face number Nomenclature for a Hexahedral Element

FIGURE 7.2.1: BOUNDARY INDEX NOTATION

7.2.6 General Functional Relationships

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

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

here Φ represents a field variable, or a source or boundary value for a dependent variable, and represents time, a spatial coordinate or a field variable. A library of built-in tabular and analytic functions for Φ is provided. In addition, it is possible to specify user-defined functions as discussed in Section 7.2.9. The user should note that the library of built-in functions is constantly being updated and additional options may have been incorporated subsequent to the date of publication of this manual. Please contact **ACRI** for the latest information

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

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

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

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

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

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

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

In addition to the piece-wise tabular options above, several analytic functions are currently available; these include:

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

where 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 ξ , respectively. Valid input options for func and ξ and their corresponding interpretations are given in Tables 7.2.4 and 7.2.5. 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 7.2.6 through 7.2.9. **By default, the value of the function at a location is computed from the value of the independent variable, ξ at the same location. However, if the modifier "STAC" is present on a command then the function is evaluated with the value of ξ at the most recent location specified by the STACK command.**

TABLE 7.2.4: VALID INPUT FUNCTIONAL FORMS

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

TABLE 7.2.5: VALID INDEPENDENT VARIABLES

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

TABLE 7.2.6: NUMERICAL INPUT FOR THE TABLE SPECIFICATION

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

TABLE 7.2.7: NUMERICAL INPUT FOR SIMPLE ANALYTIC FUNCTIONS

Numerical Value	Interpretation
N1, ..., Nn	The constants A_n for Equation 7.2.4 or the constants A through E for the functions defined by Equations 7.2.5 through 7.2.17, as appropriate. The number of values must match those required by the function. For Equation 7.2.4, the number of values must equal the number of nodes in the subregion for which the input is specified. The constant function is specified by a single numerical value. Two values are required for the linear relation, five for the polynomial and four for all other functions.

TABLE 7.2.8: NUMERICAL INPUT FOR THE SERIES FUNCTIONS

Numerical Values	Interpretation
N1	The number of sets of input in the series specified by Equations 7.2.18 through 7.2.23.
N2, ..., Nn-1	The triplet of values of A_n , B_n and C_n (in that order), for the series functions. A total of $3*N1$ values (N1 sets) must be specified.
Nn	The datum A_0 for the function.

TABLE 7.2.9: NUMERICAL INPUT FOR USER SPECIFIED FUNCTIONS

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

7.2.8 Examples of Functional Form of Input

The Tables 7.2.10 through 7.2.12 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 7.2.10: EXAMPLES OF SIMPLE ANALYTIC FUNCTIONS

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

TABLE 7.2.11: EXAMPLES OF ANALYTIC FUNCTION SERIES

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

TABLE 7.2.12: EXAMPLES OF TABULAR FUNCTIONS

Illustrative Commands
<p>//// The value of P at the boundary is a tabular function of time BOUNDary P value at Y- boundary: TABLE of values: 3 sets (TIME, value) (0., 0.01), (100., 0.10), (200, -0.20)</p>
<p>//// The flux of P at the boundary is a tabular function of time BOUNDary T at X-: FLUX is a TABLE in TIME: 3 sets (0., 5), (100., 15), (200, 7.5), h=0.001</p>
<p>//// The field values of temperature are set as a tabular function of time SET T as TABLE of values: 3 sets: (0., 0.01), (100., 0.10), (200, -0.20)</p>
<p>//// As above except that step-wise, rather than the linear, interpolation is used SET T is STEP wise TABLE: 3 sets: (0., 0.01), (100., 0.10), (200, -0.20)</p>
<p>//// As above except that the values will be repeated after 200 time units SET T is PERIodic TABLE with STEP wise interpolation: 3 sets: (0., 0.01), (100., 0.10), (200, -0.20)</p>
<p>//// Table with linear interpolation but time input set to result in steep step-like interpolation SET T for as TABLE of values: with 6 sets: (TIME, value) (0., 0.0) (1.0000, 0.0) (1.0001, 1.0) (2., 1.0) (2.0001, 0.5) (3.0000, 0.2)</p>
<p>//// Table with linear interpolation but input from a file SET T for domain as function of TIME: 20 sets from file 'TIMEVALS'</p>

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 7.2.13. [The phase modifier is currently only available with the multi-phase versions of the ANSWER™ and PORFLOW™ Software Tools; it is not available with TIDAL™ series of ACRI software.](#)

TABLE 7.2.13: VALID MODIFIERS FOR PHASE DESIGNATION

User Specification of phase	Interpretation For The Corresponding Command
FIRS	The input is specified for the first phase of the fluid. This is always the default option; if no modifier is specified, then the input is assumed to be for the first phase of the fluid.
SECO	The input is for the second phase of the fluid.
THIR	The input is for the third phase of the fluid. This modifier is available only for the PORFLOW™ Software Tool.
GAS	The input is for the GAS Phase of the fluid. The gas phase is always the last of the active fluid phases. Thus this modifier is equivalent to the FIRSt modified for a single phase gas, the SECOnd modifier for a liquid-gas, and the THIRd modifier for a liquid-liquid-gas simulation.
VAPO	Same as GAS Modifier

7.2.11 Specification of 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_{freq} and may be specified in terms of the number of steps or the time interval. The step ineterval mode is the default mode and in this case, for example, a specification of 10 will result in output at step number 10, 20, 30, etc. The time interval mode is activated if the 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 **ADAP**

PURPOSE To compute a new locally refined grid by adaptive methodology from the currently specified grid. The output of this command is an unstructured dataset.

SYNTAX **ADAP** { Φ } {N1} [**casename**]

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

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

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

COMMENTS

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

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

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

EXAMPLES

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

/ First create DUDY.

ALLOcate 'DUDY- Gradient of U with Y'

SET DUDY GRADient of U with Y

/ Now generate a new unstructured refined dataset.

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

/ The new connectivity, split connectivity, coordinates, sub-regions, initial guesses will be generated in

/ files called 'Level-1.cnc', 'Level-1.blk', 'Level-1.xyz', 'Level-1.loc', 'Level-1.init' etc.

COMMAND **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 first character must be an alphabetic character. **The first 4 characters of the user input are used as the symbolic name to define input for, or properties of, the new variable.** If more than 4 characters are specified, then the name must be enclosed in single or double quotes. If the name is longer than 4 characters but it is not enclosed in quotes then all characters except the first 4 are ignored. **The intended name must be the first modifier on the ALLOcate command.**

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 decreased.

EXAMPLES

ALLOcate VAR1

ALLOcate "VAR1 - A New Variable Defined Specifically for this Problem"

ALLOcate R1 - A variable to hold reaction rates for other variables

ALLOcate 'R1 - The reaction rate for the first chemical species'

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

SYNTAX **ALLO { TABL | MATE | ZONE } N1**

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

MATE By default memory space is allocated for 100 material types. This command must be used to increase the allocated memory if the user specifies more than 100 material types.

ZONE Same as MATE modifier.

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

EXAMPLES

ALLOcate TABLE space for 50000 words of problem specific input

ALLOcate space for 1000 MATERial Types

ALLOcate space for 1000 ZONEs

COMMAND **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 **BOUNDARY**

PURPOSE To specify boundary conditions for the computational domain.

MODE 1: **Constant Boundary Conditions**

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

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

option The type of the boundary condition.

option	INTERPRETATION
VALU	Value of the variable at the boundary node is specified (Equation 3.11.2). <i>This is the default option.</i>
FLUX	Flux of the variable (Equation 3.11.3) at the boundary is specified. <i>The flux is positive if it enters the computational domain and negative if it leaves it.</i>
GRAD	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.
MIXE	A combination of the variable value and its gradient or flux (Equation 3.11.4) at the grid element interface is specified.
FLOW	The boundary value depends on the direction of flow. For an incoming flow, the boundary value is fixed (VALU). For an outgoing flow, the normal gradient at the boundary is assumed to be zero.
EXTR	The normal gradient at the boundary is extrapolated from the neighboring values immediately inside the boundary. <i>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.</i>

subopt The nature of the MIXEd boundary condition.

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

dir The orientation index for the outward normal at the boundary. See Table 7.2.3 in Section 7.2.5 for available choices. *If no value is specified, then the boundary condition is implemented at all outermost boundaries of the selected subregion.*

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

- N1** The boundary value of the variable (VALU option, F_o of Equation 3.11.2), the flux (FLUX option, q_F of 3.11.3), the outward normal gradient (GRAD option, $-q_F/\Gamma$ of 3.11.3), or the equilibrium value (MIXE option; F_o of Equation 3.11.4). **There is no default value.**
- N2** The transfer coefficient (h_F of Equation 3.11.4) for the **MIXEd** option. This input is ignored for other options. **There is no default value.**

COMMENTS

In general, boundary conditions at any boundary segment may be specified only once. Two independent **BOUNDary** specifications must not refer to the same element for the same boundary orientation index. **Unpredictable consequences may arise if the boundary information for any segment is repeated.** However, the specification for a boundary segment that was previously identified by a unique identity (**ID=idsub**; see Section 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 7.2.4 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 7.2.5. *If no variable is specified, then the independent variable is assumed to be time.*
- option** See Mode 1 specification.
- subopt** See Mode 1 specification.
- dir** See Mode 1 specification.
- subrgn** See Mode 1 specification.
- fname** The name of the file containing numerical values N1 through Nn. *This option can be used only if the selected function is a table or one of the series functions.* See Section 7.2.2 for additional information.
- N1, ..., Nn** The numerical constants and coefficients for the selected function. See Section 7.2.7 for more details. *There are no default values for this input.*
- Nn+1** The transfer coefficient (h_F of Equation 3.11.4). In the **FLUX** mode, the input units are those of velocity, heat or mass flux divided by the units of the dependent variable, Φ . In the **GRAD** 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: X+: MIXEd: POLYnomial in Y: (10., 0.5, 0., -0.1, 0.): h_F=0.01
//      Next 2 lines specify boundary T as a function of T
BOUNdary T at Y- boundary MIXEd type: SINE SERIes in T: 3 sets
           (0., 5, 1), (100., 10, 1 ), (200, 20, 3) datum= 0., h=1.E-4
//      Next 2 lines specify mixed type P boundary as a function of time
BOUNdary T at X-: MIXEd GRADient TABLE: 3 sets: (0., 5), (100., 15 ), (200, 7.5), h=0.001
BOUNdary MIXEd GRAD P at Y+: SINE SERIes 24 sets 'BMIXEDV' h=0.01
    
```

MODE 3: Total Pressure Boundary Condition

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

TOTA The total pressure at the boundary is specified. The pressure and velocity components at the boundary are computed such that these satisfy the relation:

$$P_{Total} = P + \frac{1}{2} \rho (u^2 + v^2 + w^2) , \quad \text{where } P_{Total}, \text{ is the total pressure, } P \text{ is the}$$

local pressure, ρ is the density, and u, v, w are Cartesian velocity components. **There is no default value; this modifier must be present on the command.**

P The symbol to denote that the input is for the pressure variable. **There is no default value; this symbol must be present on the command.**

func One of the modifiers listed in Table 7.2.4 that denotes the functional form of the input. The function specifies the value of total pressure. **If no function is specified then the value is assumed to be constant.**

ξ One of the independent variables listed in Table 7.2.5. **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.**

COMMENTS

Currently this option 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.E5

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: **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 **CAPIllARY**

PURPOSE To specify the coefficients for modification of capillary pressure due to changes in temperature or species concentration.

SYNTAX **CAPI** [Φ] [N1, N2, N3]

Φ One of the symbolic character strings: T, C, C2, C3 or C4. Denotes the dependent variable that influences capillary pressure. *By default, it is assumed that the dependence is on temperature.*

N1 The coefficient, α or α_c of Equation 3.3.7. *The default value for temperature dependence is 0.0809; that for dependence on chemical species is 0.*

N2 The maximum temperature, T_c , or coefficient β_c of Equation 3.3.7. *The default value for temperature dependence is 374.15; that for dependence on chemical species is 0.*

N3 The reference temperature, T^* , of Equation 3.3.7. *The default value is 20.0.*

COMMENTS

Capillary pressure may be made a function of both temperature and one of the chemical species by two independent commands. However, it can be a function of *only one* of the chemical species.

EXAMPLES

CAPIllary pressure function of T

CAPIllary pressure function of T: 0.01, $T_c = 647.30$, $T^* = 300$ K

CAPIllary pressure function of C: a = 0.05, b=-0.01

CAPIllary pressure function of C3: a = 0.1, b= 0.001

COMMAND **CLOSE**

PURPOSE To close output devices.

SYNTAX **CLOS** [fname] [filetype | NUNIT]

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.

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

filetype	Interpretation
DEBU	Debug output file (DEBUg), Unit 17, is selected for the operation.
FLUX	Flux output file (FLUX), Unit 14, is selected for the operation.
HIST	Time History file (HISTory), Unit 13, is selected for the operation.
SAVE	Archive file (SAVE), Unit 11, is selected for the operation.
TABL	Tabulated archive file (SAVE TABLE), Unit 12, is selected for the operation.
TRAC	Particle track file (TRACK), Unit 18, is selected for the operation.

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

EXAMPLES

- CLOSE** SAVE file
- CLOSE** archive TABLe output file now
- CLOSE** particlle TRAC file immediately
- CLOSE** file by name 'MYOLDFILE'

COMMAND **CONDUCTIVITY****PURPOSE** To specify conductivity or diffusivity and the treatment of interface diffusion.**MODE 1:** **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 7.2.4 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 7.2.5. *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 or ZZ. It denotes the component of the conductivity tensor to which the input is applied. *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.***COMMENTS**

For the **PORFLOW**[™] and **TIDAL**[™] ACRI Software Tools this command is implemented only for the species transport equations.

For the **ANSWER**[™] Software Tool this command can be employed for all equations. However, if the vector mode (**dir** modifier) is invoked, then the skew diffusion terms (see Mode 2) are ignored and the wall diffusive flux is set to zero for all except the momentum 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 2: 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:

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

where, q_{ϕ} is flux from node "1" to node "2", A_f is the interface area, 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:

$$q_{\phi} = q_{\phi_{\text{internal}}} + h A_f (\Phi_1 - \Phi_2)$$

TOTA The diffusive flux at the surface is computed as:

$$q_{\phi} = h A_f (\Phi_1 - \Phi_2)$$

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 7.2.4 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 7.2.5. 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.

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 **CONTA**ct must appear in the command.

MODE 3: **Modification of Diffusion Coefficient by Richardson Number**

SYNTAX **COND** {RICH} {dir} [α , β , α_ϕ , β_ϕ]

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

$$\mu_k = \frac{\mu}{[1 + \beta R_i]^\alpha}$$

$$\Gamma_k = \frac{\Gamma}{[1 + \beta_\phi R_i]^{\alpha_\phi}}$$

Where subscript “k” denotes the direction specified by the **dir** modifier, μ and Γ are, respectively, the scalar values of the viscosity and diffusion coefficients (specified by user or computed from built-in formulae), and α , β , α_ϕ and β_ϕ empirical constants. The Richardson number, R_i , is defined as:

$$R_i = g_k \frac{1}{\rho} \frac{\partial \rho}{\partial x_k} \frac{1}{\sum_{j, j \neq k} \left(\frac{\partial u_j}{\partial x_k} \right)^2}$$

where g_k is the gravitational acceleration in the k^{th} direction (specified by the **dir** modifier), ρ is the fluid density and u_j represents the velocity components in the directions other than the k^{th} direction.

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.**

α , β , α_ϕ , β_ϕ The empirical constants for the Richardson number formula. **The default values are 0.5, 10, 1.5 and 3.33, respectively.**

EXAMPLES

CONDuction modified by RICHardson Number

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

MODE 4: Treatment of Skew Diffusion Terms

SYNTAX **COND** {SKEW | SECO } [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 second 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.

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.**

EXAMPLES

CONDuction due to SKEW terms to be included for all variables

CONDuction due to SECOndary terms to be included for U and T

MODE 5: Elimination of Diffusion Terms

SYNTAX **COND** {OFF} [Φ]

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

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

EXAMPLES

CONDuction OFF for the temperature variable: T

COMMAND **CONNECTIVITY****PURPOSE** To specify element to vertex connectivity for unstructured grid.**MODE 1:** **Vertex Connectivity for Quad or Hex Elements****SYNTAX** **CONN { VERT } {fname}**

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

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

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

For **2D** geometry, the vertices must be specified in a **counter-clock wise fashion in the x-y plane**, such that the local (ξ, η) and the direction normal to the plane form a right handed system. For 3D geometry, the vertices on "**bottom**" side must be specified first (in counter-clockwise order) followed by the corresponding vertices on "**top**" side, such that the local (ξ, η, ζ) direction forms a right handed system. (Any side may be chosen as the "**bottom**", then the topologically opposite side is considered to be the "**top**".) The local (ξ, η, ζ) direction for each element is defined by the order in which the vertices appear on this record. The local ξ axis is oriented from vertex 1 to vertex 2, the η axis from vertex 1 to vertex 4, and the ζ axis from vertex 1 to vertex 5. These then determine the local side number (1, 2, 3, 4) or the local X-, X+, Y-, Y+, Z-, Z+ sides which are used to specify the boundary and boundary conditions. These concepts are illustrated in **Figures 1 and 2**. Some further details are also given in Section 7.2.5. **This is the default option.**

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

COMMENTS

An unstructured mesh is defined by:

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

EXAMPLES

CONNectivity information on file "VERT2ELM.CNC"

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

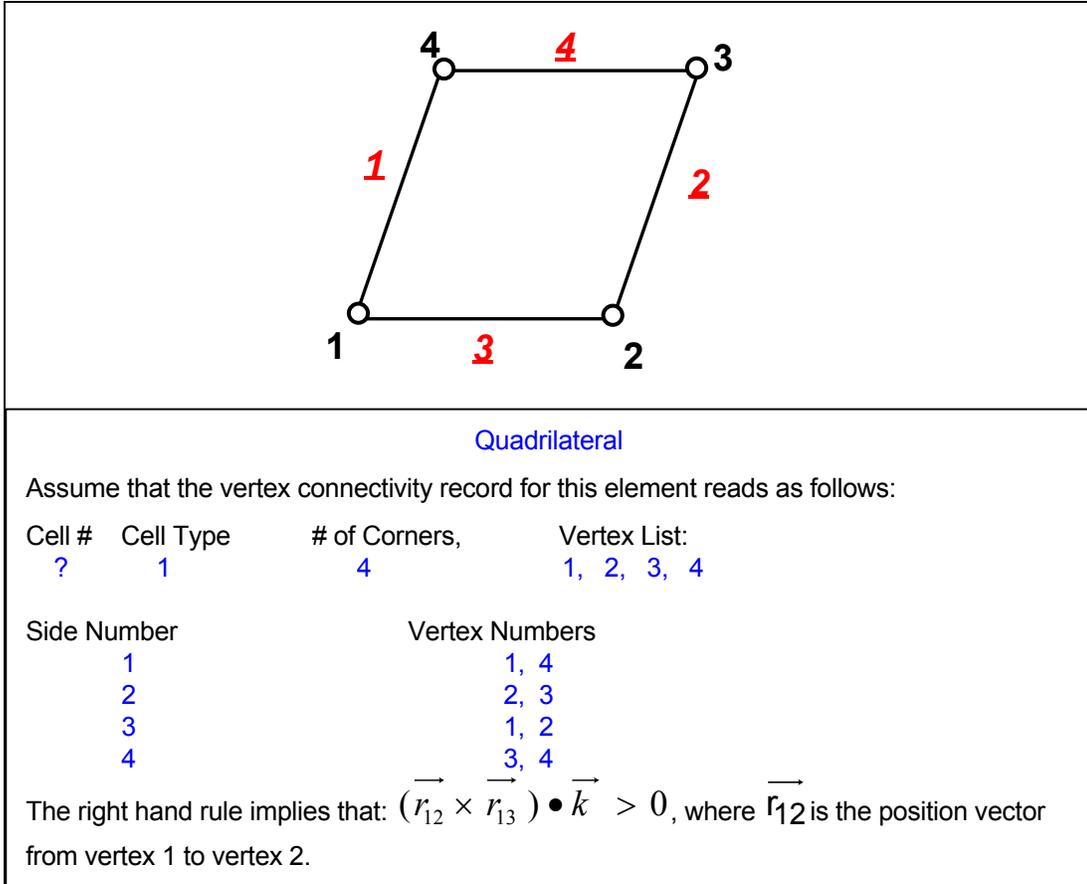


Figure 1: Relation between Vertex Numbers and Side Numbers for a Quadrilateral illustrating the application of the right hand rule. (2D only).

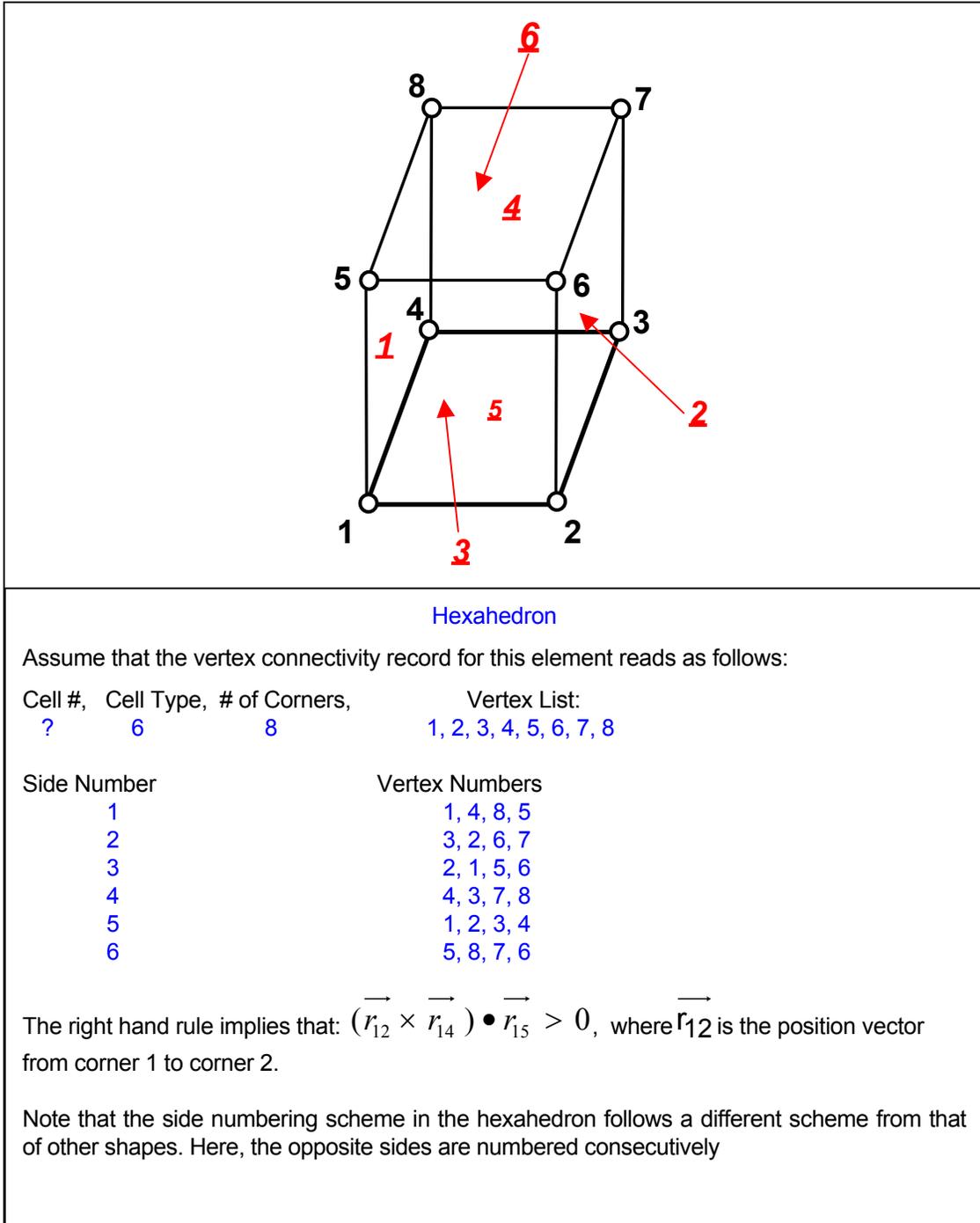


Figure 2: Relation between Vertex Numbers and Side Numbers for a Hexahedron illustrating the application of the right hand rule. (3D only).

MODE 2: Vertex Connectivity for Mixed Hybrid Elements.

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

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

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

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

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

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

Schematic of each element type and its connectivity describing the relationship of the local side numbering to the vertex connectivity is illustrated in Figures 1 through 6. Further details are given in Section 7.2.5).

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

EXAMPLES

CONNectivity for HYBRID elements on file "MIXED_ELEMENTS.CNC"

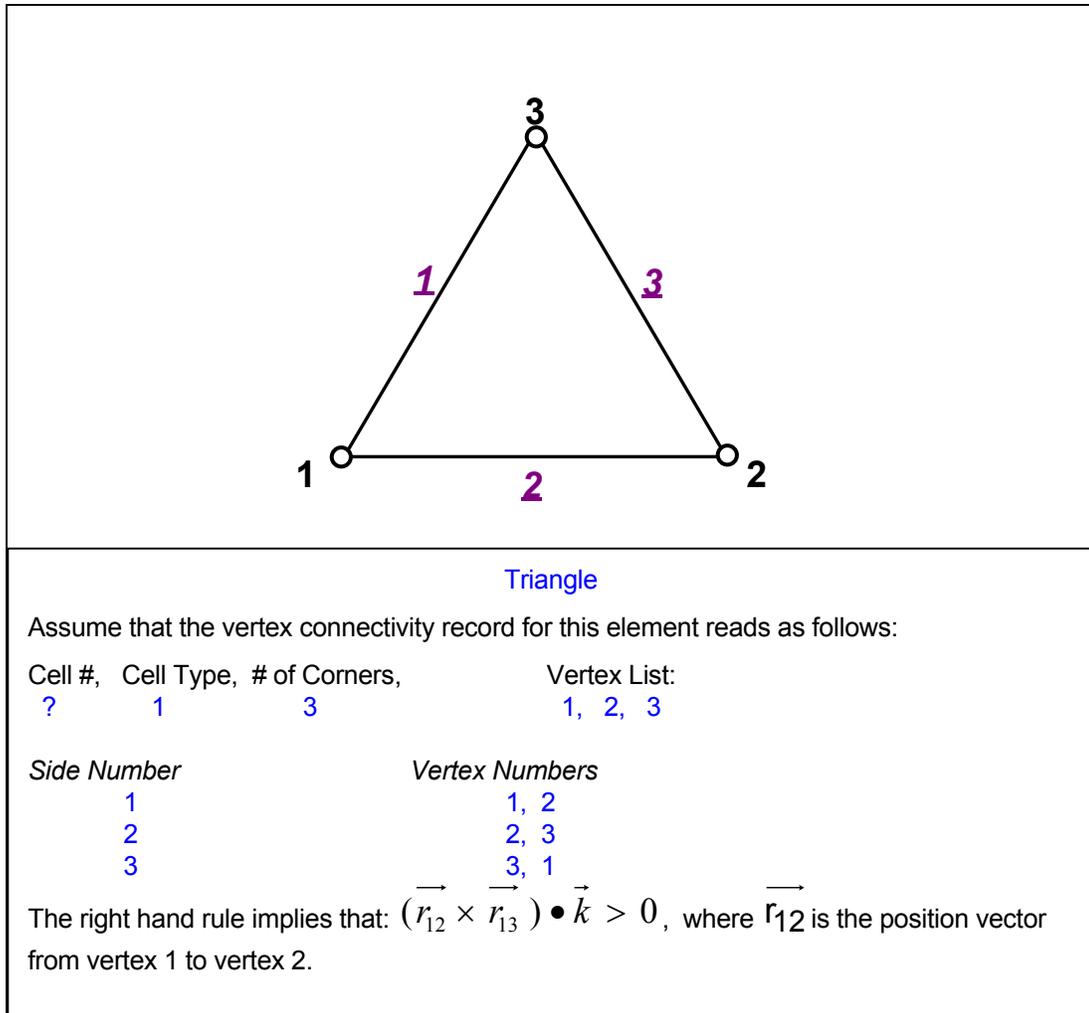


Figure 3: Relation between Vertex Numbers and Side Numbers for a Triangle illustrating the application of the right hand rule. (2D only).

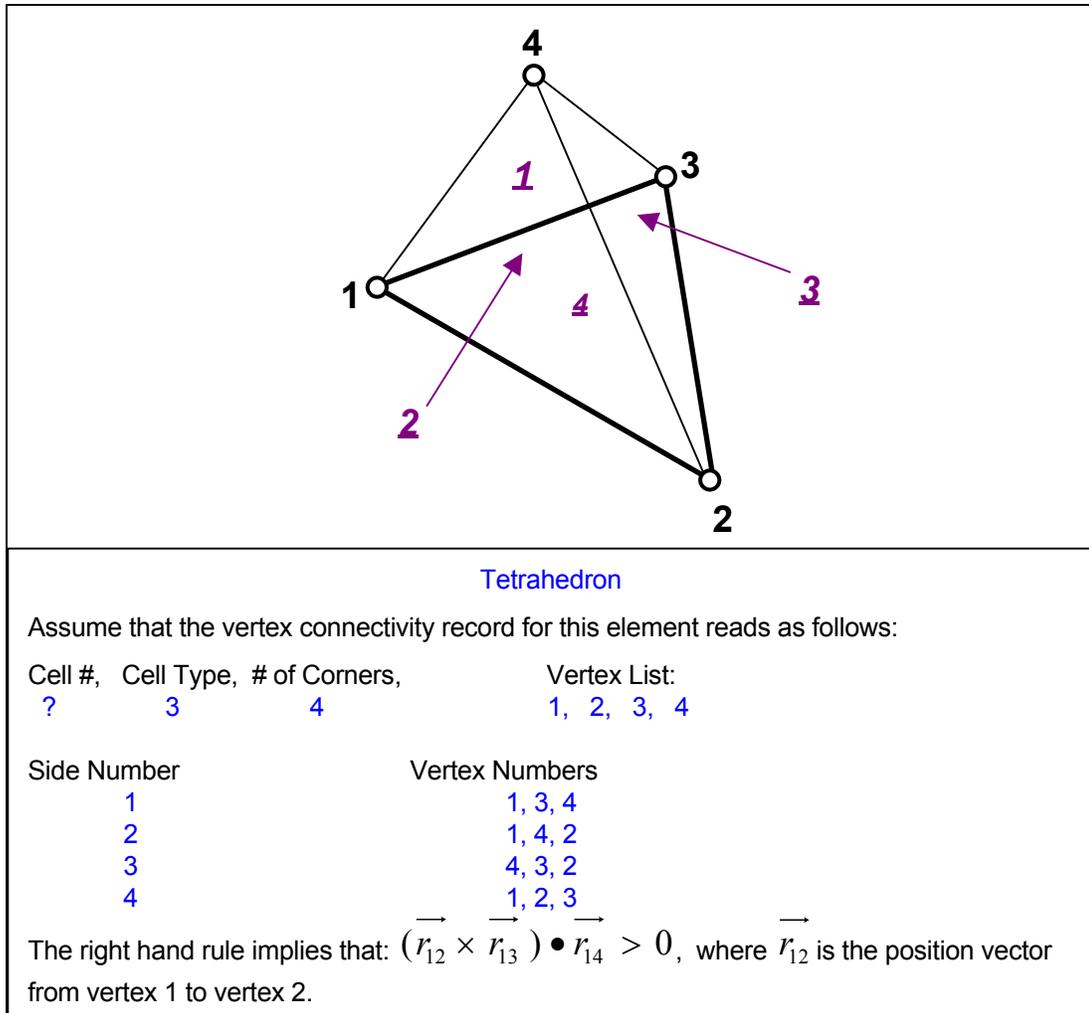


Figure 4: Relation between Vertex Numbers and Side Numbers for a Tetrahedron illustrating the application of the right hand rule. (3D only).

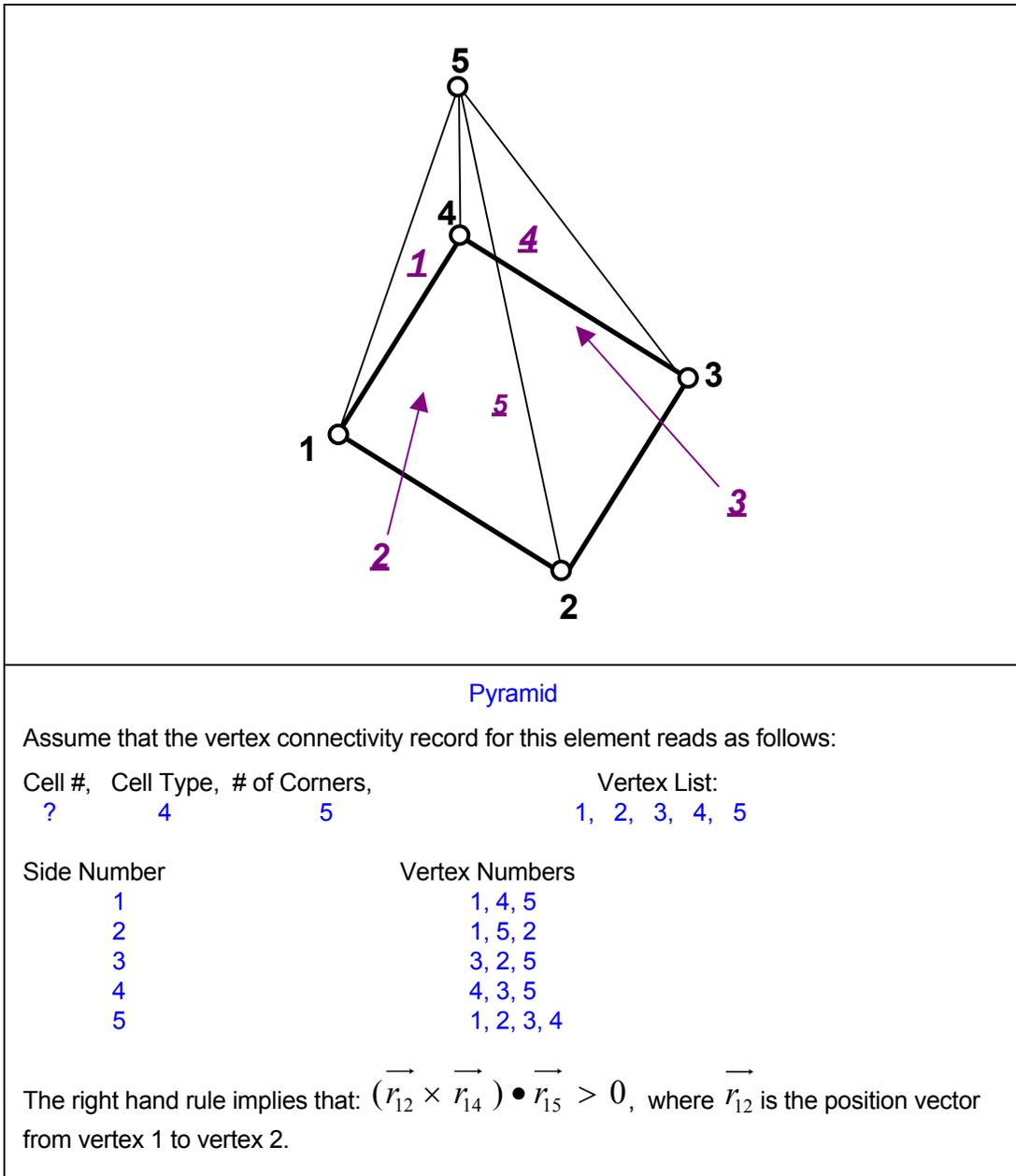


Figure 5: Relation between Vertex Numbers and Side Numbers for a Pyramid illustrating the application of the right hand rule. (3D only).

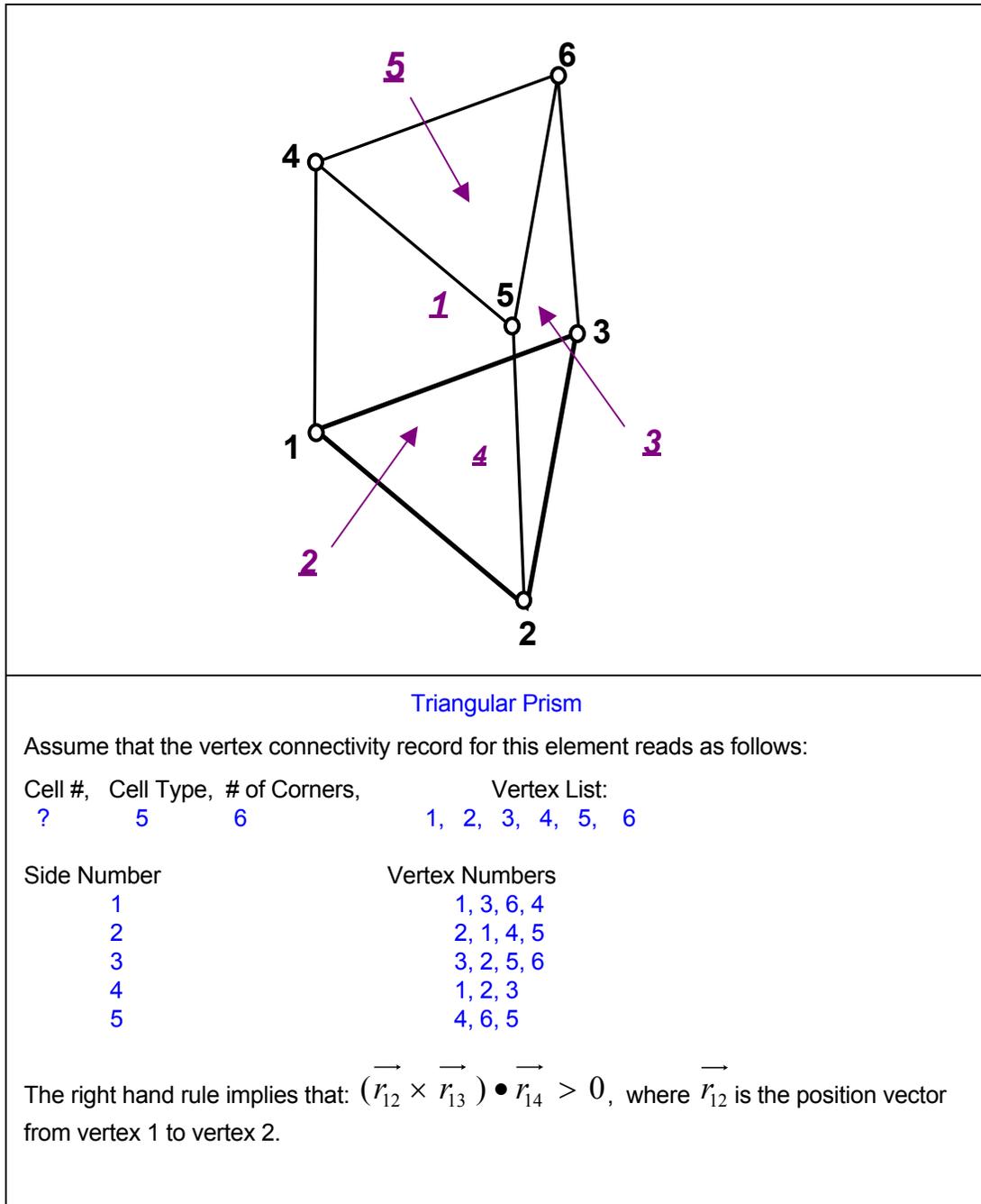


Figure 6: Relation between Vertex Numbers and Side Numbers for a Triangular Prism illustrating the application of the right hand rule. (3D only).

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

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

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

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

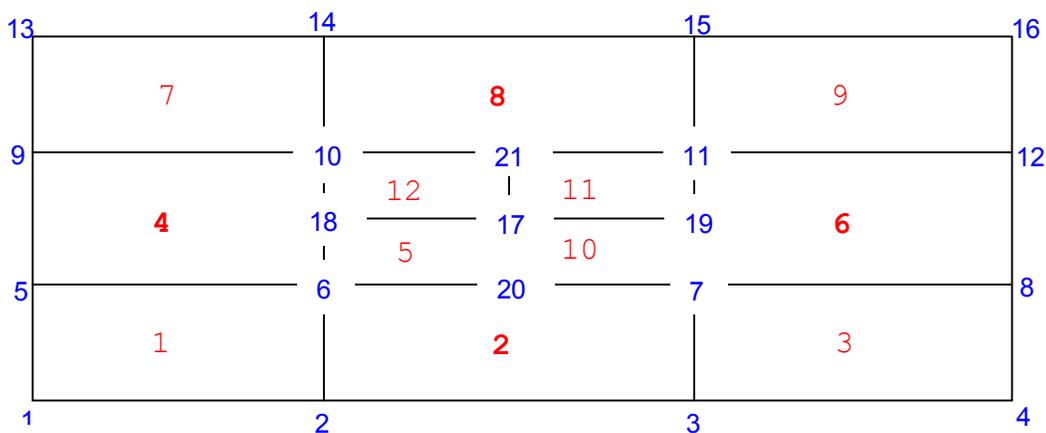
The 1st record in the file consists of a header with two numbers: the number of split elements and the total number of data items in the rest of the file. The header is followed by a number of sets of data equal to the number of split elements. The 1st record of each set consists of the element number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

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

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

fname The name of the file containing the numerical input for the split connectivity. [There is no default value; a file name must be specified.](#)

EXAMPLES



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

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

Then the SPLIt connectivity command is:

CONNectivity SPLIT on file "SPLIT.CON.

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

```

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

```

COMMAND CONVERGENCE

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

MODE 1: Specification of Convergence Tolerance and Iterations

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

Φ A symbol that denotes the dependent variable for which input is specified. The valid symbols are listed in Table 6.7.1. If no symbol is specified then the input is applied to control the outer (total) flow loop which consists of all the momentum (velocity) and mass equations that are active.

phase The fluid phase for which the input is specified. See Section 7.2.10 for available options. By default the input pertains to the 1st phase of the fluid. This modifier is currently available only for the PORFLOW™ Software Tool.

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

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

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

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

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

where A is coefficient matrix, X is the variable vector and B is the forcing function. By default this option is active unless the LOCA modifier is specified. Also, by default, the 1st equation is used unless the DIFF modifier is specified.

LOCA Convergence is judged by the maximum of the local residual defined by one of the following equations:

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

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

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

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

DIFF Convergence is judged on the basis of the 2nd or 3rd equation above based on whether the GLOB or LOCA mode of the command is selected. In each case, the 2nd equation is used by default unless the NORM modifier is present.

NORM This modifier is effective only in the presence of the DIFF modifier. In this case, the convergence residual is normalized as given by the 3rd equation based on whether the GLOB or LOCA mode of the command is selected.

- OLD** This modifier is effective only for PORFLOW and then only if the automatic time step mode is invoked. In this case, if the time step is decreased, then, by default, the most recent computed values (with a larger time step) are retained if the convergence residue was decreasing monotonically for the previous 3 iterations. However, if this modifier is present, then the computations are restarted from the values at the end of the previous time step.
- NEW** This modifier is effective only if the automatic time step mode of PORFLOW is invoked. If the time step is decreased, then computations restart from the most recent computed values (with a larger time step) rather than the values at the end of the previous time step.
- N1** The ϵ of the residual equations. The default value is 1.E-6.
- N2** Maximum number of “outer” iterations for convergence (see comments below). The default value may change for each installation. For most installations, the value is set to 1 with one exception. For the PORFLOW™ Software Tool, the number of iterations is set to 100 for the mass balance equation if the multi-phase or free-surface mode of the flow (pressure) equation is invoked.
- N3** Minimum number of iterations for the specified variable. The default value is 1.
- N4** The X_{ref} of the normalized residual equations above. The default value is 1.E-7.

COMMENTS

If the matrix coefficients are independent of the values of the dependent variable (linear equation) then a solution of the algebraic matrix equations ensures that the differential equation is satisfied. However, if the coefficients are functions of the dependent variable (non-linear equation), then the solution of the matrix equation does not guarantee that the differential equation is satisfied.

This command controls the “outer” iterations. The term “outer” refers to a complete pass through the governing transport equation where (1) the coefficients of the matrix are assembled, (2) the matrix is solved and (3) the convergence norm is checked against the convergence tolerance. For non-linear equations, it is necessary to iteratively perform these 3 steps till the differential equation is satisfied. The **N2** on this command specifies the number of passes through these steps. The solution of the algebraic matrix of equations is referred to as the process of “inner” iteration and is controlled by the **MATRIX** command.

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

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

EXAMPLES

CONVergence tolerance = 1.E-6
CONVergence criterion = 1.E-6 for SECOnd phase of fluid
CONVergence for U in LOCAI 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: LOCAI mode, value = 1.E-4, max iterations 10, min iter= 5; min value 1.E-5
CONV for flow: LOCAI mode, epsilon = 1.E-2, max iterations = 5
CONV LOCAI, eps=0.001, max iter=50, min = 10 , F_threshold=1.e-5;

CONVergence epsilon = 1.E-6, 20 iterations; always use OLD values

MODE 2: Reference Option for Overall Convergence of Solution Process**SYNTAX** CONV {REFE} [MOME | FLOW | ALL | ϕ] [N1, N2]

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

- (1) The MOMEntum option if the velocity and mass balance equations are solved.
- (2) The 1st dependent variable, if no flow equations are solved.

MOME The convergence of the solution process is judged on the basis of the individual convergence criteria specified for the momentum and mass balance equation. The process is assumed to converge only when the criteria for the momentum and mass equations are all simultaneously satisfied.

FLOW The convergence of the solution process is judged on the basis of mass balance. The flow equations must be solved for this option to be active.

ALL The convergence of the solution process is judged on the basis of the individual convergence criteria specified for the all active equations. The process is assumed to converge only when the criteria for all equations are simultaneously satisfied.

ϕ The convergence of the solution process is judged on the basis of the dependent variable denoted by the specified symbol. The valid symbols are listed in Table 6.7.1.

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

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

COMMENTS

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

EXAMPLES

CONVergence REFERENCE based on FLOW with Tolerance = 1.E-6

CONVergence REFERENCE FLOW with Tolerance = 1.E-6 with 10 outer loop iterations

CONVergence REFERENCE based on MOMEntum with flow Tolerance = 1.E-5

CONVergence REFERENCE based on ALL equations with flow tolerance =1.E-6

CONVergence REFERENCE based on ALL equations with flow tolerance =1.E-6 and 5 outer iterations

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

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

COUP **By default, at any time step, the species equations are solved sequentially only once.** This means that though inner iteration may be performed on any equation (see Mode 1 of command), each species equation is visited only once at each time step. If **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 **SPEC**ies in **COUP**led mode

CONVergence for **SPEC**ies in **COUP**led 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 **TER**minate if value exceeds 1.E50.

CONvergence **TER**minate if value reaches 1.E16.

COMMAND **COORDINATE**

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

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

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

RANG The total range (total domain length in the corresponding direction) is specified. The coordinates are calculated internally with a geometric ratio between the successive grid intervals. It is assumed that the coordinate system is structured and orthogonal.

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

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

DEGR By default, the angular input for θ is assumed to be in radian. If the modifier 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

COORDinate X: RANGE = 10.

COORDinate X: RANGE = 10., increase ratio = 1.05 !!! expanding grid

COORDinate X: RANGE = 10., decrease ratio = 0.95 !!! contracting grid

COORDinate Z: RANGE = 6.28 implement in the NODE mode

COORDinate R: RANGE = 10, ratio = 0.95

COORDinate R: RANGE = 10, ratio = 0.95 CYLIindrical

COORDinate THETA: RANGE = 270. DEGREEs

MODE 2: Structured Grid: Coordinate Minimum and Maximum**SYNTAX** **COOR** {MINI | MAXI} {dir} [CYLI] [DEGR] [NODE] {N1} [N2, N3]**MINI** The minimum and maximum values are specified. The coordinates are calculated internally, with a geometric ratio between the successive grid intervals.**MAXI** Same as MINI modifier.**dir** See Mode 1 specification.**CYLI** See Mode 1 specification.**DEGR** See Mode 1 specification.**NODE** See Mode 1 specification.**N1** The 1st coordinate value. *There is no default value; a value must be specified.***N2** The last coordinate value. *There is no default value; a value must be specified.***N3** The geometric ratio by which the size of the grid element (or grid interval) changes between successive elements (or nodes). *The default value is 1.***EXAMPLES****COOR**dinate X: MINImum = 0., maximum = 10.**COOR**dinate X: NODE values: MINImum = 0., maximum = 10. ratio = 1.05**COOR**dinate X: MINImum = 0., maximum = 10, ratio = 1.05 in NODE mode**COOR**dinate Z: MINImum = 0., maximum = 6.28**COOR**dinate THETA: MINImum 0., max = 270. DEGRees for the NODEs

MODE 3: **Structured Grid: Corners Points of the Computational Domain**

SYNTAX **COOR [CYLI] [DEGR] {N1, ..., Nn}**

CYLI See Mode 1 specification.

DEGR See Mode 1 specification.

N1, ..., Nn The coordinates of the corners of a quadrilateral for 2D simulation or those of a hexahedral for 3D simulations. Thus 8 values (x, y for each of 4 corners) must be specified for 2D and 24 values (x, y, z for each of the 8 corners) for a 3D simulation. In 2D mode, the coordinate values must be in the following order: the lower left, the lower right, the upper left, and the upper right corners. In 3D mode, the values must be in the same order as for the 2D first for the front plane (K=1) and then for the last plane (K=KMAX) of the grid nodes. For each grid line the computational domain is divided in to equal length elements. The nodes are then placed in the middle of each of the element except for the boundary nodes that are placed at the middle of the boundary face.

EXAMPLES

COORdinate corners are: (0., 0.) (1.,0.), (0., 1.) and (1., 1.) !! Unit Square

COORdinate: (0,0) (0.707,0.707), (-0.707,0.707) and (0,1.414) !45 deg Square

COORdinate: (0,0) (1.,0.), (0.,1.) and (1,1) CYLIndrical system

COORdinate: 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

COORdinate: 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**COOR**dinate X: 0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100 !!! element interfaces**COOR**dinate Y: 0, 5, 15, 25, 35, 45, 55, 65, 75, 85, 95, 100 at **NODES****COOR**dinate R: 0, 5, 15, 25, 35, 45, 55, 65, 75, 85, 95, 100 at **NODES****COOR**dinate Y: 0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100 **CYL**indrical mode**COOR**dinate **THETA**: 0, 5, 15, 25, 35, 45, 60, 75, 90, 105, 120 **DEGR**ees

MODE 5: Structured or Unstructured Grid: Element Vertex or Grid Node Coordinates**SYNTAX** **COOR** {fname} {dir} [CYLI] [DEGR] [BLOC] [NODE] [PLAN] [JIK|JKI|KIJ|KJI|IKJ] [N1]

fname The name of the file that contains the coordinate values. See Section 7.2.2 for additional information. The number of values specified must equal the number of specified directions times the number of grid vertices or grid nodes, as applicable. The specification of grid node coordinates (**NODE** modifier) is available only for a structured grid.

dir One or more of the **X, Y, Z, R** or **THETA** modifiers that, respectively, denote the x, y, z, r or θ coordinates. Up to two symbols may be specified for 2D and, 3 for 3D simulations.

CYLI See Mode 1 specification.

DEGR See Mode 1 specification.

BLOC By default, the coordinate values are assumed to be specified as sets of (x,y) or (x,y,z) for each vertex (or node) starting with the 1st vertex (or node) and ending with the last one. In the presence of this modifier, it is assumed that the values are specified in “block” mode; that is first all the x-coordinates, followed by the y and z coordinates, respectively.

NODE See Mode 1 specification; this modifier is applicable only for a structured grid.

PLAN This modifier is applicable only for a structured grid. By default, the number of coordinate values must equal the number of corners (or number of nodes). In the presence of this modifier, for a 3D grid, the input is assumed to be for a single z-plane; the other z-planes are assigned the same x and y values.

JIK This modifier is applicable only for a structured grid. By default, the values are read according to the FORTRAN DO loop convention (I, J, K) corresponding to the x, y and z direction respectively. For example, if only x coordinate for a 2D grid is specified, then the grid values x_{ij} must be in the order: $x_{11}, x_{21}, x_{31}, \dots, x_{n1}; x_{12}, x_{22}, x_{32}, \dots, x_{n2}; \dots$ and so on. If, say, both x and y coordinates for a 2D grid are simultaneously specified, then the set of values (x_{ij}, y_{ij}) must be in the order: $(x_{11}, y_{11}), (x_{21}, y_{21}), (x_{31}, y_{31}), \dots (x_{n1}, y_{n1}); (x_{12}, y_{12}), (x_{22}, y_{22}), (x_{32}, y_{32}), \dots (x_{n2}, y_{n2}); \dots$ and so on. However, if the **JIK** modifier is present, then it is assumed that the input data is in the order x_{ijk} . The I and J values are then transposed internally to the ACRI x_{ijk} format.

JKI This modifier is applicable only for a structured grid. Similar to the **JIK** modifier, except that the input is assumed to be in the (J,K,I) format.

KIJ This modifier is applicable only for a structured grid. Similar to the **JIK** modifier, except that the input is assumed to be in the (K,I,J) format.

KJI This modifier is applicable only for a structured grid. Similar to the **JIK** modifier, except that the input is assumed to be in the (K,J,I) format.

IKJ This modifier is applicable only for a structured grid. Similar to the **JIK** modifier, except that the input is assumed to be in the (I,K,J) format.

N1 The number of header records at the beginning of the file. These header records are read and ignored. The coordinate values are assumed to start with the $N1+1^{\text{st}}$ record. The default value is 0.

COMMENTS

If this command is used for cylindrical geometry in unstructured mode, then the vertices for elements must be defined such that the local element (ξ, η, ζ) coordinates are identical with the global (X, R, θ) coordinates. It is required that the external normal at Face number 3 points in the R^- and that at Face number 4 in the R^+ direction. Further details are given in Section 7.2.5 and the **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 co-located at the same point in space.**

EXAMPLES

COORDinates of **VERT**ices on file 'PROBGRID'

COMMAND **CORRELATION**

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

MODE 1: **Auto Correlation for the Variables**

SYNTAX **CORR** { Φ } [subrgn] [fname] [TIME] [NOW] [V_{frq}] [OFF]

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

$$\langle \Phi \rangle = \frac{1}{N} \sum_{n=1}^N \Phi^n$$

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

The following correlation quantities are then computed and printed:

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

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

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

subrgn The subregion for computations. If no subregion is specified, the entire domain is selected.

fname The file name for output. If a file name is specified then the output is directed to the named file, otherwise the output is directed to the standard output device. If a file name is specified, it must be different from any previously specified file name. 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 SOLVe command. If an exact time at which the output is due is not simulated, then the output is obtained at the first time step after the due time. By default the output is obtained only at the end of simulations.

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

EXAMPLES

- COR**relation for variable U for the SELEcted subregion at the end of simulations
- COR**relation for variable U for the ID=LOC_AUTO every 25 steps
- COR**relation OFF for variable U for the ID=LOC_AUTO

MODE 2: Cross Correlation for Two Variables

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

Φ_1, Φ_2 **Two** symbols that denote the variables for which correlation is desired. Valid symbols are listed in Table 6.7.1. If Φ_k denotes the instant value of the k^{th} variable at a given location, Φ_k^n its value at n^{th} time (or iterative) step, then the means over N steps are defined as:

$$\langle \Phi_k \rangle = \frac{1}{N} \sum_{n=1}^N \Phi_k^n$$

$$\langle \Phi_k^2 \rangle = \frac{1}{N} \sum_{n=1}^N (\Phi_k^n)^2$$

$$\langle \Phi_1 \Phi_2 \rangle = \frac{1}{N} \sum_{n=1}^N (\Phi_1 \Phi_2)$$

The following correlation quantities are then computed and printed:

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

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

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

- subrgn** The subregion for computations. If no subregion is specified, the entire domain is selected.
- fname** The file name for output. If a file name is specified then the output is directed to the named file, otherwise the output is directed to the standard output device. If a file name is specified, it must be different from any previously specified file name. The total number of open files in any simulation can not exceed 64.
- TIME** By default, V_{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 **SOLVe** command. If an exact time at which the output is due is not simulated, then the output is obtained at the first time step after the due time. By default the output is obtained only at the end of simulations.
- OFF** Any previously specified **CORRelation** commands for the specified variable and subregion are disabled. New commands may be subsequently specified.

EXAMPLES

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

MODE 3: Two Point Correlation for One or Two Variables**SYNTAX** **CORR** { **TWO** } { Φ_1 } [Φ_2] [**subrgn**] [**fname**] [**TIME**] [**NOW**] [V_{frq}] [**OFF**]**TWO** **Two point correlation** with matched sets of pairs of elements is specified. The subregion for this option must be previously specified with a **LOCATE CORRELATION** command. Φ_1, Φ_2 **One or Two** symbols that denote the variables for which correlation is desired. Valid symbols are listed in Table 6.7.1. The correlation formulae are identical to the two-variable cross-correlation described in Mode 2 except for the definition Φ_1 and Φ_2 . The Φ_1 is always taken to be the values of the 1st specified variable **at the elements of the 1st set of paired elements**. If **only one** variable is specified then Φ_2 is taken to be the values at the 2nd set of paired elements. If **two** symbols are specified then Φ_2 is defined to be values of the 2nd variable at the 2nd set of elements.**subrgn** The subregion for computations. **The subregion must be previously specified with a LOCATE 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, V_{frq} is interpreted as the frequency of output in terms of number of steps. In the presence of TIME this is the time intervals between successive outputs.****NOW** A record of is written to the output device at the next time step. This is in addition to the output from the V_{frq} specification. V_{frq} The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 for further details. **By default the output is obtained only at the end of simulations.****OFF** Any previously specified **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 **DATUM**

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

SYNTAX **DATU** { $N1, N2, N3$ }

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

COMMENTS

The datum coordinates enter the computations only if the multi-phase or free-surface flow options are used, or if the gravitational field is explicitly set to a non-zero value by the **GRAVity** command.

EXAMPLES

DATUM is at zero

DATUM coordinates for this problem are: $x=0.$, $y=1000$, $z=50$

COMMAND **DEBUG**

PURPOSE To obtain debug output related to specification of geometrical features, the error indicators and the matrix coefficients.

MODE 1: **Check Validity of Geometric and Grid Input**

SYNTAX **DEBU { GEOM } [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. A summary of grid statistics is written to the standard output file. Also a detailed output of orthogonality and uniformity index for each face is written to a file named 'acr_GRID_QUALITY.TMP'. The orthogonality index 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. The uniformity index is denoted by the Eulerian distance between each set of two grid nodes on either side of a face.

By default these checks are always performed.

OFF If this modifier is present, then the diagnostic and grid checks are omitted.

EXAMPLES

DEBU GEOMetry is on by default

DEBU GEOMetry checks OFF

MODE 2: Diagnostic Output of Numerical Error Indicators**SYNTAX** **DEBU** { Φ } [fname] [subrgn] [V_{frq}] [TIME]

- Φ The symbol for the dependent variable for debug output; valid symbols are listed in Table 6.7.1. There is no default value. A valid symbol must be specified.
- fname The file name for output. The default file name is 'acr_DEBUG.TMP'. A different output file may be specified for each command. If no file name is specified then the output is directed to the most recent previously specified file. If no previous file was specified, then the output is directed to the default file. The total number of open files can not exceed 64. A summary of output is also printed to the standard output device at the end of simulations.
- subrgn The subregion for computation of the error parameters. See Sections 7.2.3 and 7.2.4. If no subregion is specified, the entire computational domain is selected.
- V_{frq} The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 for further details. By default the output is obtained only at the end of simulations.
- TIME By default, V_{frq} is the computation (and output) frequency in terms of number of steps. If this modifier is present, then V_{frq} is the time interval between successive outputs.

COMMENTS

The debug output consists of the local Peclet and Courant numbers. These error-indicators provide a measure of the numerical error in the solution. Peclet number, the ratio of the convection to the diffusion term, is generally considered to be indicative of the "numerical diffusion" in the solution. The Courant number, the ratio of the convection to the accumulation term, is generally considered to be indicative of the "dispersion" in the transient component of the solution. These error indicators are based on the Taylor series analysis of the governing equation and are known to be inadequate for prediction of numerical error in many practical situations. Therefore caution should be exercised in the interpretation of these error indicators. Peclet number can not be defined for the pressure equation since it does not have a convection term. Similarly, no Courant number can be defined for steady state computation since the accumulation term is identically equal to zero.

EXAMPLES**DEBU** output of error indicators for T**DEBU** output for T every 12 steps**DEBU** output for T for SELEcted subregion at TIME interval of 1.5 units**DEBU** output for T for subregion with ID=SUBZone at TIME interval of 1.5 units**DEBU** specified output: for T on 'MYFILE.DBG'

MODE 3: Diagnostic Output of Matrix Coefficients

SYNTAX DEBU {MATR} [N1]

MATR The debug output pertaining to the matrix of the algebraic equations is produced on a file at the computation step specified by the N1 input. The governing algebraic matrix equation

$$A_{ij} X_j = B_i$$

is:

where, for the i^{th} element, A_{ij} are the matrix coefficients, X_j are the values of the dependent variable and B_i is the forcing function.

One file is generated for each governing transport equation. Each file is named `acr_MTRXxxxx.TMP` where “xxxx” are replaced by the non-blank characters of the 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] [C_D] [α] [N1, Nk], [Nk+1]

ϕ See Mode 1 specification.

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

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

option Options selected for implementation of the source.

option	INTERPRETATION
VOLU	The coefficient C_ϕ in the S_ϕ term is set equal to the volume (δV) of the element.
AREA	The coefficient C_ϕ in the S_ϕ term is set equal to the area (δA) of the element face indicated by the dir modifier.
dir	The orientation index for the element boundary associated with the source. See Section 7.2.5 for available choices. There is no default value for this input.
NORM	The coefficient C_ϕ in the S_ϕ term is computed as: $C_\phi = \sum_i A_i \cdot V_i$ 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**

C_D The drag coefficient, C_D , in the drag law (see Comments). **The default value is 0.**

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

N1, ..., Nk The normalizing vector components, V_i , if the **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

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

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: **Generic Functional Form for Density**

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

func One of the modifiers listed in Table 7.2.4 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 7.2.5. **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 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/m³

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 2: Pre-Defined Density Functions

SYNTAX **DENS** {option} [INCO | COMP] [phase] [N1, N2, N3, N4]

option The density option modifier.

POWE Density varies according to the power law Equation 3.1.1.

POLY Density varies according to the polynomial Equation 3.1.2.

LINE Density varies according to the linear Equation 3.1.3.

GAS Density varies according to the gas law Equation 3.1.4.

INCO The incompressible form of the gas law is used where the local pressure, p , is ignored in comparison to p^* in Equation 3.1.4. This modifier is effective only if the **GAS** modifier is present. This is the default option for all ACRi Software Tools except **PORFLOW™**.

COMP The compressible form of the gas law, Equation 3.1.4, is used. The local pressure, p , is added to the reference pressure, p^* , in computing density. This modifier is effective only if the **GAS** modifier is present. This is the default option for the **PORFLOW™** Software Tool

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

N1 Reference density for the fluid. For non-gaseous phases, the default value is 1. For the gas phase, the default value is computed from the reference pressure, temperature and gas properties

N2 Coefficient a_1 of Equations 3.1.1 through 3.1.3. The default value is 0.2 for Equation 3.1.1, and 0 otherwise

N3 The critical temperature, T_c , of Equation 3.1.1. The coefficient, a_2 , of Equation 3.1.2. The reference concentration, C^* , of Equation 3.1.3. The default value is 374.15 for Equation 3.1.1 and 0 otherwise

N4 Coefficient, a_3 , of Equation 3.1.2 or a_2 of Equation 3.1.3. The default value is 0.

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. In general, the **DENSity** command is recommended as the preferred mode for input of density.

EXAMPLES

DENSity by **POWER** law: $\rho^* = 1.$, Exponent = 0.25, $T_c = 374.15$ K

DENSity **POLY**nomial law: $\rho^* = 1000$, $a=1000.$, $b=0.05$, $c=0.$, $d=3.E-5$

DENSity **LINE**ar function: 997., Beta=1.0E-4

DENSity **LINE**ar function 789., Beta=1.0E-4 for **SECO**nd phase

DENSity from **GAS** law: reference value = 0.960 kg/m³

DENSity from **GAS** law reference value computed from other input

DENSity **GAS** law in **INCO**mpressible mode

DENSity **USER**-defined function # 7

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**] {N1, ..., Nn} [Nn+1] [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.**

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.

Nn+1 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.**

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 user 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'
DIAGnostics at (7,2,5) every 10 steps
DIAGnostic output for U, P and T at node (7,2) print every 10 steps; also 'EQUATIONS.DGN'
DIAGnostic output for TIME, U, BP and RT at node (7,2) print every 10 steps
DIAGnostic output for TIME, DTIME, P, DP & RP at node (7,2) every 10 steps
DIAGnostic output: U, BP, RT and TIME in that order at (7,2) every 10 steps
DIAGnostic output: TIME, U, BP and RT at node (7,2) print every 10 steps
DIAGnostic output 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] [DIVE] [ENTH] [OUTF] [VELO] [FLUX] [DPDX] [DENS] [ALL]
FLOW	By default, the equations for the velocity components U, V, W and the density/pressure correction variable, ϕ , are always solved. This modifier may be used to disable flow calculations. Flow computation may also be disabled by appropriate specification of variables on the SOLVE command.
DIVE	For steady state solution , by default a term equal to divergence times the local value of the variable is added (or subtracted) to the equations. Under most conditions of poor initial values, this enhances numerical stability. However, in some cases, it may be desirable to not add this term to the equations. This modifier may be used to disable this feature. This modifier has no effect for transient mode of solution.
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.
OUTF	In incompressible flow mode, the flow at OUTlet boundaries is always corrected so that the total outflow equals the net inflow into the domain. For incompressible flow, this is dictated by the continuity equation. If this modifier is present, then this correction is not performed. This option typically leads to slow convergence, but may be useful if the initial conditions are very poor. This modifier is active only for the ANSWER™ software.
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.
FLUX	By default the mass flux at element boundaries is corrected based on the pressure gradient. If this modifier is present, then mass flux at element boundaries is computed from algebraic average of velocity at the element nodes; correction based on pressure gradient is not carried out. This modifier is active only for the ANSWER™ software.
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.
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.
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.

EXAMPLES

DISAble FLOW

DISAble FLOW computations

DISAble ENTHalpy values read from the file for ANSWER software

DISAble DIVERgence term

DISAble FLOW and DIVERgence term

DISAble VELOCity and FLUX correction for ANSWER

DISAble DENSity variations in flow equation (Boussinesq assumption)

DISAble DENSity variations in ALL equations for PORFLOW

DISAble FLOW and DENSity variation in ALL equation

MODE 2: Disable Options for Specific Variables**SYNTAX** **DISA** { Φ } [CONV] [DIFF] [STOR] [OFF] Φ **One or more symbols** that denote the dependent variables for which the specified feature is disabled. **There is no default value.****CONV** In the absence of the **OFF** modifier, the convection term in the transport equation for the specified variable is set to zero. If the **OFF** modifier is present, then the convection term is reinstated.**DIFF** In the absence of the **OFF** modifier, the diffusion term in the transport equation for the specified variable is set to zero. If the **OFF** modifier is present, then the diffusion term is reinstated.**STOR** In the absence of the **OFF** modifier, the storage or accumulation term in the transport equation for the specified variable is set to zero. **This is equivalent to solving the steady state form of the equation.** If the **OFF** modifier is present, then the storage 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**DISA**ble CONVection term for T and C**DISA**ble DIFFusion and STORage term for U**DISA**ble CONVection and STORage terms for T**DISA**ble CONVection term for T is OFF (reinstate convection)**DISA**ble STORage term for T and C (same as steady state)**DISA**ble STORage term for T and C OFF (reinstate transient state)**COMMAND** **DISTRIBUTION COEFFICIENT****PURPOSE** To specify the distribution coefficient for the governing differential equations.**MODE 1: Distribution Coefficient as a General Function****SYNTAX** **DIST** { Φ } {func [ξ]} [subrgn] [fname | N1 ..., Nn] Φ The symbol for the dependent variable for which the distribution coefficient is specified. Valid symbols are listed in Table 6.7.1. **There is no default value; a symbol 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.****func** One of the modifiers listed in Table 7.2.4 that denotes the functional form of the input. For this input, the function specifies the value of the appropriate distribution coefficient for the corresponding \square variable. **If no function is specified then the value is assumed to be**

- constant.
- ξ One of the independent variables listed in Table 7.2.5. If no variable is specified, then the independent variable is assumed to be time.
- 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 7.2.6 for more details. There are no default values for this input.

COMMENTS

The term “distribution coefficient” is normally applied only to the transport of chemical species. However, this command may be used to specify the “storage coefficient” for any of the governing transport equations (see Chapter 2). **This command is implemented only by special arrangement. Please consult ACRI for the availability of this command for your installation of the ACRI software.**

EXAMPLES

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

MODE 2: Distribution Coefficient As Special Functions

SYNTAX **DIST** {Φ} {FREU | LANG} [subrgn] [N1] [N2]

Φ See Mode 1 specification.

FREU The concentration of the species in the solid phase, C_s , is given by the Freundlich Isotherm:

$$C_s = \rho_s k_d C^n,$$

where ρ_s is the density of solid, k_d is a distribution coefficient, C is the concentration in the liquid phase and n is an empirical exponent.

LANG The concentration of the species in the solid phase, C_s , is given by the Langmuir equation

$$C_s = \rho_s c_{smax} \frac{C}{C_{lang} + C},$$

where c_{smax} is the maximum concentration per unit mass in the solid phase and C_{lang} is the Langmuir constant. It is seen that when $C \ll C_{lang}$, the relation become equivalent to a linear isotherm with $k_d = c_{smax} / C_{lang}$.

subrgn See Mode 1 specification.

N1 The distribution coefficient, k_d , for the Freundlich Isotherm or the c_{smax} for the Langmuir equation.

N2 The power exponent, n , for the Freundlich Isotherm or the Langmuir constant, C_{lang} , for the Langmuir equation.

COMMENT

This command mode is available only with the **PORFLOW™** Simulation Tool for all the species equations and for the first phase pressure equation. It is [one of the means](#) available in **PORFLOW™** to incorporate non-linear isotherms with geochemistry. Non-linear kinetics can also be implemented by functional forms of the distribution or retardation coefficients (see Mode 1 of **DISTRibution** and **RETArdateion** commands) and by the **REACTION** command.

For the first phase pressure equation in **PORFLOW™**, the variable the species concentration, C , is replaced by P , the pressure. The units of k_d and c_{smax} are such that the units of C_s are those of volumetric unit of adsorbed gas in the solid phase divided by the mass density of solid. C_{lang} has units of pressure, P .

EXAMPLES

DISTRibute C as FREUdlich Isotherm with $k_d=0.22$, $n = 0.8$ for ID=CLAY

DISTRibute C as LANGmuir equation with $C_smax=1.89$, $C_langmuir = 0.2$ for ID=CLAY

MODE 3: **Distribution Coefficient from Fluid & Solid Phase Concentrations**

SYNTAX **DIST {Φ} {CONC} {TABL} [subrgn] [fname | N1 ..., Nn]**

Φ See Mode 1 specification.

CONC The distribution coefficient is computed from a specified table of concentration in the fluid phase as a function of the concentration in the solid. The total initial property can be specified by the **SET** command with **INVEntory** modifier. **If the initial inventory is not specified, then it is assumed that the initial amount in each element is equal to the initial concentration multiplied by the volume of the fluid in that element. The specified fluid concentration is then modified so that initial amount is distributed between the fluid and solid phases according to the specified tabular function.**

TABL The data is specified as a table of two values per data set. The first value for each set is the concentration in the solid phase and the second value is that in the liquid phase.

subrgn See Mode 1 specification.

fname See Mode 1 specification.

N1, ..., Nn See Mode 1 specification.

COMMENT

This mode of the command is available only with the **PORFLOW™** Simulation Tool. It is **one of the means** available in **PORFLOW™** to incorporate non-linear isotherms with geochemistry. Non-linear kinetics can also be implemented by functional forms of the distribution or retardation coefficients (see Mode 1 of **DISTribution** and **RETArdation** commands) and by the **REACTION** command.

EXAMPLES

DISTribute C as **CONC**entration **TABL**e of values: 6 sets

(0., 0.) (1., 0.), (1.0001, 1.), (2., 1.) (2.0001, 0.5) (3.,0.2)

DISTribute C2 in **CONC**entration mode for ID=ZN10 as **TABL**e of values: with 6 sets:

(0., 0.) (1., 0.), (1.0001, 1.), (2., 1.) (2.0001, 0.5) (3.,0.2)

DIST C **CONC**entration as **TABL**e of 20 sets from file 'KD_ALL' for ID=ZN10.

MODE 4: **Disable Previously Specified DIST Commands**

SYNTAX **DIST {Φ} {OFF} {ID=idsub}**

Φ See Mode 1 specification.

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

idsub Identifier for the subregion that appeared previously with the same identity in a **LOCA**te or **SELE**ct command.

EXAMPLES

DIST commands for C for ID=MIDDLE OFF

COMMAND **EBU**

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

MODE 1: **EBU Reaction Variables and Parameters**

SYNTAX **EBU** {idreac} {Φ} [HARM] [C_{EBU}, α, β] [γ]

idreac Character string identifying the **idreac** reaction on the **REACTION** command that is limited by the eddy breakup (EBU) limit. To activate this limit, the modifier EBU must also appear in the corresponding **REACTION** command. The EBU reaction rate is computed from:

$$R_{EBU} = C_{EBU} \rho \frac{\epsilon}{k} \Phi_{ebu}^{rms}$$

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

If R_{KIN} is the kinetic reaction rate (see **REACTION** command), then the actual EBU-limited reaction rate for the **idreac** is defined by one of the following two options:

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

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

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

Φ One or more of the symbols in Table 6.8.1 which identify the EBU limiting species (Φ_1 , Φ_2 , or Φ_3). If no symbol is specified then the species participating in the corresponding **REACTION** command are used as the limiting species.

HARM By default, the actual reaction rate is computed as the smaller of the kinetic and EBU reactions. If this modifier is present then the harmonic limiter is used.

C_{EBU} The empirical constant, C_{EBU} , in the R_{EBU} above. The default value is 3.

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

γ The constant γ in the harmonic limiter as described below. It is the 2nd, 3rd or 4th numerical value on the command depending on whether 1, 2 or 3 symbols define the EBU model. The default value is 0.5.

EXAMPLES

EBU for R1 function of FU with constant = 3.0

EBU for R2 function of CH and O2, values = 3.0 and 0.3

EBU for R2 with CH and O2, 3.0, 0.3 with **HARMonic** limiter 0.75

EBU for R2 with FU CH and O2, 3.0, 0.5, 0.3 with **HARMonic** limiter 0.75

MODE 2: **Global EBU Reaction Constants****SYNTAX** **EBU** [C_{EBU}] [I_{EBU}]

C_{EBU} The global empirical constant, C_{EBU} , for the eddy breakup relations. This value is used if a value is not explicitly specified with a Mode 1 **EBU** command for a specific **idreac** reaction. *The default value is 3.*

I_{EBU} The first step of the solution procedure at which the EBU limit is invoked. Often at the initial stages of a solution process, the flow and turbulence fields are not well developed. In this case, it is preferable to use the kinetic reaction rate (the maximum possible) during the initial stage. *The default value is 10.*

EXAMPLES

EBU limit global rate constant = 3.2

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

COMMAND **ELSE**

PURPOSE To define an IF construct to control which, if any, of one or two blocks of user input statements are executed.

SYNTAX **ELSE**

COMMENTS _____

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

COMMAND **END**

PURPOSE To signify the end of a problem.

SYNTAX **END**

COMMENTS

This command signifies the end of a problem. The input for a new problem may be continued with a new problem specification after this command. For each problem, this command must be employed as the last command. Failure to do so may cause a loss of some or all of the data and output files, depending on the host operating system.

EXAMPLES

END

END of problem number 1

COMMAND **ENDIF**

PURPOSE To define an IF construct to control which, if any, of one or two blocks of user input statements are executed.

SYNTAX **ENDIF**

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 **CLOSE** modifier is present.

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

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

filetype	Interpretation
DEBU	Debug output file (DEBUg), Unit 17, is selected for the operation.
FLUX	Flux output file (FLUX), Unit 14, is selected for the operation.
HIST	Time History file (HISTory), Unit 13, is selected for the operation.
SAVE	Archive file (SAVE), Unit 11, is selected for the operation.
TABL	Tabulated archive file (SAVE TABLE), Unit 12, is selected for the operation.
TRAC	Particle track file (TRACK), Unit 18, is selected for the operation.

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

EXAMPLES

- FILE OPEN 'mynewfile.now' on 37
- FILE OPEN OLD file 'mydatafile.now' on unit 37 in UNFOrmatted READ only mode
- FILE OPEN NEW SAVE file 'mynewsavfile.now' on in FORMatted WRITe only mode
- FILE CLOSe SAVE HISTory file now
- FILE CLOSe file by name 'OLDFILE.TMP'
- FILE CLOSe unit 11 !! same as archive SAVE file

COMMAND **FIX**

PURPOSE To fix the values or the matrix coefficients of selected variables for a subregion within the domain of calculation.

MODE 1: **Fix Variable Values**

SYNTAX **FIX** [Φ] [subrgn]

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

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

COMMENTS

This command can be used only to fix the values of a variable **inside** the domain of computation. **The values at the domain boundary cannot be fixed by this command** (the boundary values can be fixed by the **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:

$$\Phi_p^n = N1 + N2\Phi_p^o + N3\Phi_W^n + N4\Phi_E^n + N5\Phi_S^n + N6\Phi_N^n$$

where superscript 'o' denotes the existing value and 'n' denotes the new computed value. The subscripts P, W, E, S and N denote the node being computed and the nearest neighboring nodes in the X-, X+, Y- and Y+ directions, respectively. For 3D simulation, the formula is:

$$\Phi_p^n = N1 + N2\Phi_p^o + N3\Phi_W^n + N4\Phi_E^n + N5\Phi_S^n + N6\Phi_N^n + N7\Phi_D^n + N8\Phi_U^n$$

where D and U denote the neighboring nodes in the Z- and Z+ directions (Figures 4.1.3 and 4.1.4), respectively.

COMMENTS

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

OFF Previously specified **FIX** commands for Φ , for the identified subregion, are deactivated. A new specification for this subregion may follow. **This command can only be used for previous **FIX** commands that appeared with no Φ symbol or only a single Φ symbol.**

Φ A symbol that denotes the dependent variable for which the input is specified. The valid symbols are listed in Table 6.7.1. **There is no default value.**

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

EXAMPLES

FIX T in ID=FIXD **OFF**

FIX T & P **OFF** for ID=FIX1

FIX all **OFF** in ID=FIXAll

COMMAND FLOW

PURPOSE To specify the flow rate and other dependent variables at a boundary.

MODE 1: Flow Injection or Withdrawal with Fixed Variable Values

SYNTAX **FLOW** [TOTAL] [func (ξ)] [option] [subgrn] {dir} {N1}[fname|N2,...,Nn] [φ=Nn+1, ..., φ=Nm] [Nm+1, Nk] [ρB]

TOTA By default, the amount of flow specified, or computed from func (ξ), is applied to each element of the subgrn. If this modifier is present then the amount is assumed to be the total amount over whole of the subgrn. In this case, the amount is distributed equally to all the elements of the subgrn unless the VOLUME or AREA modifiers are present.

func One of the modifiers listed in Table 7.2.4 that denotes the functional form of the flow rate. If no function is specified, the value is assumed to be constant.

ξ One of the independent variables listed in Table 7.2.5. If no independent variable is specified, the variable is assumed to be time.

option Options selected for implementation of the source.

option	INTERPRETATION
VOLU	In the absence of the TOTAL modifier, the flow for each element is computed as: $Q = q \delta V$. Here q is the amount specified by the user and δV is the volume of the element. The q , in turn, is computed from $func(\xi)$ 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 subgrn .
AREA	In the absence of the TOTAL modifier, the source for each element is computed as: $Q = q \delta A$, where δA is the area of the element boundary indicated by dir . 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 subgrn in the dir direction.
INTE	By default, if dir points to a wall, then any special treatment for that wall (such as wall function for turbulent flow) is deactivated and the diffusive flux at the wall is set to zero. If the INTERNAL modifier is present then the wall treatment and wall diffusive flux are retained.
NORM	In the absence of the TOTAL modifier, the source, Q , is computed as: $Q = q \sum_i A_i \cdot V_i$ where A_i is the i^{th} direction component of the element boundary area specified by 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, ρ_B , on the command. The boundary value of density is overwritten by the specified value. If this value is omitted, then the existing boundary value at the node indicated by the dir direction is used.

- subrgn** The subregion for which the input is specified. If no subregion is specified, then entire computational domain is selected.
- dir** The orientation index for the element boundary associated with the source. See Section 7.2.5 for available choices. There is no default value for this input.
- fname** The name of the file from which numerical values **N2** through **Nn** are read. This option can be used only if the selected function is a table or one of the series functions. See Section 7.2.2 for further information.
- N1, ..., Nn** The numerical constants and coefficients for the selected function. See Section 7.2.7 for more details. There are no default values for this input.
- Nn+1, ..., Nk** The normalizing vector components, V_i , if the **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_B** The density value that multiplies the computed source. It can only be specified if the **DENSITY** modifier is present and then it must be the last value on the command. If no value is specified but the modifier **DENSITY** is present, then default value is the boundary value at the node indicated by the **dir** modifier.

COMMENTS

The **FLOW** command is primarily designed to allow injection of fluid through a solid boundary (or blocked region) inside the domain of computation. Its use is most appropriate when the scale of injection is comparable to the grid size. The command is implemented in terms of boundary “fluxes” entering the domain of computations. If the scale of injection is much smaller than the grid size, then the **SOURCE FLOW** command may be more appropriate. The injection at an exterior domain boundary is more appropriately accounted for by the **INLET** command.

In implementing this command it is assumed that at the point of injection, the boundary wall has essentially been removed. The computational treatment is akin to that for inflow such as through an **INLET** boundary. If the injection occurs through a blocked element or at an exterior boundary of the domain, then the injected values are also assumed to be the boundary values of the variables.

If the wall needs to be retained, such as for flow percolating through a solid matrix or through holes much smaller than the boundary area of the inflow element, then the **INTERNAL** modifier should be used with the command.

EXAMPLES

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

- FLOW** injection: amount = 0.001 with T=50, C=1.
- FLOW** per unit AREA of X- face: = 0.001 with T=50, C=1.
- FLOW** withdrawal: amount =-0.001 per second
- FLOW** TABLE 3 sets SELEcted (0, 0), (100, 1), (200, 0) U=1, V=0.1, W=0., T=10, K=0.001, L=1
- FLOW**: EXPONential series with TIME 7 sets from 'SOURCE' T=100, C=0.
- FLOW** q=10 X- direction T=100, U=20. NORMALized velocity 0., 1.5, 2.5 ID=VSOURce;
- FLOW** q= -10 in X- dir with NORMALized vel 1., 1.5, 2.5 and DENSity for ID=VSOURce
- FLOW** q=10 X- dir variable values: U=10, V=0, W=-20, T=100, K=0.03, L=0.5 INTERNAL with NORMALized vel 1., 1.5, 2.5 and DENSity for ID=VSOURce:

FLOW q=10 per unit VOLUME injected variables: U=10, V=0, W=-20, T=100, K=0.03, L=0.5 multiply by
DENSITY = 5 for SELEcted region

MODE 2: Flow Injection with Fixed Variable Values and Computed Momentum Components

SYNTAX **FLOW** {MOME} [func (ξ)] [TOTA] [option] [subrgn] {dir} {N1} [fname|N2,...,Nn] [φ=Nn+1, ..., φ=Nm] [Nm+1, Nk] [ρ_B]

MOME The velocity components of the injected flow are computed from the flow rate for the element, Q, as:

$$V_j = \frac{Q}{\rho_B A} n_j,$$

where V_j are the computed velocity components of the injected flow in the j^{th} direction, ρ_B is the density, A is the area of the element boundary specified by the **dir** modifier, and n_j is a normalizing vector. If the **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 NORMAL modifier is present, it is assumed that the AREA modifier is in effect.** Any specification of the **VOLUME** modifier is ignored. All other modifiers can be used as for Mode 1 Specification.
- 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 **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_B** The density value to compute the velocity components. It can only be specified if the **DENSITY** modifier is present and then it must be the last value on the command. **If no value is specified, then boundary value at the node indicated by the dir is used.**

EXAMPLES

All the examples cited for Mode 2, except those with **VOLUME** modifier, are applicable provided that the modifier **MOMEntum** is added. Some illustrative examples specific to this mode are given below.

- FLOW** with MOMEntum q=10 X- direction with ID=VSOURce; T=100, K=0.05, L=0.02
- FLOW** MOMEntum q=10 X- direction with ID=VSOURce; T=100, K=0.05, L=0.02, DENSity 5
- FLOW** MOMEntum q=10 X- direction with ID=VSOURce; T=100, K=0.05, L=0.02, NORMal 1. -1, 0
- FLOW** MOMEntum q=10 X- ID=VSOURce; T=100, K=0.05, L=0.02, NORMal 1. -1, 0, DENSity=5.
- FLOW** MOMEntum: TABLE 2 sets SELEct (0, 0), (100, 1) T=10, K=0.001, L=1

FLOW MOMEntum EXPOnential SERIes TIME 7 sets 'SOURCE' T=100, C=0. NORMAl 1. -1 SELEcte
FLOW MOMEntum q=10 X- dir INTernal for ID=VSOURce: injected variables: T=100, K=0.03, L=0.5
NORMAlized vel 1., 1.5, -0.7 and DENSity as exists

MODE 3: Flow Injection at Fixed Spherical Angles with Computed Momentum Components

SYNTAX **FLOW** {ANGL} [func (ξ)] [TOTA] [subrgn] {dir} {N1}[fname|N2,..,Nn] [φ=Nn+1,..,φ=Nm] [Nm+1, ..., Nm+5] [DENS | SPEE] [ρ_B | V_S]

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

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

where ρ_B is the density at the neighboring element and A is the area of the element boundary specified by the dir modifier. The local direction vector, n_i, for each segment of the surface is obtained from the user input of two spherical angles and a reference axis.

- func** See Mode 1 specification.
- TOTA** See Mode 1 specification
- subrgn** See Mode 1 specification.
- dir** The orientation index for the boundary associated with the source. See Section 7.2.5 for available choices. *A value must be specified; there is no default value.*
- fname** See Mode 1 specification.
- φ** See Mode 1 specification.
- N1, ..., Nn** See Mode 1 specification.
- Nn+1, .. Nm,** These values represent the value of injected variable denoted by the symbol immediately preceding the value.
- Nm+1** The spherical angle, θ in degrees measured as the deflection of the injection vector from the face normal. The magnitude of the angle must be less than 180 degrees.
- Nm+2** The spherical angle, φ in degrees measured as the rotation of the injection vector with respect to the projection of the axis vector onto the face plane.
- Nm+3, Nm+5** The direction cosines of the axis vector used to measure the angle φ; 3 values must be specified since this option is only available for 3D flows.
- DENS** The density, ρ_B, is specified as the last value on the command.
- ρ_B** See Mode 3 specification.
- SPEE** The injection speed, V_s, is specified as the last value on the command.
- V_s** The value V_s if the **SPEED** modifier is present. *There is no default value for this input.*

EXAMPLES

All examples cited for Mode 1 are applicable provided that the modifier **ANGLE** is added and appropriate input for spherical angles is appended. Some illustrative examples of the use of attributes specific to this mode are given below.

FLOW q=10., X- direction with ID=VSOURce; T=100, K=0.05, L=0.02, Theta=45, phi=20, axis=1., 0, 0

FLOW q=10., X- ID=VSOURce; T=100, K=0.05, L=0.02, Theta=45, phi=20, axis=1., 0. 0, SPEEd=120
FLOW q=10., X- ID=VSOURce; T=100, K=0.05, L=0.02, Theta=45, phi=20, axis=1., 0. 0, DENsity=2.5

MODE 4: **Flow Injection with Multiple Tabular Functions**

SYNTAX **FLOW** { Φ } {TABL} {MULT} [option] [subrgn] {N1} [fname | N2, ..,Nn]

- Φ **One or more** of the symbols that denote the dependent variables which are specified as functions of time. **There is no default value; at least one symbol must be specified.**
- TABL** The variables are specified as tabular functions of time. **This is the only available option in this mode.**
- MULT** Multiple variables are tabular functions of time.
- option** See Mode 1 specification.
- subrgn** See Mode 1 specification.
- fname** See Mode 1 specification.
- N1** The number of sets of data for the tabular functions.
- N2, ..,Nn** The N1 sets of data for the flow rate and variables as tabular functions of time. Each set must consist of time, flow rate, and one value for each variable specified by the symbol Φ on the command, in that order. Thus if 4 variables are selected, then each data set must consist of 6 values. **There is no default value; the correct number of values must be specified.** If the flow rate is negative (withdrawal of fluid), then specified value of the property is ignored since it is assumed that the fluid is being withdrawn with an amount of property equal to the local value of that property.

EXAMPLES

FLOW with MULTiple TABLe functions: 4 sets

Time	Flow	U	V	T
0.	0.001	1.00	0.02	100
1.	0.002	0.50	0.01	200
2.	0.004	2.00	0.01	500
5.	0.010	5.00	0.02	600

FLOW with MULTiple TABLe: 4 sets per unit VOLUme

Time	Flow	U	V	T
0.		0.001	1.00	0.02 100
1.		0.002	0.50	0.01 200
2.		0.004	2.00	0.01 500
5.		0.010	5.00	0.02 600

MODE 5: **Flow Injection or Withdrawal at a Boundary**

SYNTAX **FLOW** [ONLY] [func (ξ)] [option] [subrgn] {N1} [fname | N2, ...,Nn]

ONLY By default the injected (or withdrawn) flow is assumed to be accompanied by a specification of all other variables (properties) that are assumed to be automatically injected with it (see Mode 1 specification). If this modifier is present, then it is assumed that only the flow rate is specified by this command. The boundary flux of other variables is computed by a product of the injected flow and the prevailing value of the variable at the boundary node.

func See Mode 1 specification.

ξ See Mode 1 specification.

option See Mode 1 specification.

subrgn See Mode 1 specification.

fname See Mode 1 specification.

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

EXAMPLES

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

FLOW T is TABLE per unit AREA in X- direction: 3 sets (TIME, value) (0., 0.01), (100., 0.10), (200, -0.20)

FLOW for T 10 per unit AREA in X- direction for SELEcted region

FLOW for T 10 per unit area in X- direction with NORMAlized velocity of 0., 1.5, 2.5 ID=VFLOW

FLOW for T 10 in X- dir with NORMAlized vel 1., 1.5, 2.5 and DENSity for ID=VFLOW

MODE 6: **Disable Previously Specified FLOW commands**

SYNTAX **FLOW {OFF} {subrgn}**

Φ See Mode 1 specification.

OFF Previous **FLOW** commands for the identified subregion are deactivated. A new specification may follow.

subrgn See Mode 1 specification.

EXAMPLES

FLOW OFF for T for most recently SELEcted region

FLOW OFF for T for ID=MIDDLE

COMMAND FLUID

PURPOSE To specify thermal and transport properties of the fluid.

MODE 1: **Specific Heat of the Fluid**

SYNTAX **FLUI** {SPEC} [phase] {N1} [N2, N3, N4]

SPEC Input is for the specific heat of the fluid.

phase The fluid phase for which the input is specified. See Section 7.2.10 for available options. If no phase modifier exists, then the input is assumed to be for the 1st phase of the fluid.

N1 For the 1st fluid phase, the specific heat of the fluid. The default value is unity.

For the second and third fluid phases, the interpretation of this input is problem-dependent. In the absence of evaporation (see **PROBLEM** command), the property specified is the specific heat of the fluid. The default value is zero.

If the evaporation option is selected, the 2nd phase is taken to be a gas phase which may be composed of up to four different species (or components). In this instance, the first gas species is always taken to be water vapor and N1 is the specific heat of the water vapor. The default value in this instance is 1860.

N2, N3,N4 This input is used only for problems with evaporation if the **SECONd** modifier is present. In this instance these values are the specific heats of the 2nd, 3rd and 4th species of the gas phase, respectively. The default value for all components is 1000.

EXAMPLES

FLUId SPECific heat for: 4.19

FLUId SPECific heat for SECONd fluid: 1.

FLUId SPECific heat for SECONd fluid: 0.24, 0.24, 0.24, 0.24

MODE 2: Thermal Conductivity or Mass Diffusivity of the Fluid**SYNTAX** **FLUI** [Φ] {COND | DIFF} [phase] {N1}

Φ One of the symbolic character strings which denotes the dependent variable for which the input is specified. The valid symbols are: T, C, C2, C3 or C4 of Table 6.7.1. The symbol is ignored if the COND modifier is present. If the DIFF modifier is present and no symbol is specified, the default is the 1st chemical species.

COND The input is for the thermal conductivity of the fluid.

DIFF The input is for the mass diffusivity of the fluid.

phase The fluid phase for which the input is specified. See Section 7.2.10 for available options. If no phase modifier exists, the input is assumed to be for the 1st phase of the fluid.

N1 The thermal conductivity of the fluid or the mass diffusivity of the species in the fluid. The default value is zero.

EXAMPLES

FLUId thermal **COND**uctivity = 4.10

FLUId diffusivity for T = 4.10

FLUId **DIFF**usivity for first species = 4.10

FLUId **DIFF**usivity for C2 = 4.10

FLUId **DIFF**usivity for C2 in **SECO**nd phase = 4.10

MODE 3: Fluid Compressibility

SYNTAX FLUI {COMP} [phase] [ACTI] { α_f }

COMP The fluid phase compressibility; α_f of Equation 3.4.1, is specified.

phase The fluid phase for which the input is specified. See Section 7.2.10 for available options. If no phase modifier exists, the input is assumed to be for the 1st phase of the fluid.

ACTI This modifier is meaningful only for multi-phase flow simulation. By default the storage term accounts for the compressibility only of the phase for which the equation is being solved. However, if this modifier is present, the storage term also accounts for the changes in volume due to compression or expansion of the phase immediately “above” the one being simulated. In this case a term equal to:

$$R^n \alpha_f^{n+1} (1 - S^n) \Theta_E$$

is added to the effective storage term, S_e of Equation 3.4.1, for the nth phase pressure equation. In 3 phase systems (such as water/oil/gas), this effect can only be accounted for in the 1st and 2nd phases.

In many situations, it is appropriate to account for the compressibility of the phase “above” the current phase. For example, in a water/air system, when the water pressure increases, the pressure of the air phase may also increase due to capillary pressure effects. This compresses the air and additional volume is now available for storage of water. The effect is akin to the “matrix compressibility” or “matrix storage” in confined aquifer systems (see HYDRaulic command).

This effect can be very important in “dry” conditions when the saturation of a phase is zero or negligible. In this case, based on the saturation characteristic specified, both the total compressibility and the hydraulic conductivity may be zero. For example, this is a consequence of the common van Genuchten characteristics. Thus mathematically the soil can neither absorb nor conduct water (fluid). Obviously this is a mathematical artifact due the assumption of the saturation characteristic and has no basis in physics. Actually, in such dry conditions one must account for the vapor phase and other coupled effects.

This modifier, thus allows for the effect of “gas” or multi-phase compressibility to be accounted for in dry conditions. As the phase saturation approaches unity, this effect automatically disappears.

α_f The compressibility of the fluid; α_f of Equation 3.4.1. The default value is zero.

EXAMPLES

- FLUId COMPRESSibility is 5.E-10 for water
- FLUId COMPRESSibility for SECONd phase is 1.E-5
- FLUId COMPRESSibility for SECONd phase is 1.E-5 in ACTIve mode

MODE 4: Fluid Enthalpy or Latent Heat

SYNTAX **FLUI** {ENTH | LATE} {N1} [N2, N3, N4]

ENTH Enthalpy or latent heat constants for phase change are specified.

LATE Same as the modifier **ENTH**, above.

N1 The latent heat of phase change; the constant, c_0 , of Equation 3.6.3. For the evaporation option, the default value is 3.9202×10^6 ; for the freezing option, it is 1000.

N2, N3, N4 The enthalpy-temperature constants, c_1 , c_2 and c_3 of Equation 3.6.3, respectively. The default values are: $c_1 = -2441$ (evaporation), $c_1 = 1$ (freezing), and $c_2 = c_3 = 0$.

EXAMPLES

FLUId ENTHalpy constants: $h_0=3.92017E6$, $c_1=3681$, $c_2=-11.76$, $c_3=0.01772$

FLUId ENTHalpy constants: 1000 (latent heat), $c_1 = 0.5$

FLUId LATEnt heat of freezing is 1000

MODE 5: Fluid Density

SYNTAX FLUI {DENS}

DENS Density of the fluid is specified. This mode of the command has been replaced by the **DENSity** command. It has been retained only for compatibility with legacy data sets. Its use is no longer recommended because this command will not be available in future versions of PORFLOW™ Software.

COMMENTS

Any input specified here will be automatically interpreted as that for the **DENSity** command.

EXAMPLES

Please see the **DENSity** command for examples

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 of Flux Balance for a Variable**

SYNTAX **FLUX** { Φ } [subrgn] [dir] [NOW] [fname] [TIME] [$V_{\text{frq_file}}$, V_{frq}]

Φ One, and only one, symbol for the dependent variable for which the flux-balance output is required. The valid symbols are listed in Table 6.7.1. There is no default value.

subrgn The subregion for computations. If no subregion is specified, the entire domain is selected.

dir The orientation index for the boundary for which the flux output is required. See Section 7.2.5 for available choices. There is no default value for this input. In this case the convective and diffusive fluxes at the selected boundary are written to both the flux file and the standard output file in a tabulated form at the end of computations.

NOW Fluxes are computed and output is produced immediately.

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 given time 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, $V_{\text{frq_file}}$ and V_{frq} are interpreted to be the frequency of output in terms of number of steps. If this modifier is present, then $V_{\text{frq_file}}$ and V_{frq} are interpreted to be the time interval between successive outputs.

$V_{\text{frq_file}}$ The frequency (in terms of step or time interval) at which the fluxes are written to the flux file specified by 'fname'. In the step interval mode, a specification of 10 will result in output at step number 10, 20, 30, etc. In the time interval mode, whether or not the output is obtained exactly at the specified interval depends on the time step specified by the **SOLVE** command. If an exact time at which the output is due is not simulated, then the output is obtained at the first time step after the due time. 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 $V_{\text{frq_file}}$. The default value is such that a summary of fluxes is obtained only at the end of simulations.

COMMENTS

All fluxes in ACRI Software are defined so that a flux **ENTERING** the computational domain (or a subregion) is **POSITIVE** and that **LEAVING** the domain is **NEGATIVE**.

The flux for mass is reported under the flux balance for pressure (P) since the continuity equation is solved in terms of pressure.

By default, the flux balance for each relevant dependent variable is automatically computed for the whole domain. This command can be used to obtain fluxes for additional subregions, to control the output, and to deactivate the flux computations in a selective manner. If no **FLUX** command is specified, then the summary of the flux balances is still directed to the standard output file at the end of computations. The output to the flux file, on the other hand, is generated only if a **FLUX** command is explicitly specified.

The flux balance output produced by **ACRI** Software Tools consists of cumulative fluxes for the subdomain and instantaneous fluxes crossing the boundaries. The flux balance for a variable is composed of cumulative changes due to storage (Q_O and Q_N), inflow (Q_{in}), outflow (Q_{out}), and decay (Q_{decay}). The flux balance disparity, Q_d , is defined as:

$$Q_d = (Q_O - Q_N) + Q_{in} - Q_{out} - Q_{decay} ,$$

where Q_O is amount of property present originally, Q_N is the amount present now, and Q_{in} , Q_{out} and Q_{decay} are the cumulative inflow, outflow and decay, respectively.

The Q_{in} and Q_{out} are, in turn, each composed of three components:

$$Q = Q_{conv} + Q_{diff} + Q_{so} .$$

Here Q_{conv} and Q_{diff} denote the cumulative contributions due to convective and diffusive fluxes, respectively, and Q_{so} denotes the net input due to sources (inflow – outflow).

All ACRI software allows arbitrary user specified units. The actual units for the reported fluxes therefore depend on the units selected by the user. For an arbitrary property, Φ , the units of the cumulative flux are the units of Φ multiplied by the units of the density (ρ) and the units of volume (L^3). For **PORFLOW™** the fluxes for mass and transport species (but not heat) are further divided by the density of the fluid; therefore these are in volumetric rather than mass units. The units of the instantaneous fluxes are those of the cumulative fluxes divided by units of time.

A more detailed description of each of the components that are reported in the flux output is given in the table that follows on the next page. In this table:

1. Fluxes in Items 4 through 18 are in cumulative units. For flow (pressure) equation these are in terms of mass units (e.g. kg) for **ANSWER™** and **TIDAL™**, and in terms of volume units (e.g. m^3) for **PORFLOW™**. For thermal transport, the flux is reported in units of heat (e.g. J). For transport equation of a species, if the property is expressed in units of kg/m^3 , then the flux balance is in terms of kg.
2. The instantaneous fluxes in Item 19 through 21 are in rate units. The units of these are the units of the cumulative flux expressed per unit time. In fact, the cumulative flux is the time integral of the instantaneous flux.

EXAMPLES

FLUX for U for the entire domain

FLUX for T for SELEcted region every 20 steps

FLUX for V in SELEcted region: file every 20 steps; printer every 4 steps

FLUX balance for T for subregion defined by ID=VAULT every 200 steps

FLUX for T at XY plane defined by ID=PLN1 print every 10 steps

FLUX balance for T NOW for whole of the domain

FLUX balance for T for ID=VAULT OFF

FLUX for U to 'FLUX.OUT' for SELEcted region at TIME interval of 0.4

Terms and Notation Used to Report Flux Balance in Output

#	Term and Notation	Description
1.	Active Subdomain:	Particulars of the Sub-domain for which Flux Balance is given
2.	Subdomain ID:	ID assigned to the Sub-domain by the user or the default ID automatically assigned if none given by User
3.	Time:	The simulation Time at which the flux summary is computed
4.	Net flux disparity (Qin-Qout-Qa-Qdecay)	The total “disparity” or “error” in the subregion. It is a sum of the total inflow into the system minus the total outflow from the system
5.	Total inflow (Qin = Qc_in+Qd_in+Qsor):	Total inflow into the subregion by convective flux (See Item 11), diffusive flux (See item 12) and incoming (or positive) Sources (see Item 13)
6.	Total outflow (Qout = Qc_out+Qd_out+Qsnk):	Total outflow from the subregion by convective flux (Item 14), diffusive flux (Item 15) and Sinks or outgoing (or negative) Sources (Item 16)
7.	Net accumulated gain (Qa = Qn-Qo)	Net increase in the amount of property (storage) in the subregion from the start of simulations.
8.	Decay from start (Qdecay)	Net accumulated decay in the property in the subregion from the start of simulations.
9.	Total initial property in region (Qo)	Initial amount of property present in the subregion at start of simulations
10.	Total property in region now (Qn)	Amount of property currently present in the subregion.
11.	Convective influx (Qc_in):	Net inflow of property due to convection from all boundaries of the subregion from the start of simulations.
12.	Diffusive influx (Qd_in):	Net inflow of property due to diffusion or dispersion from all boundaries of the subregion from the start of simulations.
13.	Source influx (Qsor):	Net inflow of property due to sources in the subregion from the start of simulations.
14.	Convective outflux (Qc_out)	Net outflow of property due to convection from all boundaries of the subregion from the start of simulations.
15.	Diffusive outflux (Qd_out)	Net outflow of property due to diffusion or dispersion from all boundaries of the subregion from the start of simulations.
16.	Sink outflux (Qsnk):	Net outflow of property due to sources or sinks in the subregion from the start of simulations
17.	Flux disparity due to mass balance(div*F)	Error in flux balance due to the fact that the flow field itself may not be mass conservative. This is included in the total disparity reported in Item 4 above. The error in flux balance can be both due to the error in the flow and that in the solution of the transport equation.
18.	Change from last time step	Change in the amount for property from the last time step
19.	Instantaneous convective flux (in-out):	The current net inflow of convective flux at all boundaries of the subregion.
20.	Instantaneous diffusive flux (in-out):	The current net inflow of diffusive flux at all boundaries of the subregion.
21.	Instantaneous source-sink:	The current net inflow of property due to sources in the subregion.

MODE 2: Convective Flux and Flux-Average Value of a Variable

SYNTAX **FLUX** {**AVER**} [ϕ] [**option**] [**subrgn**] [**dir**] [**fname**] [**TIME**] [**NOW**] [V_{frq}] [**OFF**]

AVER The convective flux and the flux-weighted mean of the variable, its average, minimum and maximum values, the flow rate and, area at the selected boundary (or boundaries) are printed to the output device or file.

ϕ One or more symbols for the dependent variable for which output is required. Up to 10 symbols may be specified per command. The valid symbols are listed in Table 6.8.1. If no symbol is specified then output is obtained for each active variables.

option The boundary type for which output is required. More than one option may be selected. If an option is specified, then the **subrgn** and **dir** modifiers are ignored. If no option is specified, then the averages are computed for the specified **subrgn** and **dir** modifiers.

option	INTERPRETATION
INLE	Boundaries specified by the INLEt command are selected.
OUTL	Boundaries specified by the OUTLEt command are selected.
OPEN	Boundaries specified by the OPEN command are selected.
IO	All boundaries specified by INLEt , OUTLEt or OPEN command are selected.
WALL	Walls specified by WALL or BLOCK command are selected.
EXTE	All external (or outer) boundaries of the computational domain are selected.
ALL	All of the above boundaries are selected.

subrgn The subregion for computations. This specification is ignored if any of the **option** modifiers is present. If no subregion is specified, the entire domain is selected.

dir The orientation index for the boundary for which the output is required See Section 7.2.5 for available choices. If no input is given, then the output is obtained for all boundaries of the **subrgn**. This specification is ignored if any of the **option** modifiers is present.

fname The file name for output. If a file name is specified then the output is directed to the named file, otherwise the output is directed to the standard output device. The total number of open files in any simulation can not exceed 64.

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

NOW A record of fluxes is written to the output device at the next time step when the equation for the dependent variable is solved. This is in addition to the output from the V_{frq} specification.

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

OFF Any previously specified **FLUX AVERAGE** commands for the specified variables and subregion are disabled. New commands may be subsequently specified.

EXAMPLES

FLUX AVERAge for ALL boundaries

FLUX AVERAge for T (temperature) at IO boundaries

FLUX AVERAge for T (temperature) at INLEt and OPEN

FLUX AVERAge for T at INLEt and BOUNDaries

FLUX AVERAge for T at ALL on file 'FLUX.FIL'

FLUX AVERAge for T at ALL every 5 steps to file 'FLUX.FIL'

FLUX AVERAge for T at ALL at TIME=0.01 to file 'FLUX.FIL'

FLUX AVERAge to file 'FLUX.FIL' every 1 step

! All active variables by default

! All INLEt, OUTLEt & OPEN bndries

! INLEt & OPEN bndries

! INLEt & domain bndries

MODE 3: **Disable Previously Specified Flux Balance Computations**

SYNTAX **FLUX** {OFF} { ϕ } {subrgn}

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

ϕ The symbol for the dependent variable for which the previously specified flux-balance computation is disabled. The valid symbols are listed in Table 6.7.1. **One and only one character string must be specified for each command.**

subrgn The subregion for flux computations. **If no subregion is specified, the entire domain is selected.**

EXAMPLES

FLUX balance computation OFF for T for whole domain

FLUX computations OFF for T for the currently active subregion

FLUX computations OFF for T for the ID=UPPER

MODE 4: **Disable Default Flux Balance Computations**

SYNTAX **FLUX {OFF}**

OFF By default the flux balance is computed automatically, for the whole of the computational domain, for each variable for which a transport equation is solved. This is in addition to any flux balance computation in response to **FLUX** commands for that variable. If this modifier is present, then the automatic flux computations are disabled with one exception. The exception is the mass balance equation, for which the flux balance is always computed irrespective of the user input.

EXAMPLES

FLUX balance computation OFF for all variables except mass
FLUX default computations OFF
FLUX OFF

COMMAND FOR

PURPOSE To select the soil or rock material types or soil/rock zones to which the property information following the **FOR** specification is applicable.

SYNTAX **FOR** {N1} [N2, N3]

N1 The first material type number to which the property specification applies. The maximum number of material type is set by default to be 500. This can be changed by the **ALLOcate** command. *The default value is 1.*

N2 The last material type number to which property specification applies. If N2 is not specified, it is assumed to be equal to N1. *The default value is 1.*

N3 The interval in the material type number designation. The specification will be effective for N1 to N2 at increments of N3 in the manner of a FORTRAN DO loop. *If N3 is not specified, the default value is 1.*

COMMENTS

The property information to which this command applies is specified through the **HYDR**aulic, **MULTi**-phase, **ROCK**, **SOIL**, **THER**mal and **TRAN**sport commands. This input remains effective for subsequent property commands until another **FOR** command is encountered.

EXAMPLES

FOR material type 3 properties are specified by the following commands

FOR soil type 1 through 5

FOR rock type 1 through 9 in steps of 3

COMMAND **GAS**

PURPOSE To specify composition and properties of the gas phase.

MODE 1: **Gas Molecular Weights for Gas Constituents**

SYNTAX **GAS** { [$\Phi_1=N1$, $\Phi_2=N2$, ..., $\Phi_3=Nn$] }

Φ_n One or more of the symbols for gas constituents for which the molecular weights are specified. The valid symbols are problem dependent and may be selected by the user from the list of default dependent variables or those defined by the user. The user may define up to 64 gas constituents.

$N1, \dots, Nn$ Molecular weights (> 0) for the variables in the order of the symbols. There are no default values; appropriate values must be specified. If a single numerical value is specified without any symbol then it is assumed that the value specifies the molecular weight of the default gas which is assumed to be the 1st defined transport variable.

COMMENTS

If this mode of the command is missing then by default the gas is defined to be single species gas with a molecular weight of 29. The one exception to this rule is the ANSWER™ Software Tool if a REACtion command is given with default chemical reactions. In this case 8 gas species are automatically defined; these are FU, CH, O2, CO, CO2, H2, H2O and N2.

EXAMPLES

GAS molecular weights: H2O=18, CO2=44, N2 = 28., O2=32

GAS molecular weight is 18

! Gas species by default

MODE 2: **Gas Reference Pressure**

SYNTAX **GAS** { PRES | P } { N1 }

PRES The reference gas pressure, p^* of Equation 3.1.4, is specified.

P Same as the modifier PRES above.

N1 The reference pressure for the gas. **The default value is 101325.**

EXAMPLES

GAS reference PRESSure is 2.5E5

GAS reference P is 2.5E5

MODE 3: Universal Gas Constant

SYNTAX **GAS** { CONS } [func [ξ]] { N1 } [fname | N2 .., Nn]

- CONS** The universal gas constant, R_u of Equation 3.1.4, is specified.
- func** One of the modifiers listed in Table 7.2.4 which denotes the functional form of the gas constant. *If no function is specified, then the gas constant is assumed to be fixed.*
- ξ** One of the independent variables listed in Table 7.2.5. *If no variable is specified, the independent variable is assumed to be time.*
- fname** The name of the file from which numerical values **N2** through **Nn** are read. This option can be used only if the selected function is a table or one of the series functions. See Section 7.2.2 for additional information.
- N1, ..., Nn** The numerical constants and coefficients for the selected function. *If no function is specified, then the default value is 8314. If a function is specified then there are no default values for this input* See Section 7.2.7 for more details..

EXAMPLES

Generic examples for this command are given in Section 7.2.8. The command keyword (**GAS**) must replace the keyword used in these examples. Some additional examples are given below.

- GAS** CONStant is 1545 !!!! ft lbf/lb-mole R
- GAS** CONStant is LINEar function: 30. -0.015 * P
- GAS** CONStant POLYnomial in P: (10., 0.5, 0., -0.1, 0.)
- GAS** CONStant is a TABLe in P: 3 sets: (0., 0.01), (100., 0.10), (200, -0.20)

COMMAND **GEOMETRY**

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

MODE 1: **Exchange Previously Specified Coordinates**

SYNTAX **GEOM { EXCH } { 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 } { dir₁ } { dir₂ } [N1, N2]**

ROTA The coordinates in the two directions specified by dir₁ and dir₂ are rotated in the plane defined by dir₁ and dir₂ by φ₁ and φ₂ degrees, respectively, according to the following equations:

$$X_{\text{new}} = X_{\text{old}} \cos \phi_1 + Y_{\text{old}} \cos \phi_2$$

$$Y_{\text{new}} = X_{\text{old}} \sin \phi_1 + Y_{\text{old}} \sin \phi_2$$

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 rotated in the plane defined by dir₁ and dir₂.

N1 The angle φ₁ (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 φ₂ (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:

$$X_{\text{new}} = X_{\text{old}} * N1 + N2$$

dir **One or more_** of the **X, Y, Z, R** or **THETA** modifiers that, respectively, denote the x, y, z, r or θ coordinates each of which is scaled according to the equation given above.

N1 The scaling factor for the transformation defined above. **There is no default value; a value must be specified.**

N2 The offset for the coordinate. **The default value is 0.**

EXAMPLES

GEOMetry SCALe X by 0.3048

GEOMetry SCALe X by 0.3048 add 1.00

GEOMetry SCALe X and R by 0.3048 add 1.00

GEOMetry SCALe X, Y and Z by 0.3048 add 1.00

MODE 4: **Generate a Cylindrical Geometry in r- θ Plane**

SYNTAX **GEOM { CYLI | CIRC } (N1) [N2, N3]**

CYLI Generates a cylindrical grid for the computational domain in the r- θ plane.

CIRC Same as CYLI modifier.

N1 The radius of the cylinder. **There is no default value; a value must be specified.**

N2 The angle of the cylinder circumference in degrees. **The default value is 180 degrees.**

N3 The starting angle from the origin for the cylinder in degrees. **The default value is 0 degrees.**

COMMENTS

This command is available only if the computational domain is located in the 1st or 2nd quadrant.

EXAMPLES

GEOMetry is **CYL**indrical with radius = 1.

GEOMetry is **CIRC**ular with r = 1, theta = 90

GEOMetry is **CYL**indrical with radius = 1., theta = 90

GEOMetry is **CYL**indrical 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:

$$r = \text{sqrt} (x_1^2 + x_2^2)$$

$$\theta = \tan^{-1} (x_2 / x_1)$$

dir₁, dir₂ Two of the **X, Y, Z, R** or **THETA** modifiers that, respectively, denote the x_1 and x_2 directions in the Cartesian framework to be transformed to or from the r and θ directions of the cylindrical coordinate system. **By default r direction is assumed to be aligned with the y direction and the θ with the z direction.**

COMMENTS

Transformation from Cartesian to radial framework is possible only for a 3-dimensional grid since in the 2-dimensional mode only (x,r) coordinate system is available. For the transformation to a cylindrical system the resulting r is stored at the same location as x_1 and θ at x_2 . For transformation to a Cartesian system the resulting x_1 is stored at the same location as r and x_2 at θ .

EXAMPLES

GEOMetry convert R and THETA to CARTesian

GEOMetry convert Y and Z to CYLIndrical

GEOMetry convert X and Y to CYLIndrical !!! only with a 3D grid system

COMMAND **GRAVITY**

PURPOSE To specify the gravitational acceleration vector and its components.

SYNTAX **GRAV** { N1, N2, N3 }, [N4]

N1, ..., N3 The components of the gravitational acceleration vector, g_j , in the three principal directions. Because only the ratio g_j/g appears in the equations, a relative value for this ratio may be specified. A positive value implies that the acceleration vector is directed in the principal direction; a negative value implies that it is directed in opposition to the principal direction. **The default values are set so that, for multiphase or free-surface flow, N2 is -9.81 for 2D flow and N3 is -9.81 for 3D flow; otherwise, the values are 0.**

N4 The absolute value of the gravitational constant, g , to be used for computing pressure head, P , from thermodynamic pressure, p , or vice versa. This input is necessary only if N1 through N3 are specified in a relative mode. **The default value is 9.81.**

COMMENTS

If flow is single-phase and the density is uniform, then the buoyancy term, B_j , is uniform and constant. The net buoyancy contribution is then zero because it is the gradient of the buoyancy term that appears in the pressure equation. In this instance, the gravitational components may be set equal to zero. It should be noted that these conditions lead to the pressure head, P , being identical to the total head, H .

EXAMPLES

GRAV ity constants are: 0., -9.81	! Orientation against y-axis
GRAV ity constants are: -6.937, -6.937, 0.	! 45 deg to x & y
GRAV ity constants are: 0., 0., -9.81	! Orientation against z-axis
GRAV ity constants are: 0., 0., 1, 32.2	! Relative value: dir +Z

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: 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

READ(NUNIT) (NI(N), NJ(N), NK(N), N=1,NBLOCKS)

DO N = 1, NBLOCKS

 READ(NUNIT) (((XC(I,J,K), I=1,NI(N)), J=1,NJ(N)), K=1,NK(N)

 READ(NUNIT) (((YC(I,J,K), I=1,NI(N)), J=1,NJ(N)), K=1,NK(N))

 READ(NUNIT) (((ZC(I,J,K), I=1,NI(N)), J=1,NJ(N)), K=1,NK(N))

 READ(NUNIT) (((IBLANK(I,J,K), I=1,NI(N)), J=1,NJ(N)), K=1,NK(N)) !May be omitted

ENDDO

NBLOCKS is the numbers of blocks of grid information for a multi-block grid,

NI, **NJ** and **NK** are, respectively, the number of vertices in the x, y and z directions,

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.

EXAMPLES

GRID is PLOT3D format on file 'PLOT3D.XYZ'

GRID in PLOT3D data in UNFormatted data on file 'PLOT3D.XYZ'

GRID in PLOT3D XYZ data in UNFormatted data on file 'PLOT3D.XYZ'

COMMAND HISTORY

PURPOSE To obtain output of the time history for dependent variables at selected nodes.

MODE 1: Specification of Location of Time History Nodes

SYNTAX **HIST** [**OFF**] {**N1**, ..., **Nn**} [**fname**]
 N=m in unstructured mode, = 2m in 2D and =3m in 3D mode; $m \leq 1024$

OFF The nodes specified by **N1** through **Nn** are switched to "off" condition. In this case, no output is generated for these nodes till a subsequent specification of the node is encountered.

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 OFF modifier is not present, then the nodes specified by the current command are added to any previously specified nodes. However, if the OFF modifier is present then the specified nodes are switched "off". **A cumulative maximum of 2048 nodes, over all HISTory commands, may be specified.**

fname The file name for time history output. **The default file name is 'acr_HISTORY.TMP' unless a name was specified on a previous HISTory command** At any time only one time history file may be open. If a new file name is specified, then the old file is closed and all subsequent output is directed to the new file.

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 2: Specification of Variables and Output Options for Mode 1**SYNTAX** **HIST** Φ **[fname]** **[TABL]** **[PLOT]** **[NOW | OFF | ON]** **[TIME]** **[V_{freq}]**

Φ One or more symbols that denote the variables for which the time history output is required. The valid symbols are listed in Table 6.8.1. By default, the output is obtained for all variables for which the equations are solved. The variables may be specified only once for each time history file. New set of variables may be specified only if the file name is changed by a subsequent command.

fname The file name for time history output. The default file name is 'acr_HISTORY.TMP' unless a name was specified on a previous **HISTORY** command. At any time only one time history file may be open. If a new file name is specified, then the old file is closed and all subsequent output is directed to the new file.

TABL The time history data are printed in a tabular form at the end of simulations.

PLOT A printer plot of time history of variables is generated at the end of simulations.

NOW One record of time history output is produced immediately.

OFF Time history output is discontinued.

ON Time history output is resumed if it was previously suppressed.

TIME By default, **V_{freq}** is interpreted to be the frequency of output in terms of number of steps. If this modifier is present, then **V_{freq}** is interpreted to be time interval between successive outputs.

V_{freq} 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.

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 3: History at an Arbitrary Point in Space**SYNTAX** **HIST** {COOR} { ϕ } [fname] [LINE | SQUA] [N1,..Nn] [V_{frq}] [TIME]**COOR** History is required at an arbitrary point in space specified by its (x, y, z) coordinates. **ϕ** **One or more** symbols for the variables for which the history is required. **No more than 10 symbols can be specified on one command.** The valid symbols are listed in Table 6.7.1. **There is no default value.****fname** The file name for output. **The default file name is 'acr_HISTORY_XYZ.TMP'.** A different output file may be specified for each command. **If no file name is specified then the output is directed to the most recent previously specified file. If no previous file was specified, then the output is directed to the default file.** The total number of open files in any simulation can not exceed 64. A summary of output is also printed to the standard output device at the end of simulations.**LINE** The values at the specified (x, y, z) location are computed by linear interpolation from its nearest neighbors. **This option is active by default.****SQUA** The values at the specified (x, y, z) location are computed by inverse distance squared interpolation from its nearest neighbors.**N1, ..., Nn** The grid coordinates (x, y, z) of the point. Two numerical values for 2D and three for 3D input modes must be provided. **There is no default value for this input; the required number of numerical values must be specified.****V_{frq}** The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 for further details. **If present, this must be last value on the command. By default the time history is generated at every step.****TIME** **By default, V_{frq} is interpreted to be the frequency in terms of number of steps.** If this modifier is present, then **V_{frq}** is interpreted to be the time interval between successive outputs.**COMMENTS**

This command generates output of history of values at an arbitrary location in space which may or may not coincide with the computational nodes. The history of values at computational nodes can be obtained by Mode 1 specification.

EXAMPLES**HISTORY** of U at point with COORDinate x=1.35, y=0.796 every 3 steps**HISTORY** of U at point with COORDinate x=1.35, y=0.796 every 20 steps with inverse SQUARE interpolation**HISTORY** of U, V, W, T at COORDinate x=1.35, y=0.796, z=0.0975 at TIME interval of 1.75 units**HISTORY** of U, T, C at COORDinates (1.35, 0.796, 0.0975) TIME interval 0.235 on 'HISTORY.XYZ'**HISTORY** U, T, C COOR (1.35, 0.796, 0.0975) TIME interval 0.235; SQUARE method 'HISTORY.XYZ'

MODE 4: History of Source for a Variable**SYNTAX** HIST {SOUR} { ϕ } [subrgn] [fname] [OFF] [V_{frq}] [TIME]**SOUR** Output for the source term for ϕ is required. **ϕ** The symbol for the variable for which the source inventory is required. Only those symbols may be specified for which a differential equation is solved. The valid symbols are listed in Table 6.7.1. There is no default value.**subrgn** The subregion for computations. If no subregion is specified, the entire domain is selected.**fname** The file name for output. The default file name is 'acr_SOURCE.TMP'. A different output file may be specified for each command. If no file name is specified then the output is directed to the most recent previously specified file. If no previous file was specified, then the output is directed to the default file. The total number of open files in any simulation can not exceed 64. A summary of output is also printed to the standard output device at the end of simulations.**OFF** Any previously specified command for the specified ϕ and subrgn is deactivated.**V_{frq}** The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 for further details. By default the time history is generated at every step.**TIME** By default, V_{frq} is interpreted to be the frequency in terms of number of steps. If this modifier is present, then V_{frq} is interpreted to be the time interval between successive outputs.**COMMENTS**

This command generates output of the history of "source" term for the variable. This includes all source (or sink) terms for the variable including the terms originally present in the governing equation (e.g. pressure gradient terms for momentum equations for ANSWER™) and those specified by the user through the various SOURCE, DECAY and REACTION commands. The output is the integral of the source for the variable ϕ over the volume of the subrgn.

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

EXAMPLES**HISTORY** SOUR or C for the entire domain**HISTORY** for SOURce of C for the entire domain**HISTORY** of SOURCE for T for SELEcted region every 20 steps**HISTORY** of SOURce for T for subregion defined by ID=VAULT every 200 steps**HISTORY** of SOURCE for T for ID=VAULT OFF**HISTORY** of SOURCE for Ton 'SOURCE.OUT' for SELEcted region at TIME interval of 0.4

MODE 5: History of Mass or Property Inventory for a Variable**SYNTAX** **HIST** {**STOR**} { Φ } [**subrgn**] [**fname**] [**OFF**] [**V_{frq}**] [**TIME**]**STOR** Output for the storage or accumulation term for Φ is required. Φ The symbol for the variable for which the inventory is required. Only those symbols may be specified for which a differential equation is solved. The valid symbols are listed in Table 6.7.1. There is no default value.**subrgn** The subregion for computations. If no subregion is specified, the entire domain is selected.**fname** The file name for output. The default file name is 'acr_STORAGE.TMP'. A different output file may be specified for each command. If no file name is specified then the output is directed to the most recent previously specified file. If no previous file was specified, then the output is directed to the default file. The total number of open files in any simulation can not exceed 64. A summary of output is also printed to the standard output device at the end of simulations.**OFF** Any previously specified command for the specified Φ and **subrgn** is deactivated.**V_{frq}** The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 for further details. By default the time history is generated at every step.**TIME** By default, **V_{frq}** is interpreted to be the frequency in terms of number of steps. If this modifier is present, then **V_{frq}** is interpreted to be the time interval between successive outputs.**COMMENTS**

This command generates output of the history of “accumulation” or “storage” term for the variable. The output is the integral of the property represented by the variable Φ over the volume of the **subrgn**.

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

EXAMPLES**HIST** **STOR** for C for the entire domain**HISTory** of **STORAge** for T for **SELEcted** region every 20 steps**HISTory** of **STORAge** for T for subregion defined by **ID=VAULT** every 200 steps**HISTory** of **STORAge** balance for T for **ID=VAULT** **OFF****HISTory** of **STORAge** for T on “**STORAGE.OUT**” for **ID=REGION1** at **TIME** interval of 0.4

COMMAND **HYDRAULIC**

PURPOSE To specify the reference hydraulic properties of the host porous matrix or of the planar or linear features.

SYNTAX **HYDR** {N1} [N2, N3, N4]

N1 The matrix compressibility, $\alpha_s \geq 0$), that is used to compute the effective storativity, S_e , according to Equation 3.4.1. The default value is 1.

N2, N3, N4 The reference values (≥ 0) of the three components of the hydraulic conductivity tensor, K_j of Equation 3.5.2 in the three principal directions, (x,y,z) or (x,r, θ), respectively. All off-diagonal components ($i \neq j$) are assumed to be zero. **The default value is 0.**

EXAMPLES

HYDRraulic properties: ss = 0.2, $K_x^* = 2$, $K_y^* = 0.2$, $K_z^* = 0.2$ ft per day

HYDRraulic ss = 0.2, $K_x = 2$; $K_y = 0.2$

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. If no file name is provided on the command, then the user is prompted for a file name at run time. 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.

EXAMPLES

INCLude file 'INCLUDE.NOW'

INCLude commands from file INCLUDE.NOW

INCLude ! user will be prompted

INCLude ? ! user will be prompted

COMMAND **INITIAL**

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

SYNTAX **INIT** { $\Phi = \Phi_0$ } [N1 ..., Nn]

Φ The symbol that denotes the field variable for which input is provided. The valid symbols are listed in Table 6.8.1. **One, and only one, symbol must be specified.** There is no default value.

Φ_0 The initial value for the variable.

N1, ..., Nn These values define the subregion to which the input is applied. In the default mode, the subregion is defined by four values for the 2D and 6 for the 3D input mode. **If only one value, that is N1, is specified, then it is assumed that N1 is the material or zone number to which the input applies.** **If this input is completely omitted, then the subregion is defined to be the entire domain.**

COMMENT

This command is available only for structured grids. **SET** command is a much more powerful option for setting field values of the variables for structured or unstructured grids.

EXAMPLES

INITial U is 0.1 everywhere

INITial T is 1.E-3 from (2,2) to (7,9) ! 2D input mode

INITial T is 1.E-3 from (2,2,2) to (7,9,4) ! 3D input mode

INITial T is 1.E-3 for zone 5

COMMAND **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 and Table 7.2.3 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** [$\phi=N1, \phi=N2, \dots, \phi=Nn$] { **HYBR** | **COND** | **QUIC** | **CENT** }

ϕ One or more of the symbols that denote the heat or mass transport variables for which the profile specification is effective. The valid symbols are listed in Table 6.7.1. If no symbol is specified, the input is assumed to be effective for all variables.

HYBR The hybrid scheme (Runchal, 1972) is used for integration. This is the default option.

COND The CONDIF, a total variation diminishing scheme (Runchal, 1987), is used for integration. This option is currently available in a limited manner. Please check with ACRI for its use.

QUIC An generalized flux-conservative form of the QUICK scheme (Leonard, 1979), is used for integration.

CENT The second order central-difference scheme is used for integration. The scheme may generate over- and under-shoots if the local grid Peclet number exceeds a value of 2. It is known to be unstable in the linear sense if the local grid Peclet number exceeds 2.

N1, ..., Nn The diffusion control parameter (quantity R_{max} of Runchal, 1987) for the variable denoted by the symbol immediately preceding the value if the CONDIF scheme is selected; otherwise this input is ignored. The recommended values are between 2 and 10. Numerical diffusion decreases with increasing value of this parameter. The default value is 10.

COMMENTS

The default option should be adequate for most applications. However, if the local grid Peclet number (Equation 4.2.1) is significantly larger than 10 and the direction of flow is not (approximately) aligned with any of the coordinate directions, the CONDIF or QUICK scheme may be desirable. See Sections 4.2 and 4.4 for additional discussion.

EXAMPLES

INTEgration for U by **COND**if scheme

INTEgration for T by **HYBR**id profile (same as default)

INTEgration for T by modified **QUIC**k scheme

INTEgration for T=10., C=8., C2=5. by **COND**if scheme

INTEgration for T=10., C=8., C2=5. by **CENT**ral Difference scheme

COMMAND **LIMIT****PURPOSE** To specify the limiting values for field variables**SYNTAX** **LIMI** { Φ } { Φ_{MIN} , Φ_{MAX} | OFF} Φ A symbol that denotes the variable for which the limits are being set.

OFF All previously set limits for the selected variable are deactivated.

 Φ_{MIN} The minimum permissible value for the variable. Any computed value that is less than Φ_{MIN} is set to Φ_{MIN} . Φ_{MAX} The maximum permissible value for the variable. Any computed value that is greater than Φ_{MAX} is set to Φ_{MAX} .**COMMENTS**

The default values for all field variables are set to -10^{30} as the minimum and 10^{30} as the maximum. However, these values may be internally revised for some of the variables (such as mass concentration) based on the nature of the problem and that of the variable. This command may be used to override the internally set values, if so desired.

EXAMPLES**LIMIT** for T minimum = 300**LIMIT** for T minimum = 0 maximum = 100**LIMIT** for T OFF !Disable any previously specified or internally set limits

COMMAND	LOCATE
PURPOSE	To locate and identify a subregion in the domain of computation for later reference by other input commands.
MODE 1:	Subregion Specification by Grid Indices for a Structured Grid
SYNTAX	LOCA [ID=idsub] [FIEL] {N1, ..., Nn} , [Nn+1, Nn+2, Nn+3] [fname]
idsub	A unique identifier for the subregion. 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 LOCAt e or SELEct on the command can make a generic reference to the specified subregion until the next LOCAt e or SELEct command. See Sections 7.2.3 and 7.2.4 for additional details.
FIEL	If the LOCAt e 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 FIELd modifier is present, then the subregion comprises only the interior filed nodes (or elements) and the exterior boundary nodes are not included in the subregion. Presence of this modifier makes no difference to those commands that operate only on the interior field elements such as the FIX and SOURc e command. Other commands, such as the SET and COND uctivity, which can be used to set values of variables at both field and boundary nodes will be affected by this modifier.
N1, ..., Nn	The grid indices (I,J,K) for the subregion. A total of 4 numerical values for 2D and 6 for 3D input must be provided. If the values specified lie outside this range then the input values are clipped to lie in the valid range specified by the GRID command. There is no default value; a valid set of values must be specified. See Section 7.2.3 for additional information.
Nn+1	The x-direction node interval for subregion. Only those nodes occurring at this interval, starting with N1 specification are considered to be part of the subregion. For example, a specification of 3 results in the nodes located at I=N1+1, N1+4, N1+7, etc. to be included in the selected subregion; the intermediate nodes are excluded. The default value is 1.
Nn+2,Nn+3	The y and z direction node intervals in the manner described above. The default value is 1.
fname	If a valid file name is present (see Section 7.2.2) then an ordered list of element numbers selected by the command is written to the file. For structured grids a list of corresponding (I,J,K) grid indices is also written to the file.

EXAMPLES

LOCAte subregion from (6,10) to (31,10)
LOCAte subregion (6,10) to (31,10) with ID=DMN1
LOCAte region ID=DMN2 as: (6,10,1) to (31,10,15) with interval (2,3,1)
LOCAte region (6,10) to (31,10) with interval (2,3)
LOCAte subregion (1,1) to (22,22) only **FIELd** nodes as ID=INNR
LOCAte subregion (1,1,3) to (22,22,7) only **FIELd** nodes as ID=INNR output on file 'INNR.IJK'

MODE 2: Subregion Specification by Grid Coordinates of Rectangular Windows

SYNTAX **LOCA** {**COORD**} [**ID=idsub**] [**INTE**] [**NOT**] [**EXCL**] [**FIEL | BOUN**] {**N1, ..., Nn**} [**fname**]

COORD **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

- LOCA**te region with **COORD**inates (0., -20.) to (11571.5, 80.)
- LOCA**te subregion **ID = DMN2** for grid **COORD**inates (0., 0.) to (100, 1500)
- LOCA**te subregion **ID = DMN2** for **COORD**inates (0., 0.) to (100, 1500) **EXCL**ude selection
- LOCA**te **ID = DMN2** for grid **COORD**inates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2)
- LOCA**te grid **COORD**inates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2) and **EXCL**ude selected
- LOCA**te **COORD**inates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2) select **INTER**section & **EXCL**ude
- LOCA**te **COORD**inates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2) select **INTER**section & **EXCL**ude
- LOCA**te **COORD**inates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2) select **NOT INTER**section & **EXCL**ude
- LOCA**te **COORD**inates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2) select output on “**DMN2.LOC**”
- LOCA**te **COORD**inates (0., 0.) to (2.5,1.5) and (3.25,1.0) to (5.0,5.2) select only **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 $N_{23D} * N_{vertex}$ 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 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

LOCAte SPHEre radius=0.4, center coordinates (2., 0.5) !2D
LOCAte SPHEre radius=0.4, center coordinates (2., 0.5, 0.5)
LOCAte SPHEre radius =0.4, center coordinates (2., 0.5, 0.5) FIEL node only
LOCAte SPHEre radius =0.4, center coordinates (2., 0.5, 0.5) output to file 'SPHERE.LOC'

MODE 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 **LOCAt**e command is currently implemented in a limited manner; please consult ACRI before its use.

EXAMPLES

LOCAte MATERial type 3 as the active subregion**LOCAt**e ZONE number 5 as subregion with ID=TYP5**LOCAt**e 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 **FIELd** 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

LOCA te COORdinate LIST: (0.,0.), (2,2), (1.53,1.37), (23.1,27.2)	! List for 4 points in 2D
LOCA te COORdinate LIST: (0.,0,0), (2,2,2), (1.53,1.37,1), (23.1,27.2,2)	! List for 4 points in 3D
LOCA te COORdinate LIST of 500 elements as ID=BIG from file 'XYZ.LST'	! Read from file
LOCA te COOR LIST 500 as ID=BIG from file 'XYZ.LST' only FIELD	! Read file; only field
LOCA te COOR LIST: (0.,0.), (2,2), (1.53,1.37), (23.1,27.2) output to 'FILE.rgn'	! List with output file

MODE 9: Subregion Specification by a List of Elements

SYNTAX **LOCA** {LIST} [ID=idsub] [FIEL] [IJK | SEQU] [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.

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. Only the internal elements can be specified; the boundary nodes are automatically included.

SEQU The numerical input specifies a sequence of element numbers.

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 In the absence of the SEQUence and the IJK modifiers, these numbers identify the elements that define the subregion. In the presence of the IJK modifier these numbers 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. If the IJK modifier is absent but the SEQUence modifier is present, then only N1, N2 and N3 are meaningful. In this case, these three numbers specify the starting, the ending and the interval index for the element numbers in the sense of a FORTRAN DO loop.

EXAMPLES

- LOCA**te element numbers LIST: 1, 2, 3, 7, 17, 29 ! Structured Grid
- LOCA**te element numbers: 1, 2, 3, 7, 17, 29 ! Default for Unstructured Grid
- LOCA**te LIST of 500 elements from file 'ZONE.BIG' ! Read from file
- LOCA**te 5000 elements from file 'ZONE.BIG' ! Read from file for Unstructured Grid
- LOCA**te LIST by IJK indices: (2,5), (3,3), (5,7), (9,2), (3,4) for a two-dimensional structured grid
- LOCA**te LIST of IJK indices: (2,5,2), (3,3,5), (5,7,2), (9,2,3), (3,4,5) for three-dimensional structured grid
- LOCA**te element SEQUential LIST 1 through 51 interval=10 ! Same as above
- LOCA**te LIST by SEQUence 31 to 51 ! All elements from 31 to 51,inclusive
- LOCA**te LIST 79, 22, 33, 34, 89, 2, 7 with ID=LST1 and output on file "LST1.LOC" ! Write to file

MODE 10: **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 11: Subregion of Matched Elements**SYNTAX** **LOCA** {**CORR**} {ID=idsub1} {ID=idsub2} [ID=idsub] [fname]

CORR A subregion of correlated elements is defined for special purposes. The subregion consists of two sets of elements which are paired with each other. The pairing takes place in a parallel sequential mode where the 1st element in the 1st subregion is paired with the 1st element of the 2nd subregion, and so on. This type of subregion can only be used with certain commands (e.g. **CORR**elation) where the values at the elements in the 1st set are paired with, or dependent on, the values of the elements in the 2nd set.

idsub1 The name of the 1st subregion to be operated upon. *There is no default value; a valid name of a previously specified subdomain with the ID modifier must be specified.*

idsub2 The name of the 2nd subregion to be operated upon. *There is no default value; a valid name of a previously specified subdomain with the ID modifier must be specified.*

idsub A unique identifier for the subregion. **Only the first 8 characters are meaningful; any subsequent characters are ignored.** This identifier may be subsequently used in an input command to provide selective input for that subregion. If the identifier is omitted, then the modifier **LOCA**te or **SELE**ct on the command can make a generic reference to the specified subregion until the next **LOCA**te or **SELE**ct command. See Sections 7.2.3 and 7.2.4 for additional details.

fname If a file name is present then a file is generated that lists the elements and, for structured grids, a list of (I,J,K) grid indices. *A valid name (see Section 7.2.2) must be specified.*

EXAMPLES

LOCAte CORRelated Elements from ID=REGION1 and ID=REGION2

LOCAte CORRelated Elements from ID=REGION1 and ID=REGION2 as ID=MATCHED on file='match.fil'

MODE 12: 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****LOCA**te (element, surface) PAIRs: (35,1), (53,3), (77,2), (13,4)**LOCA**te PAIR : (35,1), (53,3), (77,2), (13,4) as boundary ID=BNDRy**LOCA**te PAIR : 500 sets as ID=BNDRy from file 'BOUNDARY.BIG'**LOCA**te PAIR : (35,1), (53,3), (77,2), (13,4) as boundary ID=BNDRy**LOCA**te PAIR : (35,1), (53,3), (77,2), (13,4) as boundary ID=BNDRy**LOCA**te PAIR : (35,1), (53,3), (77,2), (13,4), (28,5), (33,3), (35,6)**LOCA**te PAIR (35,1), (53,3), (77,2), (13,4), (28,5), (33,3), (35,6) output on file "Boundary.LOC"

MODE 13: Boundary Specification by List of Vertices

SYNTAX **LOCA** { VERT } [ID=idsub] [fname] [EXCL] [SEQU] [option] {N1 | N1, ..., Nn }

VERT The boundary region defined by the specified sets of vertices is selected.

idsub See Mode 1 specification.

fname If a file name is present then the file may be used for input or output. The input mode is activated if a file name is specified with only one numerical value (N1). The file may contain a header field with up to 100 lines (records) of information. The header field, if present, must end with the record "END HEADER" and the numerical values must immediately follow this record. In the output mode, a file is generated that lists the element and surface number pairs and, for structured grids, a list of (I,J,K) grid indices. In the file input or output mode, a valid name (see Section 7.2.2) must be specified.

EXCL If the modifier is present, then the boundary specified by the vertices is excluded and the complimentary set of the domain boundary is selected.

SEQU The numerical input specifies a sequence of element numbers.

option

option	INTERPRETATION
ANY	By default a boundary element (or node) is included in the selection only if all its vertices are included in the list of vertices. If ANY modifier is specified then a boundary element is included if any of its vertices is included in the list.
ONE	A boundary element is included if at least one of its vertices is included.
TWO	A boundary element is included if at least two of its vertices are included.
THRE	A boundary element is included if at least two of its vertices are included.
FOUR	A boundary element is included if at least two of its vertices are included

N1 If the file input mode is activated, then, N1 denotes the number of vertices to be read from the file. If the file input mode is not activated, then N1, ..., Nn are described below.

N1,..,Nn In the absence of the 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

LOCATE: VERTices 1,3,7,8,10,20,55,99,203,105,77
LOCATE VERTices: in SEQUENCE from 23 through 231 in steps of 3 as boundary ID=BNDRy
LOCATE VERTices: 500 values as ID=BNDRy from file 'BOUNDARY.BIG'
LOCATE VERTices: in SEQUENCE 23 to 231 in steps of 3 ID=BNDRy output file: 'Boundary.LOC'

MODE 14: **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 15: Boundary Specification as a Complimentary Surface of a Previous Subregion

SYNTAX { } { } { } [] []

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 or type of element sub domains or on previous commands defined with a modifier**

The identification or name of the previous subregion to be operated upon. *There is no default value; a valid name must be specified.*

The orientation index of the subregion with the complimentary surface. See Table 7.2.3 in Section 7.2.5 for available choices. *This input is not required if was defined by a command.*

See Mode 1 Specification.

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

- LOCA**te UNIO**n** of ID=DMN1 and ID=DMN2
- LOCA**te UNIO**n** of ID=DMN1 and ID=DMN2 as ID=DMN3
- LOCA**te INTER**se**ction ID=DMN1 and ID=DMN2
- LOCA**te UNIO**n** of subregions ID = DMN1 and ID=DMN2 EXCL**u**de selection only FIEL**d** elements
- LOCA**te UNIO**n** of ID=DMN1 and ID=DMN2; EXCL**u**de selected and name ID=DMNM12
- LOCA**te ID=DMN1 plus ID=DMN2 select INTER**se**ction & EXCL**u**de
- LOCA**te ID=DMN1 plus ID=DMN2 select NOT INTER**se**ction & EXCL**u**de output on file 'DMN12.LOC'

MODE 16: Specification of a General Second Order Surface

SYNTAX **LOCA** { SURF } { BOUN } { N1, N2, , Nn } [TOLE] { Nn+1 } [ID=idsub] [fname]

SURF All boundaries of the elements in the vicinity of a general second order surface are selected. The 2D and 3D surfaces are specified, respectively, as:

$$C_0 = C_1 x + C_2 y + C_3 x^2 + C_4 y^2 + C_5 x y$$

$$C_0 = C_1 x + C_2 y + C_3 z + C_4 x^2 + C_5 y^2 + C_6 z^2 + C_7 x y + C_8 y z + C_9 z x$$

This type of specification results in the location of a boundary Pair type of subregion that consists of the element nearest to the surface and the element boundary in the vicinity of the surface.

BOUN If this modifier is present then only the external boundary nodes are searched to satisfy the surface constraint. Otherwise all the element boundaries are searched.

N1,..,Nn These specify the coefficients C_0 through C_5 (for 2D) or C_0 through C_9 (for 3D). Any trailing coefficients not specified are assumed to be zero. At a minimum C_0 through C_2 (for 2D) or C_0 through C_3 (for 3D) must be specified.

TOLE If this modifier is present then the last value on the command is assumed to be the tolerance for judging the proximity to the specified surface.

Nn+1 The tolerance for judging the proximity to the surface. This numerical value must be present if the modifier **TOLErance** is specified. The default value is 10^{-7} .

idsub See Mode 1 specification.

fname If a file name is present then a file is generated that lists the element boundary pairs, and, for structured grids, a list of (I,J,K) grid indices. A valid name (see Section 7.2.2) must be specified.

EXAMPLES

LOCAte SURFACE 2 = 1. *x + 1. * y ! 2D mode
LOCAte external BOUNDary SURFACE 2 = 1. *x + 1. * y + 0. * z ! 3D mode
LOCAte SURFACE 1 = -1. x -1. y -1. z + 1. xx + 1. yy +1. zz +0. xy -1. yz
LOCAte SURFACE 1 = -1. x -1. y -1. z + 1. xx + 1. yy +1. zz +0. xy -1.yz ID=SURF2 file='SURF2.LOC'
LOCAte BOUNDary SURFACE 1 = -1. -1. -1. + 1. + 1. +1. TOLerance=1.E-9 ID=SURF2

MODE 17: Matched Surface Subregion from Specified List

SYNTAX **LOCA** {**MATC**} {**LIST**} [**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 **TRANser** command that transports fluid flux along with specified fluid properties from one surface to another with or without transformation.

LIST An explicit list of matched pairs of elements and surface number 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 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 18: Matched Surface Subregion from Two Previously Defined Subregions.

SYNTAX **LOCA** {**MATC**} {**ID=**idsub1} [**dir1**] {**ID=**idsub2} [**dir2**] [**INTE**] [**ID=**idsub] [**fname**]

MATC A matched surface type of subregion is defined which is a special type of subregion consisting of two matched surfaces. It can only be used with the **TRANser** 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 Table 7.2.3 in 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 Table 7.2.3 in 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

LOCate **MATCh** X- direction of **ID=DMN1** and X+ direction of **ID=DMN1** as **ID=MATCHED**

LOCate **MATCh** X- direction of **ID=DMN1** and X- of **ID=DMN2** **INTERnal** surfaces as **ID=MATCH2**

LOCate **MATCh** X- direction of **ID=DMN1** and X- of **ID=DMN2** as **ID=MATCH2** also file='MATCH.FIL'

MODE 19: Subregion by Union or Intersection of Two Previous Subregions

SYNTAX **LOCA** {UNIO | INTE} {ID=idsub1} {ID=idsub2} [NOT] [EXCL] [ID=idsub] [FIEL] [fname]

- UNIO** The new subregion is created from the union (elements that belong to either) of two previously specified subregions. There is no default value. Either this or the **INTER**section 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 **INTER**section 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.

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

- LOCA**te UNIO n of ID=DMN1 and ID=DMN2
- LOCA**te UNIO n of ID=DMN1 and ID=DMN2 as ID=DMN3
- LOCA**te INTER section ID=DMN1 and ID=DMN2
- LOCA**te UNIO n of subregions ID = DMN1 and ID=DMN2 EXCL ude selection only FIEL d elements
- LOCA**te UNIO n of ID=DMN1 and ID=DMN2; EXCL ude selected and name ID=DMNM12
- LOCA**te ID=DMN1 plus ID=DMN2 select INTER section & EXCL ude
- LOCA**te ID=DMN1 plus ID=DMN2 select NOT INTER section & EXCL ude output on file 'DMN12.LOC'

MODE 20: **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

COMMAND MATERIAL**PURPOSE** Define material types and porous media properties.**MODE 1:** Material Type for a Subregion**SYNTAX** **MATE** [TYPE] { N_{Mat} } [subrgn] [FIEL] [dir]**TYPE** Identifies a material type with unique properties. If this modifier is omitted and more than 4 numerical values are specified, then the input is taken to be for material type.**N_{Mat}** A number that designates the material type or zone. A distinct number should designate each different material that has its own matrix or characteristic properties. The default value is 1. By default the maximum assigned number is limited to 100. If numbers exceeding 100 require to be defined, then the **ALLOCATE MATERIAL** command must be used to specify the maximum number of material types. Though the material numbers may be assigned in any arbitrary sequence, most efficient use of memory results if the numbers are defined sequentially.**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**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.**COMMENTS**

This command divides the domain of computation into regions with distinct properties. This zone number is used by a number of material 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.

EXAMPLES**MATE**rial type 1 ! total domain**MATE**rial type 3 as subregion ID=COARSE_SAND**MATE**rial type 5 as the currently SELEcTted subregion**MATE**rial type 5 for only the Y+ boundary of ID=GRAVEL**MATE**rial type 6 for only the FIELD nodes of ID=GRAVEL

MODE 2: **Material Type for a Rectangular Window**

SYNTAX **MATE** [TYPE] { N_{Mat} } [N1, ..., Nn] [COOR]

TYPE Identifies a material type with unique properties. If this modifier is omitted and more than 4 numerical values are specified, then the input is taken to be for material type.

N_{Mat} 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. 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.

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).

EXAMPLES

- MATE**rial TYPE 1 ! total domain
- MATE**rial TYPE 3 from (1,1) to (11,7) ! 2D input mode
- MATE**rial TYPE 5 from (1,1,1) to (11,7,5) ! 3D input mode
- MATE**rial TYPE 5 with **COOR**dinate 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.

EXAMPLES

MATErial type information from 'TYPE.DAT'

MODE 4: Specification of a Fracture or Borehole

SYNTAX **MATE** {FRAC | BORE} [TYPE] { N_{Mat} } {N1, ..., Nn} [COOR] {Nn+1} [Nn+2]

FRAC The material type designation is for a fracture. This feature is only available with the structured grid mode of **PORFLOW™**.

BORE The material type designation is for a borehole. This feature is only available with the structured grid mode of **PORFLOW™**.

TYPE See Mode 1 specification.

N_{Mat} See Mode 1 specification.

N1, ..., Nn The grid indices (I,J,K) or coordinates (x, y, z) of the material type or zone. See Section 7.2.3. If this input is omitted, then the subregion is defined to be the total domain. A total of 4 values **must** be specified for 2D and 6 for 3D geometry.

COOR See Mode 1 specification.

Nn+1 Width of the planar feature if the modifier **FRAC** is present; outer diameter of the linear feature if the modifier **BORE** is present. There is no default value; a value must be specified.

Nn+2 Inner diameter of the linear feature if the modifier **BORE** is present. By default it is assumed that the inner diameter is zero.

COMMENTS

Each fracture (planar feature) or borehole (linear feature) must be specified by a different MATERIAL command. A fracture that is not oriented along one of the principal axes may be specified by its components in a stair-step manner. In the 3D mode, a fracture may be in the xy (N4=N7), yz (N2=N5), or zx plane (N3=N6). Similarly, a borehole may be in the x (N3=N6, N4=N7), y (N2=N5, N4=N7), or z (N2=N5, N3=N6) direction. In the 2D mode, only linear features may be present; they may be in the x (N3=N5) or y (N2=N4) direction. The cross-sectional area of the linear element, with D and d, respectively, as the outer and inner diameters, is calculated as:

$$\text{Area} = (\pi/4) (D^2 - d^2) ,$$

EXAMPLES

- MATERIAL** 3 from (6,10) to (31,10) BORE hole in X-dir dia = 0.2
- MATERIAL** type 4 from (6,10) to (6,19) BORE in Y: D=0.2, Inner d=0.1
- MATERIAL** type 3 from (6,10,2) to (6,10,7) BOREhole in Z dia = 0.1
- MATERIAL** type 3 from (6,10,2) to (31,10,7) FRACture in XZ width = 0.1
- MATERIAL** 5 FRACture in YZ: COOR (0.5, 0.7, 1.35) (0.9, 0.7, 1.45) W=0.05

MODE 5: **Specification of Material Density**

SYNTAX **MATE {DENS} {N1}**

DENS The material density is specified.

N1 The density of the dry solid component, ρ_s (> 0), of the porous material. The default value is 1.

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, Θ_E . The default value is 1.**N2** The total porosity, Θ_T . If no value is specified, total porosity is set equal to the effective porosity.**N3** The diffusional porosity, Θ_D . If no value is specified, diffusional porosity is set equal to the effective porosity.**EXAMPLES**

MATErial porosity: 0.35**MATE**rial porosities: effective = 0.1, total 0.2**MATE**rial porosities: effective = 0.1, total 0.2, diffusive 0.15**MATE**rial porosities: 3*0.15**MATE**rial porosities: 0.2, 0.25, 0.21**MATE**rial porosities are 0.45, 1.50, 1.25 **OVER**ride internal check

MODE 7: Specification of Material Density and Porosity**SYNTAX MATE [PROP] {N1} [N2, N3, N4]**

PROP The material properties are specified. In this mode both density and porosity are specified on one command. If this specification is omitted and the modifier TYPE is not present, and the number of numerical values on the command is less than 5, then the specification is assumed to be for material properties.

N1 The density of a dry, solid component, ρ_s (> 0). The default value is 1.

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

N3 The total porosity, Θ_T . If no value is specified, total porosity is set equal to the effective porosity.

N4 The diffusional porosity, Θ_D . If no value is specified, diffusional porosity is set equal to the effective porosity.

COMMENTS

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 (τ_{ij} of Equation 2.2.8 and 2.3.9) of the porous matrix. Only components in the principal directions are used. The numerical values must be between 0 and 1. **The default value is 1.**

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 [$\Phi=N1, \Phi =N2, \dots, \Phi =Nn$] [dir] [option] [auxiliary]
Φ	One or more symbols that denote the variable(s) for which the N1, N2, ..., etc. and the specified option are effective. Valid symbols are listed in Table 6.7.1.
dir	One or more of the characters: X, Y, Z to denote the direction in which the matrix sweeps are made. For example, a specification of X results in the matrix equations being solved for the x-direction nodes in increasing order of the I grid index, for fixed values of the J and K indices. This input is effective only for the ADI method. By default, the matrix is swept in all active coordinate directions.
option	<p>ADI Matrix is solved by an alternating direction implicit method. This is the default option.</p> <p>SOR Matrix is solved by implicit successive over relaxation where values from the current iteration are used where available.</p> <p>EXPL Matrix is solved by explicit successive over relaxation where only the values from the old iteration are used.</p> <p>USER Matrix is solved by an algorithm supplied by the user. See Section 7.2.9.</p>
auxiliary	<p>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.</p> <p>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.</p> <p>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.</p>
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 EXPLicit SOR

MATRix for T USER option 5 ! Matrix Solver supplied by User

MODE 2: **Number of Inner Matrix Iterations**

SYNTAX **MATR {ITER} { N1 | ϕ =N1, ϕ =N2, ..., ϕ =Nn }**

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

ϕ Symbols that denote the variable(s) for which the N1, N2, ..., etc. are effective. Valid symbols are listed in Table 6.7.1. **If no symbol is specified, and only one numerical value is present, then the input is applied to all variables for which a transport equation is solved.**

N1,..., Nn Number of times (≥ 1) per time step that the matrix is solved for the variable denoted by the symbol immediately preceding the value. **The default value is 1.**

COMMENTS

Once the matrix coefficients are assembled, the matrix is solved iteratively till the specified convergence criterion (**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 Diagonal Coefficient**

SYNTAX **MATR {COEF}{V_min }**

COEF The minimum value for the matrix diagonal coefficient is specified.

V_min The minimum threshold for any of the diagonal coefficients of the matrix of equations. The default value is 1.E-20.

COMMENTS

The magnitude of the diagonal coefficient for any of the algebraic equations must be greater than zero otherwise the matrix is indeterminate. However due to limitations of digital arithmetic, and round-off and truncation errors, the value may become very small. This input allows the user to tune the minimum threshold to the machine accuracy.

EXAMPLES

MATRix COEFFicient minimum value = 1.E-30

MODE 4: 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 of the University of Texas is used. The solvers are described in: "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. The source code and manuals are available at <http://www.netlib.org/itpack>. **A special agreement is required for use of this solver package.**

ELIM Integer parameter IPARM(1) is set to 1. This implies that equations below a threshold level are dropped from the matrix system.

ELIM OFF Integer parameter IPARM(1) is set to 0. This turns off any previously specified ELIM modifier. **This is also the default setting.**

Φ One or more symbols that denote the variable(s) for which the specified options are effective. Valid symbols are listed in Table 6.7.1.

precon Preconditioner component of the Matrix Solver.

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

accel Accelerator Component of the Matrix Solver

accel	INTERPRETATION
CONJ	Conjugate Gradient accelerator.
BCGS	The Biconjugate Gradient Squared accelerator.
GMRE	GMRES accelerator. This is the default option.
ORTH	ORTHOMIN accelerator.
CGNR	Conjugate Gradient applied to Normal Equations.
LANC	Lanczos with ORTHOMIN accelerator.

option Modifier for the specification or matrix or solver method

option	INTERPRETATION
MODI	If CHOLesky preconditioner is selected, then modified incomplete Cholesky decomposition is used; otherwise this input is ignored.
PERM	If CHOLesky preconditioner is selected, then matrix is red-black permuted; otherwise this input is ignored.

EXAMPLES

MATRix for P by NSPCg (permuted CHOLesky).

MATRix for P from NSPCG with JACObi preconditioner and BCGS accelerator

MATRix for P is SYMMetric; use NSPCG with JACObi and BCGS

MATRix U V W P use NSPCG preconditioner: NEUMann poly. of DEGRee 3 GMRES accelerator

MATRix P NSPCG NEUMann DEGRee 3 GMRES

MATRix NSPCg P NEUMann DEGRee 2 TRUNcated GMRES ELIMinate equations below threshold.

MATRix NSPCg turn previously specified ELIMination OFF

MODE 5: Matrix Solvers from the HYPRE Library, Lawrence Livermore National Laboratory

SYNTAX **MATR** {HYPR} [Φ] [precon] [accel] [option] [VECT= N_{vector}] [LEVE= N_{level}] [ELEM=N_{elem}]

HYPRE One of the parallelized matrix solvers from the HYPRE package of the Lawrence Livermore National Laboratory (University of California), Livermore, California is used. A recent summary of the HYPRE library is described in: "Falgout, R.D. and Yang, U. M., "HYPRE: A Library of High Performance Preconditioners," in *Computational Science - ICCS 2002 Part III*, P.M.A. Sloot, C.J.K. Tan, J.J. Dongarra, and A.G. Hoekstra, Eds., Lecture Notes in Computer Science, vol. 2331, pages 632-641, 2002, Springer-Verlag. Also available as Lawrence Livermore National Laboratory technical report UCRL-JC-146175. The source code and manuals are available at <http://www.llnl.gov/CASC/hypre/>. **A special agreement is required for use of this solver package.**

Φ One or more symbols that denote the variable(s) for which the specified options are effective. Valid symbols are listed in Table 6.7.1.

precon Preconditioner component of the Matrix Solver.

precon	INTERPRETATION
AMG	Algebraic Multi-Grid Preconditioner. This is the default option.
CHOL	Incomplete Cholesky Factorization..
SPAI	The SParse Approximate Inverse preconditioner.

accel Accelerator Component of the Matrix Solver

accel	INTERPRETATION
CONJ	Conjugate Gradient accelerator.
BCGS	The Biconjugate Gradient Stabilized.accelerator
GMRE	GMRES accelerator. This is the default option.

option Modifier for the specification or matrix or solver method

option	INTERPRETATION
SYMM	By default the matrix is assumed to be non-symmetric. If this modifier is present, then the matrix is assumed to be symmetric.
INDE	By default the matrix is assumed to be positive definite. If this modifier is present, then the matrix is assumed to be indefinite This may be the case if Central Difference Integration method is used (see INTEgration Command)

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

N_{vector} The number of previous vectors that are stored to derive the new iterate of the solution. The typical values are between 5 and 20. **The default is set at 5.**

LEVE For the **SPAI** preconditioner this modifier controls the level of the pattern matrix. The pattern matrix is computed from the matrix **A** by dropping small amplitude elements. For the **AMG** preconditioner this modifier controls the maximum number of coarse levels. This modifier is ignored for other preconditioners.

N_{Level} The numerical value for the **LEVEL**. For the **SPAI** 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.

ELEM This modifier controls the specification of the maximum number of non-zero elements that are to be retained in any row of the modified matrix during factorization.

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.**

EXAMPLES

MATRix by HYPRe solver

MATRix for P from HYPRe solver print detailed DIAGnostics

MATRix for P from HYPRE with AMG preconditioner and CG accelerator

MATRix for P is SYMMetric; use HYPRe with CHOLesky with CG

MATRix U V W use HYPRE preconditioner: SPAI of LEVEI=3 GMRES accelerator

MATRix U V W use HYPRE CHOLesky ELEMents=20, with GMRES.

MODE 6: Matrix Control Parameters for NSPCG and HYPRE Packages

SYNTAX **MATR** [ZERO = V_{Zero}] [TOLE = V_{Tole}] [DIAG] [SUMM] [OFF]

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.

1.192×10^{-7}	IEEE real with 32 bit precision.
2.22×10^{-16}	IEEE real with 64 bit precision. This is the default.
7.1×10^{-15}	Cray XMP.
1.49×10^{-8}	Dec 10 (single precision)
4.768×10^{-7}	IBM 370 / 158 (single precision)

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 NSPCG solvers this is also the threshold value at which the ELIMination option of Mode 2 becomes active, i.e., the value below which equations are eliminated from the system. For the NSPCG the default value is set at 500 times the value of machine zero. For the SPAI preconditioner the typical values are between 0.01 and 0.1 and the default is set at 0.05. For the CHOL 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.

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.

EXAMPLES

MATRix TOLerance 1.0E-10

MATRix machine ZERO set to 1.0E-300.

! This is useful if NSPCG thinks that the ||RHS|| is less than machine zero and returns prematurely.

MATRix machine ZERO =1.0E-300, TOLerance=1.E-20. Generate SUMMary DIAGnostics also

COMMAND **MULTIPHASE**

PURPOSE To specify the nature of the characteristic curve and the values of the empirical characteristic constants for multiphase or variably saturated flow. This command automatically initiates the multi-phase mode of simulation.

MODE 1: **Tabulated Data Input**

SYNTAX **MULT** {**TABL**} [**COND**] [**HEAD**] [**COMP**] [**phase**] {**N1**} {**fname** | **N2, ..., Nn**}

TABL The moisture characteristic is specified in tabular form.

COND By default, the tabulated data are for the S- ψ characteristic. If the modifier **CONDuctivity** is present, the specification is assumed to be for the S- k_r characteristic.

HEAD By default, the relative moisture characteristic is assumed to be specified as S vs. k_r . If the modifier **HEAD** is present, the characteristic is assumed to be tabulated values of ψ vs. k_r .

COMP k_r for the second phase is computed from the complimentary function defined by Equation 3.5.11. The modifier **SECOnd** must be present for this input to be effective.

phase The fluid phase for which the input is specified. See Section 7.2.10 for available options. *If no phase modifier exists, the input is assumed to be for the first phase of the fluid.*

fname The name of the file from which numerical values **N2** through **Nn** for the moisture characteristic curve are read. See Section 7.2.2 for additional information.

N1 The number of sets of tabulated values which follow.

N2 The first value of S for the S- ψ or S- k_r characteristic or the first value of the ψ for the ψ - k_r characteristic. The numerical value must be ≥ 0 . See Section 3.3 for more details.

N3 The first value of ψ for the S- ψ or the first value of k_r for the S- k_r or ψ - k_r characteristic. The input value may be scaled internally by the **SCALE** command. The numerical value must be ≥ 0 .

N4, ..., Nn The sets of values of S, ψ or k_r in a manner similar to **N2** and **N3**. A total of **N1** sets (including **N2** and **N3**) must be specified.

EXAMPLES

MULTiphase flow: TABLE of 4 sets: (0,1.E6), (0.1,1.E3), (0.9,1.E2), (1.,1.)

MULTiphase: COND: TABLE 4 sets: (0,0), (0.2,0.4), (0.8,0.7), (1,1) !S vs. k_r

MULTiphase CONDuctivity COMPlimentary SECOnd phase: TABLE of 4 sets:
(0,0), (0.2,0.4), (0.8,0.7) (1., 1.)

MULTi COND vs HEAD: TABLE 4: (100,0) (60,0.4) (40,0.7) (0, 1) ! \square vs. k_r

MULTi COND vs HEAD: TABLE of 40 sets from file 'SVSKRH'

MODE 2: **The Brooks & Corey Characteristic**

SYNTAX **MULT** {**BROO**} [**BURD** | **MUAL**] [**COND**] [**COMP**] [**phase**] {**N1**, **N2**} [**N3**, **N4**, **N5**]

- BROO** Brooks and Corey relations (See Chapter 3) are used.
- BURD** k_r is computed from the Burdine (1953) predictive model. This is the default option.
- MUAL** k_r is computed from the Mualem (1976) model.
- COND** By default, the moisture characteristic is used for both phase saturation and relative hydraulic conductivity. However, if the modifier **COND** is present, the specification is used only to calculate the relative conductivity.
- COMP** See Mode 1 specification.
- phase** See Mode 1 specification.
- N1** The λ of Equation 3.3.8 (> 0). There is no default value for this input.
- N2** The ψ_a of Equation 3.3.8 (> 0). There is no default value for this input.
- N3** The residual saturation, S_r , of Equation 3.3.3 (≥ 0). The default value is 0.
- N4** The residual gas-phase saturation, S_g , of Equation 3.3.3 (≥ 0). The default value is 0.
- N5** The minimum value of k_r . The default value is 10^{-6} .

EXAMPLES

- MULT**iphase flow: BROOKs & Corey: lamda = 0.5; air entry p = 0.1
- MULT**iphase: BROOKs with MUALem: lamda=4.5, h* = 5.4, 2*0
- MULT**iphase: BROOKs for SECONd phase with: N=3.5; Aow=0.100
- MULT**iphase: BROOKs CONDUCTivity SECONd phase: lamda=3, psi=0.1
- MULT**iphase: BROOKs CONDUCTivity COMPLimentary SECONd l=3, psi=0.2

MODE 3: **The Van Genuchten Characteristic**

SYNTAX **MULT** {VAN} [MUAL | BURD] [COND] [COMP] [phase] {N1, N2} [N3, N4, N5, N6]

- VAN** van Genuchten relations (See Chapter 3) are used.
- MUAL** k_r is computed from the Mualem (1976) model. This is the default option.
- BURD** k_r is computed from the Burdine (1953) predictive model.
- COND** By default, the moisture characteristic is used for both the phase saturation and relative hydraulic conductivity. However, if the modifier **COND** is present, the specification is used only to calculate the relative conductivity.
- COMP** See Mode 1 specification.
- phase** See Mode 1 specification.
- N1** The N of Equation 3.3.9 (> 0). There is no default value for this input.
- N2** The α of Equation 3.3.9 (> 0). There is no default value for this input.
- N3** The residual saturation, S_r of Equation 3.3.3 (≥ 0). The default value is 0.
- N4** The residual gas phase saturation, S_g of Equation 3.3.3 (≥ 0). The default value is 0.
- N5** The exponent, M, of Equation 3.3.9; the default value is $1-1/N$ if the Mualem model is selected and $1-2/N$ if the Burdine model is selected.
- N6** The minimum value of k_r ; the default value is 10^{-6} .

EXAMPLES

- MULT**iphase flow VAN Genuchten: $n = 2.0$, $\alpha = 0.5$
- MULT**iphase: VAN Genuchten BURDine: $N=3.5$; $A_{lf}=0.167$, $s_r=0.067$, $s_g=0.090$
- MULT**iphase: VAN Genuchten: $N=3.5$ $a=.167$ $s_r=.067$ $S_g=.09$, $M=0.5$ $k_r=0.02$
- MULT**iphase: VAN Genuchten for CONDuctivity only with: $N=3.5$; $A_{ow}=0.1$
- MULT**iphase: VAN Genuchten COND SECOnd COMPlimentary: $N=3.5$; $a=0.1$

MODE 4: Saturation as a Function of Capillary Pressure**SYNTAX** **MULT** {EXPO | LOGA | POLY} [phase] {N1, N2} [N3, ..., N7]**EXPO** Saturation is an exponential function of ψ (Equation 3.3.11).**LOGA** Saturation is a logarithmic function of ψ (Equation 3.3.12).**POLY** ψ is a 4th-order polynomial function of saturation (Equation 3.3.13).**phase** See Mode 1 specification.**N1** The exponent N of Equations 3.3.11 or 3.3.12, or the constant A of Equation 3.3.13. **There is no default value for this input.****N2** The constant α of Equations 3.3.11 or 3.3.12, or the constant B of Equation 3.3.13. **There is no default value for this input.****N3** The constant C of Equations 3.3.11 through 3.3.13. **The default value is 0 for Equation 3.1.13 is 1 otherwise.****N4** The residual saturation, S_r of Equation 3.3.3 (≥ 0), for the exponential or logarithmic relations, or the constant D for the polynomial relation. **The default value is 0.****N5** The residual saturation, S_g of Equation 3.3.3 (≥ 0), for the exponential or logarithmic relations, or the constant, E, for the polynomial relation. **The default value is 0.****N6** The residual saturation, S_r for the polynomial relation. **The default value is 0.****N7** The residual saturation, S_g for the polynomial relation. **The default value is 0.****EXAMPLES****MULT**iphase EXPOnential characteristic: N=2., alpha=0.2**MULT**iphase LOGARithmic relation: n = 1.0, alpha = 1, C=200, sr=0.05, sg=0.**MULT**iphase POLYnomial: a = 1., b = -1.**MULT**iphase POLYnomial: a=10, b=-5., c=-2.5, d=-1.5, e=-1, sr=0.1, sg=0.05

MODE 5: Saturation as a Function of Temperature**SYNTAX** **MULT** {WHEE} {N1, N2} [N3, N4]**WHEE** Saturation relation is that of Wheeler (1973) given by Equation 3.3.14.**N1** The exponent, λ , of Equation 3.3.14 (> 0). The default value is 10.**N2** The delay temperature, T_d , of Equation 3.3.14 (> 0), for the freezing/thawing algorithm. The default value is 1.E-30.**N3** The residual saturation, S_r , of Equation 3.3.3 (≥ 0). The default value is 0.**N4** The residual saturation, S_g , of Equation 3.3.3 (≥ 0). The default value is 0.**EXAMPLES**

MULTiphas flow: with WHEEler option: n=4, Td=0.1**MULT**iphas flow: WHEEler option: n=4, Td=0.1, sr=0.2, sg=0.02

MODE 6: Relative Hydraulic Conductivity as a Function of Saturation.

SYNTAX **MULT** {COND} {POWE | POLY} [COMP] [phase] {N1, N2} [N3,..., N8]

- COND** The input is for the relative hydraulic conductivity, k_r .
- POWE** k_r is a power law of saturation (Equation 3.5.6).
- POLY** k_r is a 4th-order polynomial of saturation (Eqn. 3.5.7).
- COMP** See Mode 1 specification.
- phase** See Mode 1 specification.
- N1** The exponent N of Equation 3.5.6 for the power law relation or the constant A of Equation 3.5.7 for the polynomial relation. **There is no default value for this input.**
- N2** The constant A for the power law (**default value=1**) or the constant, B, for the polynomial relation (**default value=0**).
- N3** The constant B for the power law, or the constant C for the polynomial relation. **The default value is 0.**
- N4** The constant C for the power law, or the constant D for the polynomial relation. **The default value is 0.**
- N5** The residual saturation, S_r of Equation 3.3.3, for the power law or the constant E for the polynomial relation. **The default value is 0.**
- N6** The residual gas saturation, S_g of Equation 3.3.3, for the power law or S_r for the polynomial relation. **The default value is 0.**
- N7** The minimum k_r for the power law relation (**default value = 10^{-6}**) or S_g for the polynomial relation (**default value = 0.**)
- N8** The minimum k_r for the polynomial relation. **The default value is 10^{-6} .**

EXAMPLES

- MULT**iphase CONDuctivity: POWER law: N=4.
- MULT**iphase CONDuctivity POWER: N=4, A=1 (B,C,sr,sg)=4*0. min=1.E-7
- MULT**iphase COND COMPLimentary SECONd POWER law: N=4. A=1. sr=0.2
- MULT**iphase POLYnomial: a = 1., b = -1., C=0.5, D=-0.5

MODE 7: Relative Hydraulic Conductivity as a Function of Capillary Pressure.

SYNTAX **MULT** {COND} {EXPO | LOGA | INVE} [COMP] [phase] {N1, N2} [N3, N4]

COND	The input is for the relative hydraulic conductivity, k_r .
EXPO	Relative conductivity is an exponential function of ψ as defined by Equation 3.5.8.
LOGA	Relative conductivity is a logarithmic function of ψ as defined by Equation 3.5.9.
INVE	Relative conductivity is an inverse power law function of ψ as defined by Equation 3.5.10.
COMP	See Mode 1 specification.
phase	See Mode 1 specification.
N1	Exponent, N, of Equations 3.5.8, 3.5.9 or 3.5.10. There is no default value for this input.
N2	Constant, α of Equations 3.5.8, 3.5.9 or 3.5.10. There is no default value for this input.
N3	Constant, C, of Equations 3.5.8 through 3.5.10. The default value is 1.
N4	Minimum permissible value of k_r . Any computed value smaller than this value is set equal to this value. The default value is 10^{-6} .

EXAMPLES

MULTiphasE **EXPO**ntial **CONDU**ctivity characteristic: N=2., alpha=0.2

MULTiphasE **LOGA**rithmic **CONDU**ctivity: n = 1.0, alpha = 1, C=200.

MULTiphasE **CONDU**ctivity **INVE**rse power law: N=2., alpha = 0.05, C=1.

MULTiphasE **CONDU**ctivity **INVE**rse **COMPL**imentary **SECO**nd phase: N=2, alpha=0.05

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
PURPOSE	To modify built-in default options.
SYNTAX	OPTI [NEW OLD] [LINE GRAD FIXE] [QUAD VOLU] [ENTH TEMP]
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.
LINE	For ANSWER™ , pressure at inlet, outlet and open boundaries is computed from linear extrapolation. This is the default mode. This is equivalent to assuming that the 2 nd 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 that of LINEar extrapolation.
FIXE	For ANSWER™ , pressure at inlet, outlet and open boundaries is assumed to be fixed. This option is more suited for compressible flow computations.
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.
ENTH	The thermal diffusion term is computed from the product of the effective thermal conductivity divided by the specific heat and the gradients of the enthalpy. This leads to a very stable numerical formulation of the enthalpy equation and is accurate as long as the gradients of the specific heat are much smaller than those of the 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 gradients of temperature. This formulation is correct if enthalpy is not a linear function of temperature or the gradients of specific heat are large. However, this formulation may be numerically less stable than that described by the ENTHaply modifier above. This is the default option if conjugate heat transfer in the solid blocks (SOLID command) is specified.

EXAMPLES

- OPTI**on GRAD for pressure ! For extrapolation of pressure for this highly skewed grid
- OPTI**on use ENTHalpy ! Formulation for conjugate heat transfer
- OPTI**on use TEMPerature formulation for heat transfer even though there are no solid blocks.
- OPTI**on NEW geometry computations with QUADRatic interpolation

COMMAND	OUTLET
PURPOSE	To specify an outflow boundary for the domain of computation
SYNTAX	OUTL { dir } [subrgn] [DIAG] [OFF] [AREA N1]
dir	The orientation index for the outlet boundary. There is no default value; a value must be specified.
subrgn	The subregion to be identified as an outlet. 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 outlet boundary for the identified subregion is deactivated. A new specification for this subregion may follow.
AREA	For steady state or incompressible flow, the flow rate at each outlet port is adjusted so that the total outflow equals the total inflow (See Comments below). If this modifier is present, then the outflow rate at each port is distributed in proportion to the ratio of its area to the total outflow area. If this modifier as well as the N1 specification are omitted, then the outflow is distributed in proportion to the computed outflow rate. The default practice is equivalent to treating all the outlets as a single outlet. This modifier is effective only if multiple OUTLet commands are specified.
N1	The flow rate or flow fraction that exits through this outlet as a fraction of the total outflow through all outlet ports (See Comments below). This modifier is effective only if multiple OUTLet commands are specified.

COMMENTS

An outlet boundary, by definition, is assumed to be a boundary where the normal gradients of all dependent variables are zero. This command provides a compact way to specify the zero normal gradient boundary conditions for all variables at a given boundary. If multiple **OUTLet** commands are specified, then the flow rate through the i^{th} port, Q_i , is adjusted as:

$$Q_i = Q \frac{F_i}{\sum_i F_i}$$

Where Q is the total outflow and F_i is the fraction through the i^{th} port. If the **AREA** modifier is present, then F_i is set equal to the area of the port, if **N1** is specified, then it is set equal to **N1**. If neither **AREA** nor **N1** is specified, then F_i is set equal to the computed flow (before any adjustment). **Though it is possible to specify, a mix of these options, it is recommended that a consistent practice be followed.** That the **AREA** modifier or **N1** be either specified for all or omitted for all **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] [TABL] [plane] [subrgn] [STAT] [BOUN] [NARR WIDE] [V_{frq}] [TIME] [NOW OFF ON]
Φ	One or more of the character strings listed in Table 6.8.1 that represent the corresponding variables for which the output is desired. The output is produced in the order of specification of the variables.
TABL	For structured grids, by default, the output is presented plane-by-plane in an ordered two-dimensional tabular form. In the presence of this modifier the output is presented as a linear table of values ordered by their element number. Two commands, with and without the TABLE modifier, may be used to obtain output in both plane-by-plane and tabulated form. For unstructured grids, plane-by-plane output is not possible; the output can only be obtained as a table of values ordered by their element numbers. This modifier is ignored.
plane	One of the character strings: XY , XZ or YZ . Because three-dimensional arrays are printed in a two-dimensional, tabular format, the user is given the option of selecting the plane of presentation. By default, the tables are printed for xy planes. This modifier is effective only for structured grids.
subrgn	The subregion for which the output is required. If no subregion is specified, the entire computational domain is selected. See Sections 7.2.3 and 7.2.4.
STAT	In addition to the tabular output of the variables, 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.
BOUN	By default only a summary of geometric information about the domain boundaries is produced with the standard output. If this modifier is present, then a detailed output of the location, geometrical properties and interconnectivity of each of the boundary surfaces is produced.
NARR	The output tables are produced in an 80-column (narrow) format.
WIDE	The output tables are produced in a 132-column (wide) format. This is the default mode.
V_{frq}	The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 for further details. The 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.
NOW	Tabular output is produced immediately.
OFF	Tabular output is completely suppressed.
ON	Tabular output is reactivated if it was previously suppressed.

COMMENTS

If this command is omitted, output for the active variables is automatically produced 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.

EXAMPLES

OUTPUT: U, V, W in NARROW tabular format

OUTPUT: T, V, U and P in WIDE tabular format NOW with variable STATISTICS

OUTPUT: T, V, U and P in WIDE tabular format NOW as TABLE

OUTPUT: U, V, T and P in WIDE tabular format with BOUNDARY details

OUTPUT: for U and V for the SELECTED active subregion

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 tabular format NOW

OUTPUT: U, V, P and T in NARROW format NOW and every 20 steps

OUTPUT TABLES for V, W, P and K and L by YZ planes every 15 steps

OUTPUT OFF for all variables

OUTPUT ON once again

OUTPUT NOW for subregion specified by the most recent LOCATE command

MODE 2: Output of Properties and Coefficients for the Differential Equations**SYNTAX** **OUTP** { Φ } {DIFF|STOR|SOUR|MATR|RESI} [subrgn] [NARR|WIDE] [NOW] [N1]

Φ	One or more of the symbols that denote the dependent variable(s) 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.
DIFF	The diffusion coefficient for the variable is printed to the output file in the standard tabular format for a structured grid. If the diffusion coefficient is a tensor, then the output is produced for the diagonal component in each direction.
STOR	The storage coefficient for the variable is printed to the output file in the standard tabular format for a structured grid.
SOUR	The net source (algebraic sum of source and sink) for the variable is printed to the output file in the standard tabular format for a structured grid.
MATR	The matrix coefficients for the variable are printed to the output file in the standard tabular format for a structured grid. 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.
subrgn	See Mode 1 specification.
NARR	See Mode 1 specification.
WIDE	See Mode 1 specification.
NOW	The output is produced at the next step following the command.
N1	The output is produced at the step number given by the specified numerical value. If this value is omitted or is less than unity, then the output is produced at the end of the simulations.

COMMENTS

This command is effective only for structured grids. For unstructured grids the SAVE command may be used to generate the same information in a block format. **Each command may specify only one type of output. However, multiple commands may be used in succession to obtain different types of output.**

EXAMPLES

OUTPut: for U, T and P of DIFFusion coefficients at the final stage
OUTPut: for T of DIFFusion, coefficients NOW (at the next step)
OUTPut: for T of STORAge coefficients NOW (at the next step)
OUTPut: for T of MATRix coefficients at step number 52
OUTPut: for T of STORAge coefficients at step number 52
OUTPut: for C : SOURce at step number 52
OUTPut: of MATRix for T and P at step number 75
OUTPut: of RESIdue for T at the end

MODE 3: Output of Balance of Convective and Diffusive Flux for each Element**SYNTAX** **OUTP** { ϕ } { FLUX } [CONV | DIFF] [AREA] [subrgn] [NARR | WIDE] [NOW] [N1]

ϕ	One or more of the symbols that denote the dependent variable(s) 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.
AREA	The computed flux is divided by the projected area of the face for output.
subrgn	See Mode 1 specification.
NARR	See Mode 1 specification.
WIDE	See Mode 1 specification.
NOW	The output is produced at the next step following the command.
N1	The output is produced at the step number given by the specified numerical value. If this value is omitted or is less than unity, then the output is produced at the end of the simulations.

COMMENTS

This command is effective only for structured grids. For unstructured grids the SAVE command may be used to generate the same information in a block format for later post-processing. Each command may specify only one type of output; however multiple commands may be specified to obtain output of different types.

EXAMPLES

OUTPut: FLUX for T for each element at the final stage
OUTPut: CONVective FLUX for T NOW (at the next step)
OUTPut: DIFFusive FLUX for T at step number 52
OUTPut: DIFFusive FLUX divided by AREA for T at step number 52
OUTPut: DIFFusive FLUX divided by AREA for T at step # 52 in NARRow mode for ID=SUBREGION

COMMAND **PAUSE**

PURPOSE To cause a temporary halt in the calculations.

SYNTAX **PAUS**

COMMENTS_____

Operator intervention is required to restart the calculation process. This command allows for operator action during interactive execution.

EXAMPLES_____

PAUSe and await operator action

COMMAND PERIODIC

PURPOSE To select the periodic boundary option for boundary conditions of the domain.

MODE 1: Periodic Boundary Without Velocity Transformation

SYNTAX PERI { X | Y | Z | fname } [option]

X | Y | Z The boundary conditions for the corresponding x, y or z direction boundaries are determined automatically from the requirement of periodicity. A **BOUNDary** command with a corresponding boundary index (X-, X+ or Y-, Y+, or Z-, Z+) must not be specified if this option is selected.

fname The name of the file which contains the information about the periodic boundary locations. This file must contain a list with one record for each pair of matching periodic surfaces (called “lower” and “upper”). Each record must consist of only 4 integers in the order:

1. Element number for the periodic surface at “lower” periodic boundary.
2. Surface number for the periodic surface (or face) of the element above.
3. Element number containing the periodic surface at “upper” periodic boundary.
4. Surface number for the periodic surface (or face) of the element above.

The definition of the “lower” or “upper” boundary is arbitrary. The element numbers that appear in the periodic list must consist only of the internal field elements. Boundary node numbers must not appear in this list. The surfaces are numbered from 1 to 4 for Quad and from 1 to 6 for Hex elements. For hybrid unstructured grids, the number of faces depends on the type of element. See **LOCAt**e command with **PAIR** modifier and Sections 7.2.3 and 7.2.4 for further details of the manner in which these surface numbers are assigned.

option Options selected for implementation of the **PERIodic** command.

option	INTERPRETATION
LINE	For a structured grid, by default, an ADI cyclic solver is used in the periodic direction. If this modifier is present, then a linear form of the ADI solver is used. This modifier has no effect if the solver used does not have a cyclic option. Of the default solvers only the ADI has the cyclic option. Please check with ACRI for other solvers.
CYCL	For an unstructured grid, by default, a linear solver is used in the periodic direction. The periodicity is imposed explicitly by forcing the boundary values to be periodic. If this modifier is present, then the cyclic solver is used. The user must ensure that the unstructured grid is setup in a manner that the ADI “lines” in the periodic direction run unbroken from one boundary to the other. Otherwise the solution may be corrupted with unforeseen consequences
Φ	One or more (≤ 54) symbols that denote the variables which are periodic (or not periodic if OFF modifier is specified) at the θ_1 and θ_2 boundaries. By default all variables are assumed to be periodic.
OFF	The periodic boundary conditions are not applied to the variables specified by the Φ modifier. All variable not so identified are assumed to be periodic.
PRES	By default it is assumed that the pressure at a periodic surface is also cyclic; that is there is no net pressure loss in the direction of periodicity. If this modifier is

	present, then any initial pressure difference between the matched elements of the periodic surfaces is maintained.
--	--------------------------------------------------------------------------------------------------------------------

EXAMPLES

PERIodic boundaries in X

PERIodic boundaries in X, Y and Z

PERIodic boundary in X for U,V,W,T,P only

PERIodic boundary at X option OFF for P and FU variable

PERIodic boundary surface data on file: 'PERIODIC.DAT'

PERIodic boundary in Z direction with LINEAR solver

PERIodic boundary unstructured grid but use CYCLic solver; data on file: 'PERIODIC.DAT'

MODE 2: Periodic Boundary With Radial to Cartesian Velocity Transformation

SYNTAX PERI { X | Y | Z | fname } {THET} [option] [N1, N2, N3]

X | Y | Z The boundary conditions for the corresponding x, y or z direction boundaries are determined automatically from the requirement of periodicity. A **BOUNDary** command with a corresponding boundary index (X-, X+ or Y-, Y+, or Z-, Z+) must not be specified if this option is selected.

fname See Mode 1 specification.

THET This modifier denotes that the velocity components are Cartesian but the problem is periodic in the circumferential (θ) direction of a cylindrical coordinate system. At times a problem which is periodic in cylindrical (x,r,θ) coordinates is solved instead in a Cartesian (x,y,z) system. Then, the Cartesian (U,V,W) velocity components at the two periodic boundaries located at $\theta = \theta_1$ and $\theta = \theta_2$ obey the transformed periodicity relations:

$$\begin{aligned}
 U_2 &= U_1 \\
 V_2 &= V_1 \cos \Psi - W_1 \sin \Psi \\
 W_2 &= W_1 \cos \Psi + V_1 \sin \Psi
 \end{aligned}$$

where

$$\Psi = \theta_2 - \theta_1$$

option See Mode 1 specification.

N1 The angle of periodicity, Ψ , in degrees. There is no default value, if the modifier THET is present, then a value must be specified.

N2 A number ($1 \geq N2 \geq 3$) that denotes the order of the radially-directed component of velocity among the three velocity components (U,V,W). The default value is 2.

N3 A number ($1 \geq N3 \geq 3$) that denotes the order of the circumferentially-directed component of velocity among the three velocity components (U,V,W). The default value is 3.

EXAMPLES

- PERIodic boundaries in Z with THETA = 45 degrees
- PERIodic boundaries in Y with THETA = 45 degrees v = 1, w = 2
- PERIodic boundary in Z with THETA = 18 degrees for U,V,W,T,P only
- PERIodic boundary in Z with THETA = 18 degrees but option OFF for P and FU variable
- PERIodic boundary at THETA = 20 degrees surface data on file: 'PERIODIC.DAT'

MODE 3: Periodic Boundary With General Velocity Transformation

SYNTAX PERI {X|Y|Z|fname} {TRAN|STAC} [option] [N1, ..., N9]

X|Y|Z The boundary conditions for the corresponding x, y or z direction boundaries are determined automatically from the requirement of periodicity. A **BOUNDary** command with a corresponding boundary index (X-, X+ or Y-, Y+, or Z-, Z+) must not be specified if this option is selected.

fname See Mode 1 specification.

TRAN The velocity vector at the identified periodic boundaries is transformed according to:

$$\begin{bmatrix} U2 \\ V2 \\ W2 \end{bmatrix} = \begin{bmatrix} n1 & n2 & n3 \\ n4 & n5 & n6 \\ n7 & n8 & n9 \end{bmatrix} \cdot \begin{bmatrix} U1 \\ V1 \\ W1 \end{bmatrix}$$

STAC The velocity vector at the identified periodic boundaries is transformed according to the transformation specified in the most recent **STACK TRANSformation** command.

option See Mode 1 specification.

N1,..N9 The transformation matrix to compute the velocity components at the “higher” (say, at Z+) periodic boundary from those at the “lower” (say, at Z-) periodic boundary if the **TRAN modifier is present**. This information is ignored otherwise. **There is no default value, if the modifier TRAN is present, then a set of 9 values must be specified.**

EXAMPLES

PERIodic boundaries in Z Velocity **TRANS**formed by:

```
1      0      0
0      0     -1
0      1      0
```

PERIodic boundaries in Z Velocity **TRANS**formed by:

```
1.0      0.      0.0
0.0      0.995004  -0.0998334
0.0      0.0998334  0.995004
```

PERIodic boundaries in Z Velocity by **STACK TRANS**formation

PERIodic boundary in X by **STACK TRANS**formation for U,V,W,T,P only

PERIodic boundary **TRANS**formation as follows with surface data on file: 'PERIODIC.DAT'

```
1.0      0.      0.0
0.0      0.995004  -0.0998334
0.0      0.0998334  0.995004
```

PERIodic boundary by **STACK TRANS**formation surface data on file: 'PERIODIC.DAT'

COMMAND PHASE

PURPOSE To specify phase change option for multi-phase option.

SYNTAX **PHAS** {CHAN | CLAY | VAPO } [TEMP | T] [N1] [N2, N3, N4]

CHAN Phase change occurs at a fixed temperature.

CLAY Phase change occurs according to Clayperon's vapor-pressure relation.

$$P_s = P_0 \text{Exp} \left(a_1 - \frac{a_2}{T_s - T_0} \right) ,$$

where, P_s is the vapor pressure, T_s is the phase change temperature, and P_0 , T_0 , a_1 and a_2 are empirical constants.

VAPO Same as **CLAY**.

TEMP The temperature at which phase change occurs if the **CHANge** modifier is specified; otherwise this input is ignored.

T Same as **TEMP** modifier.

N1 The temperature at which the phase change occurs if **CHANge** modifier is specified; otherwise it is the base temperature T_0 of the vapor pressure equation given above. **The default value is 45.**

N2 The pressure P_0 of the vapor pressure equation. **The default value is 131.57894.**

N3 The constant a_1 of the vapor pressure equation. **The default value is 18.3443.**

N4 The constant a_2 of the vapor pressure equation. **The default value is 3841.1954.**

COMMENTS

The phase change option is an advanced option which is currently available only under a consulting arrangement with **ACRI**.

EXAMPLES

PHASe CHANGE at 100 Degrees Centigrade

PHASe CHANGE according to CLAYperon's equation: T0=45, p0=131.57894, a1=18.3443, a2=3841.1954

COMMAND PRECIPITATION**PURPOSE** To specify the precipitation or uptake reactions for dissolved species.**MODE 1:** **Precipitation Reaction with Constant Solubility.****SYNTAX** **PREC {idreac}{Φ}{Ψ} {ω, Cs} [subrgn]**

idreac A unique identifier for the precipitation (or uptake) reaction rate where the species Φ precipitates to generate species Ψ or the species Ψ is converted to species Φ by uptake . The reaction rate is computed as:

$$\text{Rate} = \omega (C_s - \Phi) .$$

The computed rate is algebraically added to the equation for Ψ and subtracted from the equation for Φ. The reaction rate units are the units of the species divided by time.

Φ The symbol for the species that is precipitated or re-adsorbed by the reaction.

Ψ The symbol for the species that is produced by precipitation or dissolved by uptake.

ω The reaction time constant (units of frequency) for the precipitation/uptake reaction rate. **There is no default value; a value must be specified.**

C_s The solubility limit for the species (units of Φ). **There is no default value; a value must be specified..**

subrgn The subregion for which the idreac reaction is computed. **If no subregion is specified, the reaction is computed for the entire computational domain.** See Sections 7.2.3 and 7.2.4.

EXAMPLES

PRECipitation reaction R1 from C1 to C2 with frequency = 1000; Solubility = 0.02

PRECipitation reaction R2 from C2 to C3 with time constant = 0.05 ; Solubility = 0.30

MODE 2: Precipitation Reaction with Variable Solubility**SYNTAX** **PREC** {idreac} { Φ } { Ψ } { C_S } { ω } [subrgn]

idreac A unique identifier for the precipitation (or uptake) reaction rate where the species Φ precipitates to generate species Ψ or the species Ψ is converted to species Φ by uptake. The reaction rate is computed as:

$$\text{Rate} = \omega (C_S - \Phi) .$$

The computed rate is algebraically added to the equation for Ψ and subtracted from the equation for Φ . The reaction rate units are the units of the species divided by time.

Φ The symbol for the species that is precipitated or re-adsorbed by the reaction.

Ψ The symbol for the species that is produced by precipitation or dissolved by uptake.

C_S The symbol for a valid variable which represents the solubility limit for the species as a function of space, time or other state variables defined by the ACRi software.

ω The reaction time constant (units of frequency) for the precipitation/uptake reaction rate. *There is no default value; a value must be specified.*

subrgn The subregion for which the idreac reaction is computed. *If no subregion is specified, the reaction is computed for the entire computational domain.* See Sections 7.2.3 and 7.2.4.

EXAMPLES

PRECipitation reaction R1 from C1 to C2 with solubility defined by C3 and a frequency = 1000

PRECipitation reaction R2 from C2 to C3 with C4 as solubility and time constant = 0.05

MODE 3: Precipitation Reaction with Variable Solubility

SYNTAX **PREC** { ELEM } { idreac } { Φ } { Ψ } { ξ } { ω, C_{Sat} } [subrgn]

ELEM Element mode of the precipitation reaction is activated. In this mode, the solubility of the isotope is decreased by the presence of other elements according to a modified Raoult's Law. With C_{Sat} as the solubility of the pure element, the solubility, C_S, in the presence of other elements is given by:

$$C_S = C_{Sat} \frac{(M_i^{liquid} + M_i^{solid})}{\sum_{j=1}^n (M_j^{liquid} + M_j^{solid})}$$

where M^{liquid} and M^{solid} are, respectively, the molar concentration of elements in the liquid and solid phases.

idreac A unique identifier for the precipitation (or uptake) reaction rate where the species Φ precipitates to generate species Ψ or the species Ψ is converted to species Φ by uptake. The reaction rate is computed as:

$$\text{Rate} = \omega (C_S - \Phi) .$$

The computed rate is algebraically added to the equation for Ψ and subtracted from the equation for Φ. The reaction rate units are the units of the species divided by time.

Φ The symbol for the species that is precipitated or re-adsorbed by the reaction.

Ψ The symbol for the species that is produced by precipitation or dissolved by uptake.

ξ The symbol for the variable that represents the sum of the molar concentrations of all the species that participate in reducing the solubility of the element; **the denominator in the above equation for C_S.**

ω The reaction time constant (units of frequency) for the precipitation/uptake reaction rate. **There is no default value; a value must be specified.**

C_{Sat} The solubility limit for the species (units of Φ) in its pure element form. **There is no default value; a value must be specified..**

subrgn The subregion for which the idreac reaction is computed. **If no subregion is specified, the reaction is computed for the entire computational domain.** See Sections 7.2.3 and 7.2.4.

COMMENTS

For this mode of input, it is preferable that all species concentrations be expressed in terms of moles. In this case the **SET SUM** command can be used to define the sum of molar concentrations, ξ. If the species concentrations are in terms of mass units, then **SET LINEAR** command must be used to define ξ with appropriate coefficients.

EXAMPLES

PRECipitation **ELEM**ENT form: R1 for C1 and C2 with molar variable C3 frequency = 1000, csat=0.05

PRECipitation **ELEM**ENT form: R1 for C1, C2 molar variable C3 omega=1000, csat=0.05 ID=REAC1

PRECipitation **ELEM**ENT form: R1 for C1, C2, sum=C3, omega=1000, csat=0.05 **SELEC**Ted region

COMMAND	PRINT
PURPOSE	To generate output of details of mass flow rate and statistical measures of flow variables.
MODE 1:	Print Statistics for Flow and Selected Variable
SYNTAX	PRIN [FLOW] [STAT] [Φ] [GEOM] [subrgn]
FLOW	The mass inflow into and, outflow from, all inlets, outlets, open boundaries and sources are computed and written to the standard output device as soon as the command is encountered.
STAT	The statistics related to the minimum, maximum and average value of specified variable(s) is computed and written to the standard output device as soon as the command is encountered. The variables for which the statistics are computed must be identified by the Φ modifier(s)
Φ	One or more symbols that denote the dependent variables for which the STAT modifier is effective. The valid symbols are listed in Table 6.8.1. There is no default value.
GEOM	The face areas and volume of the subregion identified on the command are printed on the output file.
subrgn	The subregion for which the output of face areas and volume is required. If no subregion is specified, the entire computational domain is selected. See Sections 7.2.3 and 7.2.4.

EXAMPLES

PRINT FLOW immediately

PRIN STATistics for U, V, T now

PRIN FLOW and STATistics for U, V, T now

MODE 2: Integrated Averages for Selected Variables at Uniformly Spaced Locations

SYNTAX **PRIN** { Φ } {**AVER** | **PROF**} {**N1**} {**coordinate**} [**subrgn**] [**dir**] [**TOLE=V_{tol}**] [**NORM**]
 [**BASE=V_{base}**] [**MASS** | **AREA** | **VOLU**] [**OUTP** | **fname**]] [**V_{frq}**] [**TIME**]

Φ One or more of the symbols that represent the variables for which output is desired. The valid symbols are listed in Tables 6.8.1 and 6.8.2. There is no default value.

AVER The averages of the specified variable are computed at a number of specified locations.

PROF Same as **AVER** modifier.

N1 The number of locations at which the averages are computed. It must be the first numerical value on the command. There is no default value; a numerical value (>0) must be specified.

coordinate One of: **X**, **Y**, **Z**, **R** or **THETA** modifiers. It denotes the coordinate of locations at which the averages are computed. The **Z**, **R** or **THETA** can only be used for three-dimensional geometry. In this case is $R = \sqrt{y^2 + z^2}$ and $THETA = \arctan(z/y)$.

subrgn The subregion for computations. If no subregion is specified, the entire domain is selected.

dir The orientation index for the boundary of the sub-domain if the averages are to be computed only at the boundary. See Section 7.2.5 for available choices. There is no default value.

TOLE The coordinate tolerance for inclusion of the elements in averaging at a location. With S_i as the **coordinate** of the i^{th} location, all elements that satisfy: $S_i - V_{tol} \leq S < S_i + V_{tol}$ are included in averaging. By default, the tolerance is set equal to half the interval between successive uniformly spaced coordinates from 0 to 1. For example if **N1**=10, then interval will be 0.1 and V_{tol} will be set of 0.05. However if the modifier **TOLE** is present, then the tolerance is set to the user specified value V_{tol} .

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

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

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

$$\Phi_{output} = \frac{\Phi_{computed} - \Phi_{base}}{\Phi_{norm} - \Phi_{base}}$$

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

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

V_{base} The user specified value for Φ_{base} . This input must be specified only if the **BASE** modifier is present and then it must immediately follow the modifier.

MASS	This modifier is significant only if the BASE modifier is specified. By default Φ_{norm} is taken to be arithmetic mean of the values for the sub-domain. If this modifier is present, then Φ_{norm} is the mass or mass-flux weighted mean of the variable.
AREA	This modifier is significant only if the BASE modifier is specified. By default Φ_{norm} is the arithmetic mean of the values for the sub-domain. If this modifier is present, then Φ_{norm} is the area-weighted mean in the presence of the dir modifier or volume-weighted mean otherwise.
VOLU	Same as the AREA modifier.
OUTP	The output is directed to the standard output file unit.
fname	The file name for output. If a file name is present then the output is directed to the named file, otherwise the output is directed to the standard output device. The total number of open files in any simulation can not exceed 64.
V_{frq}	The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 for further details. If V_{frq} is specified, it must be the last numerical input on the command. The default value is set so that output is obtained only at the end of simulations.
TIME	By default, V_{frq} is the frequency of output in terms of number of steps. If this modifier is present, then V_{frq} is interpreted to be time interval between successive outputs.

COMMENTS

This command generates a series of profiles of integrated averages for a variable. For example, if the selected coordinate is **X**, the **subrgn** denotes a 3D sub-domain and there is no **dir** modifier, then this command will generate integrated averages across the yz planes of the sub-domain at **N1** uniformly spaced locations in the x-direction. If the **subrgn** and **dir** denote a xy plane, and the coordinate selected is **X**, then it generates integrated averages across the y-direction at **N1** locations along the x-coordinate of the plane. This command with the **BASE** modifier can be used to generate the pattern and profile factors which are commonly used in the aircraft and gas turbine industry to denote the variation of temperature at the outlet plane of the combustor. In this case, the variable Φ should be the symbol **T** (for Temperature) and the base value, **V_{base}**, should be specified as the average inlet temperature (commonly called T3). The Φ_{norm} is automatically selected to be the computed arithmetic mean (generally called T4). If T4 is to be mass-weighted mean then the modifier **MASS** must also be present.

EXAMPLES

```

PRINT Profile of T at 20 X locations      ! 20 values versus x for whole of the domain
PRINT PROFile of U, V, T at 20 X locations      ! Multiple variables
PRINT AVERAge of U at 20 X locations for ID=OUTLET in X+ direction      !Output for outlet plane.
PRINT PROFile of STATistics for T at 20 X locations for ID=OUTLET in X+ direction      !min, max & stats
PRINT PROFile of NORMalized T at 20 X locations for ID=OUTLET in X+ direction
PRINT AVERAge for T at 20 R for ID=OUTLET. in X+ direction BASE value = 375
PRINT AVERAge for T at 20 R for ID=SECTION1 BASE value = 375 MASS weighted
PRINT AVERAge for T at 20 Y locations with TOLerance=0.01 for SELEcted subregion in X+ direction
PRINT AVER for T at 20 Y locs with TOLerance=0.01 and BASE=375 for ID=OUTLET in X+ every 50 steps
PRINT AVER T 20 Y locs TOLerance=0.01, BASE=375 , ID=OUTLET X+ at TIME interval of 0.05 units

```


MODE 4: Forces and Moments for a Selected Variable at a Specified Location

SYNTAX **PRIN** Φ { **FORC** | **MOME** } [subrgn] [dir] [X_i] [Φ_0] [V_{freq}] [TIME] [NOW] [OFF] [fname]

Φ The symbol for the variable for which the “force” or “moment” is to be computed. The valid symbols are listed in Tables 6.8.1. *if no symbol is specified then pressure, P, is selected.*

FORC “Force” and “Moment” of a variable for the surface of a selected subregion are defined as:

$$F_i = \int (\Phi + \Phi_0) n_i dA$$

$$M_i = F_i \otimes (X_i - Y_i)$$

F_i is the force vector for the Φ variable,

Φ_0 is a reference datum for the Φ variable,

n_i is the normal vector at the surface of the subregion,

A is the area of the surface,

M_i is the moment vector for the Φ variable,

⊗ denotes the cross product of vectors,

X_i is the location of the pivot point around which the moment is computed,

Y_i is the point of intersection of the vector from X_i and the force vector such that the vector from X_i is normal to force vector. This is automatically computed.

MOME Same as **FORC**e modifier.

subrgn The subregion for computations. *If no subregion is specified, the entire domain is selected.*

dir The orientation index for the boundary of the subregion for which the force is computed. See Section 7.2.5 for available choices. *If no value is specified then the total (closed) boundary of the subregion is elected. In this case the computed force will be the net force on the selected body.*

X_i The coordinates of the pivot point around which the moment of the force is computed. *Two values must be specified for 2D and three for 3D geometry. If specified these must be the first set of numerical values on the command. The default value is zero.*

Φ_0 Datum for the selected variable Φ . *If specified this must be the 3rd numerical value for 2D and 4th numerical value for 3D geometry. The default value is zero.*

V_{freq} The frequency (step or time interval) at which the output is written to the output device. See Section 7.2.11 for further details. *If specified this must be the 4th numerical value for 2D and 5th numerical value for 3D geometry. By default the output is obtained only at the end of simulations.*

TIME *By default, V_{freq} is interpreted as the frequency of output in terms of number of steps. If this modifier is present, then it is interpreted as the time interval between successive outputs.*

NOW The force and moment are computed at the first step after the command.

OFF Any previously specified command for the specified Φ and **subrgn** is deactivated.

fname The file name for output. *The default file name is ‘acr_FORCE.TMP’. A different output file may be specified for each command. If no file name is specified then the output is directed to the most recent previously specified file. If no previous file was specified, then the output is directed to the default file. The total number of open files can not exceed 64. A summary of output is also printed to the standard output device at the end of simulations.*

EXAMPLES

PRINT FORCE for ID=BODY

PRINT FORCE of P for ID=BODY in X+ direction with moments around point (0., 2.354, 5.75)

PRINT FORCE of T for ID=BODY in X+ dir; around (0., 2.35, 5.7); add Pref = 1.E5 file='FORCE_FIL

PRINT FORCE of U ID=BODY in Y+ dir with moments (0., 0.); add Pref = 1.E5 every 20 steps !2D case

PRINT FORCE ID=BODY in Z+ dir; pivot (0., 2.35, 5.7); Pref = 1.E5, every 20 steps

PRINT FORCE Z+ of ID=BODY; pivot (0., 2.35, 5.7); Pref = 1.E5, every TIME = 100 'FORCE_OUT' NOW

PRINT FORCE of P OFF for Z+ direction of ID=BODY

MODE 5: Convective and Diffusive Fluxes for a Variable at Specified Boundaries

SYNTAX **PRIN** {FLUX} [Φ] [DETA] [option] [subrgn] [dir] [fname] [V_{frq}] [TIME] [NOW] [OFF]

FLUX The convective and diffusive fluxes for a the variable at specified boundaries are written to the output file. The output includes the mean, minimum and maximum values for the variable, and the flow rate at the boundary (or boundaries).

Φ One or more symbols for the dependent variable for which output is required. Up to 10 symbols may be specified per command. The valid symbols are listed in Table 6.7.1. If no symbol is specified then output is obtained for each active variables for which a transport equations is solved.

DETA By default only a summary of the total fluxes and variable values for each boundary is written to the output device. If this modifier is present, then output also contains the detail for each segment (element) of the boundary.

option The boundary type for which output is required. More than one option may be selected. If an option is specified, then the subrgn and dir modifiers are ignored. If no option is specified, then the averages are computed for the specified subrgn and dir modifiers.

option	INTERPRETATION
INLE	Boundaries specified by the INLEt command are selected.
OUTL	Boundaries specified by the OUTLet command are selected.
OPEN	Boundaries specified by the OPEN command are selected.
IO	All boundaries specified by INLEt , OUTLet or OPEN command are selected.
WALL	Walls specified by WALL or BLOCK command are selected.
EXTE	All external (or outer) boundaries of the computational domain are selected.
ALL	All of the above boundaries are selected.

subrgn The subregion for computations. This specification is ignored if any of the option modifiers is present. If no subregion is specified, the entire domain is selected.

dir The orientation index for the boundary for which the output is required See Section 7.2.5 for available choices. If no input is given, then the output is obtained for all boundaries of the subrgn. This specification is ignored if any of the option modifiers is present.

fname The file name for output. If a file name is present then the output is directed to the named file, otherwise the output is directed to the standard output device. The total number of open files in any simulation can not exceed 64.

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

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

NOW A record of fluxes is written to the output device at the next time step when the equation for the dependent variable is solved. This is in addition to the output from the V_{frq} specification.

OFF Any previously specified **PRINT FLUX** commands for the specified variables and subregion are disabled. New commands may be subsequently specified.

EXAMPLES

PRINT FLUX for EXTErnal boundaries for all variables

PRINT FLUX of U , V and T for INLET boundaries at every 100 steps

PRINT FLUX of U and T for X- direction of ID=OBJECT at every TIME=20 units

PRINT FLUX of T for X+ direction of currently SELEcted subregion at the end of simulations

PRINT FLUX of T for OUTLET boundaries at every 100 steps on file "FLUX_T_OUTLET.FIL"

PRINT FLUX of T for ALL boundaries NOW and at every TIME=20 units on file "FLUX_T_ALL.FIL"

PRINT FLUX of T OFF for X- direction of ID=OBJECT

MODE 6: Convective and Diffusive Fluxes for a Variable at Specified Boundaries

SYNTAX **PRIN** {**SHEA**} [**DETA**] [**option**] [**subrgn**] [**dir**] [**fname**] [**V_{frq}**] [**TIME**] [**NOW**] [**OFF**]

SHEA The shear stresses and y^+ at specified boundaries are written to the output file. The y^+ is the normalized distance to the node which is meaningful for turbulent boundary layer type of flows. For laminar flow, it is reported as zero. The output also includes the normal distance from the boundary to the nearest fluid node and the mean velocity at the near boundary nodes. *Strictly speaking, the output consists of the momentum gain or loss at the boundary and not the shear stress. However, in most instances where a boundary-layer type of flow occurs, the two can be considered to be equivalent.*

DETA *By default only a summary of the total fluxes and variable values for each boundary is written to the output device. If this modifier is present, then output also contains the detail for each segment (element) of the boundary.*

option The boundary type for which output is required. More than one option may be selected. *If an option is specified, then the subrgn and dir modifiers are ignored. If no option is specified, then the averages are computed for the specified subrgn and dir modifiers.*

option	INTERPRETATION
INLE	Boundaries specified by the INLEt command are selected.
OUTL	Boundaries specified by the OUTLet command are selected.
OPEN	Boundaries specified by the OPEN command are selected.
IO	All boundaries specified by INLEt , OUTLet or OPEN command are selected.
WALL	Walls specified by WALL or BLOCK command are selected.
EXTE	All external (or outer) boundaries of the computational domain are selected.
ALL	All of the above boundaries are selected.

subrgn The subregion for computations. *This specification is ignored if any of the option modifiers is present. If no subregion is specified, the entire domain is selected.*

dir The orientation index for the boundary for which the output is required See Section 7.2.5 for available choices. *If no input is given, then the output is obtained for all boundaries of the subrgn. This specification is ignored if any of the option modifiers is present.*

fname The file name for output. *If a file name is present the output is directed to the named file, otherwise the output is directed to the standard output device. The total number of open files can not exceed 64.*

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

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

NOW A record of fluxes is written to the output device at the next time step when the equation for the dependent variable is solved. This is in addition to the output from the V_{frq} specification.

OFF Any previously specified **PRINT SHEAR** commands for the specified variables and subregion are disabled. New commands may be subsequently specified.

EXAMPLES

PRINT SHEAR stress for all WALLS

PRINT SHEAR for EXTERNAL boundaries every 100 steps

PRINT SHEAR T for X- direction of ID=OBJECT at every TIME=20 units

PRINT SHEAR for X+ direction of currently SELEcted subregion at the end of simulations

PRINT SHEAR for OUTLET boundaries at every 100 steps on file "SHEAR_T_OUTLET.FIL"

PRINT SHEAR for ALL boundaries NOW and at every TIME=20 units on file "SHEAR_T_ALL.FIL"

PRINT SHEAR OFF for X- direction of ID=OBJECT

COMMAND **PROBLEM****PURPOSE** To specify the general nature and type of problem to be solved.**MODE 1:** **Free Surface or Unconfined Flow****SYNTAX** **PROB { UNCO | FREE SURF } [dir] [AREA]****UNCO** Unconfined, free-surface mode of simulation is selected.**FREE** Unconfined, free-surface mode is selected if the modifier **SURF** is simultaneously present.**SURF** Unconfined, free-surface mode is selected if the modifier **FREE** is simultaneously present.**dir** The coordinate direction that is normal to the plane of the phreatic surface. This direction must be one of the **X, Y, Z, R** or **THETA** modifiers which, respectively, denote the x, y, z, r or θ coordinates. **By default, the y direction is assumed to be the direction of the normal to the phreatic surface for two-dimensional problems and the z direction for three-dimensional problems.****AREA** Unconfined areal mode is selected. In this mode, the equations are solved in a two-dimensional integrated form where the effect of the third direction appears as a dynamically variable depth of the water body.**COMMENTS**

By default, the upper and lower limits for the pressure head are automatically computed from the coordinates and datum specifications. The default limits represent the phreatic surface when the surface reaches either the bottom of the aquifer or the top of the ground surface. However, the user may explicitly alter these limits. Alteration may be necessary, for example, when the bottom of the aquifer is not at the bottom of the grid system or when ponding occurs at or above the ground surface. In these cases, the lower value of the pressure head (representing the bottom of the aquifer) is specified as the P2 pressure field and the upper value (representing the top of the ponding surface) is specified as the P3 pressure field.

EXAMPLES**PROBLEM** is UNCONfined**PROBLEM** with FREE SURF**ace**; bedding plane normal to direction Y**PROBLEM** with FREE SURF**ace** in X direction**PROBLEM** with FREE SURF**ace** in AREAl mod

MODE 2: **Active Two or Multiple Phase with Variable Saturation**

SYNTAX **PROB { TWO | MULT }**

TWO The active two phase variable saturation mode is selected. A pressure equation is solved for each of the two phases. The saturation of the 1st phase is determined for the capillary pressure (the difference between the 2nd and 1st phase pressure). The saturation of the 2nd phase is equal to one minus the saturation of the 1st phase.

MULT The general multi-phase variable saturation mode is selected. The number of pressure equations solved is determined from user input (see comments below). This mode is also automatically activated by a **MULTiphase** command. The saturation of the last phase is always obtained as one minus the sum of the saturation of the other phase.

COMMENTS

For a typical vadose zone problem, involving water as the primary phase and air as the secondary phase, **there are two phases but only 1 pressure equation is solved**. It is assumed that the pressure of air phase is constant. For this type of flow only 1st (water) phase is active; the 2nd (air) phase is passive.

If the 2nd phase is also active and its pressure is not constant, then a second pressure equation must be solved. **In this case the MULTiphase command with TWO modifier must be specified.**

The next stage of complexity in multiphase flow is reached when there are two active phases of immiscible liquids (say, water and oil) and one passive phase (say, air). The pressure of the air phase is assumed constant and only 2 pressure equations, those for water and oil phase, are solved.

The most complex option involves three active phases. In this case 3 pressure equations, one for each phase, are solved.

With the exception of the two active phases, all other modes of solution can be automatically determined by PORFLOW from the user input.

EXAMPLES

PROBLEM is TWO phase with water and oil as the liquid

PROBLEM is MULTiphase with three phases

MODE 3: Phase Change with Freezing and Thawing

SYNTAX **PROB** {FREE} [N1]

FREE Freezing/thawing phase-change mode is selected.

N1 An empirical constant to determine the maximum rate of freezing or thawing from step to step. A small value results in gradual changes in freezing/thawing behavior; a large value results in more abrupt changes from one time step to another. The default value is 1000.

EXAMPLES

PROBLEM with phase change: FREEzing

PROBLEM with phase change: FREEzing rate=500

MODE 4: Phase Change with Evaporation and Condensation**SYNTAX** **PROB** {EVAP} [EQUI | NONE] [FIXE] [N1,N2,N3,N4]**EVAP** Evaporation/condensation phase-change mode is selected.**EQUI** Generally, a thermodynamic equilibrium or non-equilibrium mode of phase change is automatically selected based on the nature of the problem. However, if this modifier is present the equilibrium mode is selected.**NONE** The non-equilibrium mode of phase change is employed.**FIXE** The time constant, **N1**, for phase change is adjusted for transient flow to account for the changes in time step. Presence of the **FIXEd** modifier results in **N1** being held at its specified value.**N1** The time constant C_m of Equation 3.7.1. The constant has units of $[t^{-1}]$ and represents the speed of phase change due to a balance between hydraulic conductivity (to replenish the fluid), thermal energy (to satisfy the enthalpy demand) and mass transfer (to restore equilibrium). The numerical value represents the limiting (slowest) process among these three. **The default value is 0.0025.****N2** Thermal tolerance for phase change. If the local temperature differs from the equilibrium value by less than **N2**, the effect of phase change on thermal energy balance is ignored. **The default value is 0.01.****N3** Exponent α of Equation 3.7.1. **The default value is 0.0625.****N4** The relaxation factor for the source term. The new source term at any stage is computed as a weighted average of the previous value and the newly computed value. **The default value is 1.0.****EXAMPLES**

PROBLEM with EVAPoration**PROBLEM** NONEquilibrium EVAP: $C_m=0.01$, $dt=0.02$, $power=0.20$, $RF=0.5$

COMMAND **PROPERTY**

PURPOSE To specify the mode of interpretation and calculation of the properties of the host porous matrix.

SYNTAX **PROP** [ϕ] [**EFFE**] [*option*] [**SATU**] [α]

ϕ One or more of the symbols that represent a corresponding variable for which the input is effective. The valid symbols are listed in Table 6.7.1. If no symbol is specified, the input is assumed to be effective for all variables.

EFFE By default, **N1** in the **THERmal** command is assumed to be the specific heat of the solid, c_s , and **N1** in the **TRANsport** command is assumed to be the partition coefficient, k_d . The matrix properties are computed as weighted averages of the solid and fluid properties. However, if the modifier **EFFE** is present, then **N1** in the **THERmal** command is interpreted to be the effective volumetric specific heat, α_h , of the matrix and **N1** on the **TRANsport** command is assumed to be the retardation factor, R_d . The existence of fluid in the porous matrix is assumed to have been already accounted for by the user and any values directly specified for the fluid are ignored.

option The diffusion coefficient is computed at a element interface as described in the table below.

option	INTERPRETATION
ARIT	Arithmetic mean of the two nearest grid node values.
GEOM	Geometric mean of the two nearest grid node values.
HARM	Harmonic mean of the two nearest grid node values. This is the default option.
UPWI	Value at the node located upstream of the interface.
MINI	The smaller of the two nearest grid node values.
MAXI	The larger of the two nearest grid node values.

SATU This modifier is effective only if the **HARMonic** modifier is also present. The mean is computed by a weighted harmonic average of the variable value multiplied by $(S^n)^{1/\alpha}$ where S^n is the phase saturation and α is the inverse of the exponent. This is the default option if the multiphase problem mode is invoked.

α The inverse of the exponent of saturation if the **SATUrated HARMonic** option is invoked; otherwise this input is ignored. The default value is 2.

EXAMPLES

- PROP**erty at interface for all variables by arithmetic average
- PROP**erty in **EFFE**ctive mode; use **GEOM**etric means
- PROP**erty averages for T and C by **UPWI**nd means
- PROP**erty in **EFFE**ctive mode for C by **MAXI**mum method
- PROP**erty in **EFFE**ctive mode for P by **HARMONIC SATURATED** based with power 4.

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

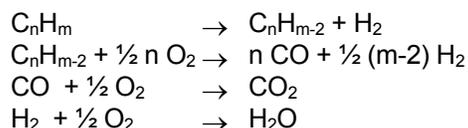
COMMAND REACTION

PURPOSE To select the nature and mode of control of chemical reaction for reactive flows.

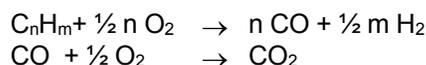
MODE 1: **The Built-In 2 or 4-Step, 8 Species Reaction Mechanism.**

SYNTAX **REAC [TWO] [KINE | PROD | HYDR | OFF]**

TWO By default, the reaction mechanism is composed of four steps:



If this modifier is present, then the reaction mechanism is assumed to be:



See Section 3.6 and comments below for further details.

KINE By default, the reaction rate for the built-in reaction mechanism is taken to be the smaller of the chemical kinetics (Arrhenius) reaction rate and the eddy break up (EBU) rate to account for turbulence. If this modifier is present, then the EBU model is deactivated and the reaction rate is that given by the chemical kinetics. See Section 3.6 and comments below for further details.

PROD This modifier is used if the user selects the explicit reaction mode (Mode 2) but still wants to employ the algebraic relations given in Section 3.6.2 to determine the mass concentrations of O₂, H₂O, CO₂, and N₂ from the solved species and conserved variables.

HYDR The reaction is for a hydrocarbon fuel but the user explicitly specifies the equilibrium relations for those chemical species for which a transport equation is not solved.

OFF The default 4-step reaction is switched off.

COMMENTS

The default reaction system consists of the 4 steps and 8 species described in Section 3.6. The total gas phase consists of 8 chemical species: C_nH_m, C_nH_{m-2}, CO, H₂, O₂, H₂O, CO₂, and N₂. The default fuel is C₁₀H₁₉. This can be changed with the **FUEL** commands. The default Arrhenius reaction constants are given in Table 3.6.1; these can be changed by the **ARR**henius command. The EBU constants can be changed by the **EBU** command.

EXAMPLES

REACTive flow with default options

REACTion with TWO step mechanism default options

REACTion process is KINEtically controlled

REACTion process; use default PRODuct relations.explicit reactions supplied later

REACTion for HYDRocarbon fuel; next commands explicitly supply the reaction details

REACTion built-in system switched OFF. Explicit reaction options of Mode 2 may be specified.

MODE 2: User-Specified Chemical Reaction Rate.

SYNTAX REAC {idreac} { Φ } [EBU] [TEMP] [subrgn] [option] {N1, ..., N5} [Nk]

idreac A unique identifier for the reaction rate. The reaction rate is computed as:

$$\text{Rate} = Z_0 \exp\left(-\frac{E}{R_u T}\right) \Phi_1^\alpha \Phi_2^\beta \Phi_3^\gamma T^\vartheta$$

where Z_0 is the pre-exponential constant, E is the activation energy, R_u is the universal gas constant, (E/R_u is the activation temperature), T is the temperature in absolute units, α , β , γ and ϑ are exponents and Φ 's are the reactant concentrations. **The Z_0 is generally reported in the literature in molar units where the Φ 's are molar concentrations.** The actual computed Φ 's in ACRi Software may be in mass, molar, volumetric or user specified units. Therefore the specified value of Z_0 may need to be converted to appropriate units internally.

Φ **One or more** symbols that specify the chemical species participating in the reaction as reactants. **A maximum of three species can be specified.**

EBU The reaction is limited by the eddy breakup (EBU) rate (see **EBU** command).

TEMP **By default N2 specifies the activation energy, E (e.g. J/kg-mole).** If this modifier is present, then **N2** is the activation temperature (E/R_u) in units of Kelvin.

subrgn The subregion for which the **idreac** reaction is computed. **If no subregion is specified, the reaction is computed for the entire computational domain.** See Sections 7.2.3 and 7.2.4.

option The modifier for interpretation of numerical input

option	INTERPRETATION
MASS	By default it is assumed that the computed Φ 's are in mass units and that N1 is given in molar units. Therefore N1 is internally converted to appropriate mass units. If this modifier is present, then no unit conversion is performed. It is assumed that N1 is specified in appropriate units for the computed Φ 's.
CGS	By default, N1 and N2 are assumed to be in specified SI (kg-mole - m ³ - J - K) units. If this modifier is present, then N1 and N2 are assumed to be in the cgs (gm-mole - cm ³ - cal - K) units and the activation energy in calories. The specified values are converted to SI units on the assumption that the computed Φ 's are mass concentrations.
LIFE	N1 is in units of time [t] representing the half -life of the reaction. In this case N2 must be the activation temperature [T].
DECA	N1 is in units of inverse of time [t ⁻¹] representing the decay rate of the reaction. In this case N2 must be the activation temperature [T].

N1 The pre-exponential constant, Z_0 . The specified value may be internally converted to appropriate units based on the **option** modifier. For example, in the CGS system, units are cm³ⁿ⁻³/molesⁿ⁻¹-s, where $n = N3 + N4 + N5$.

N2 Activation energy, E , (J/kg-mole or cal/gm-mole) or activation temperature, E/R_u , (Kelvin).

N3,N4,N5 Stoichiometric coefficients or exponents, α , β and γ , for the species specified by Φ 's. **There is no default value; one value must be specified for each identified reactant.**

Nk Exponent of the temperature, ϑ , in reaction rate equation; it is the 4th, 5th or 6th numerical value depending on the order of the reaction (1, 2 ,or 3 reactants). **The default value is 0.**

EXAMPLES

REACTION R1 preexp=1.960E+18, activ_e=2.48E+04 FU ^ 0. and O2 ^ 0.5

REACTION R1 FU, O2 1.960E21 2.062E8 0.0 0.5 1.07 CGS EBU TEMP SELEcted subregion

REACTION R2 CH, O2, FU: 5.395E16, 2.50E04 0.0 0.9 1.18 -0.57 TEMP for ID=RGN1 active

REACTION R1 MASS based preexp=1.50, activ_T= 600 FU ^ 1. and O2 ^ 1

REACTION R1 MASS FU, O2 1.50 0. 1 0.5 1.07 EBU ! No effect of temperature

REACTION R2 DECAy frequency FU: 2.00, 1000 1.0 for SELEcted region

REACTION R2 between C1 & C2 Half Life = 0.01 Active_T=0., exponents: 1 & 1 ID=RGN1

MODE 3: Fast-Chemistry Reaction Mechanism**SYNTAX** **REAC** {FAST} {N1, N2} {fname | N3, ..., Nn}

FAST The two-parameter PDF fast-chemistry reaction model is used. In this mode, two differential equations are solved for the mean and variance of the mixture fraction. The values of the mean mixture fraction and variance at each node are then used to determine the properties (temperature, species concentrations, and density) at the node by a lookup table procedure. The lookup table consists of the properties as a function of the mixture fraction mean and variance. The table is constructed by assuming that the mean and variance describe the PDF (usually a beta-function) of the properties at each point of the domain. The properties for each mean and variance value in the table are then determined by convoluting the corresponding PDF with the properties, which are assumed to be a function of the mixture fraction and are determined by chemical equilibrium considerations. This model is based on the approach described by Jones and Whitelaw (1982).

fname The name of the file containing the fast-chemistry lookup table. The total number of values in the table must equal **N1** times **N2**.

N1 The number of rows of data in the fast-chemistry lookup table. Each row represents the properties (temperature and species concentrations) for a given value of the mean and the variance. *There is no default value.*

N2 The number of columns of data in the fast-chemistry lookup table. The first column represents the mean, the second column the variance, and each of the others represents a property. *There is no default value*

N3,...,N5 These represent the numerical values in the table. If the file name is specified, then these values are read from the file; otherwise **N1** times **N2** values must be specified with the command. *There is no default value.*

EXAMPLES

REACTION FAST chemistry; lookup table has 1000 rows and 5 columns on file 'FAST_CHEM'

REACTION FAST chemistry 1000 rows & 5 columns values !!!! 5000 values must be specified)

MODE 4: Empirical Constants for Fast-Chemistry Reaction**SYNTAX** REAC {FAST} {CONS} {N1, N2}**FAST** The two-parameter PDF fast-chemistry reaction model is used.**CONS** The empirical constants required for the fast-chemistry reaction are specified.**N1** The constant C_{g1} (>0) for fast-chemistry reaction (see Jones and Whitelaw, 1982). The default value is 2.**N2** The constant C_{g2} (>0) for fast-chemistry reaction (see Jones and Whitelaw, 1982). The default value is 2.**EXAMPLES**

REACTION FAST chemistry CONStant cg1 = 3.0**REACTION** FAST chemistry CONStants: cg1 = 2.0 and cg2 = 3.0**REACTION** FAST chemistry CONStants: cg1 = 2.0 and cg2 = 3.0

MODE 5: **Disable Previously Specified Reaction**

SYNTAX **REAC** {idreac} [subrgn] {OFF}

idreac A unique identifier for the reaction rate.

subrgn Same as Mode 2.

OFF Any previously specified reaction with **idreac** identifier for the specified subregion is disabled. A new specification may follow.

EXAMPLES

REACTION R1 for subdomain ID=R1DMn switched OFF

REACTION R2 for SELEcted subdomain OFF

COMMAND **READ****PURPOSE** To read the archive file for basic problem information and initial conditions.**MODE 1:** **Read a Generic Archive File****SYNTAX** **READ {fname} [fmt] [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).****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

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 the input command file must be identical to the file that generated the ‘acr_RESTART.TMP’ file except for changes in, or presence of, **READ**, **SAVE** and **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. **If no file name is specified, then the restart input is obtained from the file named ‘acr_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 REST

READ REStart file from a previous run

READ REStart data from file ‘STAGE_1.FILE’

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 | HYDR } [ABSO | FREE | 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.

HYDR The “hydrostatic” mode of ANSWER™ is invoked. By default the body force in the momentum equations is computed as (ρg_j) where g_j is the body force vector component in the x_j direction. If this modifier is present, then the body force is computed as $(\rho - \rho^*) g_j$. This option is identical to the PROBLEM command with ATMOSPHERIC option. Please see the PROBLEM command for further details.

ABSO The absolute base for the temperature scale, T_a , is specified. This modifier is effective only if the modifier T or TEMP is also simultaneously present. The default value is 273.15.

FREE 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

- REFE P is 2.5E5 !!! N/m^2
- REFE T = 30. deg C
- REFE T = 60. F
- REFE T ABSOLute base = 459.67 R
- REFE ABSOLute TEMPerature base = 459.67 R
- REFE FREEZing TEMPerature is = 32 R
- REFE BOILing TEMPerature is = 212 R
- REFE HYDRostatic component added to pressure field

MODE 2: Specify Constants of Options for Pressure Related Choices

SYNTAX REFE { P } [NODE | COEF | NORM] { N1..Nn }

P The specified input pertains to the pressure equation.

NODE In the ANSWER™ software, the pressure for incompressible flow is computed with reference to a datum. By default the pressure is set to zero at the first inlet node or the first open node in the computational domain. If this modifier is present then the pressure is set to zero at the grid node specified by N1 through Nn. For the compressible flow mode of the ANSWER™ software, and for the other ACRi software packages (PORFLOW™, TIDAL™, etc.), the pressure is in absolute mode and this information is used only for computation of the pressure coefficient, if required (see below).

COEF The location of the reference value for computation of the pressure coefficient is specified. The pressure coefficient, C_p , is defined as:

$$C_p = \frac{P - P_{ref}}{0.5 \rho V^2}$$

In the equation above, P is the computed pressure, P_{ref} is the pressure at the specified location, ρ is the density and V is the fluid speed. By default the location of the reference pressure is assumed to be the same as that of the datum (see NODE).

NORM The normalizing pressure is specified. By default the normalizing pressure is unity except for PORFLOW™ it is the pressure head ($\rho^* g$) 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

COMMAND **REGENERATION**

PURPOSE To specify regeneration rate of one species from another in the decay chain.

SYNTAX **REGE** { Φ } { Ψ } [**N1**]

Φ The symbol for the species that is generated by the decay of another species (denoted by Ψ) in the decay chain. The valid symbols are listed in Table 6.7.1.

Ψ The symbol for the species that is decaying to generate the species denoted by Φ . The valid symbols are listed in Table 6.7.1.

N1 The factor to convert the decay rate of Ψ to the source for the decay product Φ . For species in mass units this number is between 0 and 1. However if the species are in user-defined units (such as curies) then this number may take on any value. **The user must ensure that the value supplied is consistent with the defined units of Φ and Ψ . If no value is specified, then the factor is assumed to be equal to 1.**

EXAMPLES

REGEneration of C2 from decay of species C1 ; converted fraction is 1.0 (total conversion)

REGEneration of C2 from C1; conversion factor = 1000 !! (C1 in kg but C2 in g units)

REGEneration of C2 from C1; conversion factor = 60 !! to account for different units of C1 and C2

COMMAND RELAX

PURPOSE To specify relaxation factors for iterative solution of the matrix of equations.

SYNTAX **RELA** { [$\Phi=N1, \Phi=N2, \dots, \Phi=Nn$] | [OFF] }

Φ One or more of the symbols for which the relaxation parameters are specified. The valid symbols are those listed in Table 6.7.1 plus **RHO**, **VIS** and **PP**.

OFF All relaxation parameters are set to unity. Any other input in the command is ignored. In essence, the relaxation feature is disabled.

N1, ..., Nn Relaxation factor (Θ of Equation 4.2.2) for the variable denoted by the symbol immediately preceding the value. The numerical value must be larger than 0 and less than 2.

COMMENTS

This command is effective in both steady state and transient modes of solution. The relaxation factor affects the convergence of the numerical solution. If the solution shows instability, a relaxation factor with a value less than unity may help obtain a stable solution. Alternatively, if the convergence rate is too slow, a value greater than unity may result in more rapid convergence. A value larger than 2 will lead to exponentially unstable growth of the solution.

In the steady state modes the relaxation factor affects the values of the variable during outer-loop step-to-step iterations. In the transient mode, a relaxation factor is used during execution of the inner loop in which multiple iterations (**N2** on **CONV**ergence command) are used at each time step. A more complete discussion of the role of the relaxation parameter is given in standard textbooks (for example, Varga, 1962).

EXAMPLES

RELAxation factor for P = 0.7

RELAxation factors: T = 1.2, C = 0.9

RELAxation factors: T = 0.7, C = 0.9, C2 = 0.7

RELAxation factors: P=0.1, P2=0.1, T=0.1, S = 0.5

RELAxation OFF

COMMAND **RENAME**

PURPOSE To rename the output variables listed in Table 6.8.1.

SYNTAX **RENA** { Φ = name}

Φ The current symbol for the output variable that is to be renamed.

name A character string that defines the new symbol and name for the variable currently denoted by the symbol, Φ . The name must be enclosed in single or double quotes and must be a character string no longer than 64 characters. The first four characters of this name will be used as the new symbol for all subsequent references to this renamed variable.

EXAMPLES

```
// Rename "U" to "XDIR Velocity of Fluid Flow" for all future reference
RENAme U = 'XDIR Velocity of fluid flow'
// Rename "T" to "TEMPERATURE (THERMODYNAMIC)"
RENAme T = 'TEMPERATURE (THERMODYNAMIC)'
// Rename "FF" to "TC99 Technetium 99 in solution"
RENAme FF = "TC99 Technetium 99 in solution"
```

COMMAND **RETARDATION**

PURPOSE To specify the retardation coefficient for the governing differential equations.

COMMENTS_____

This command is identical to the **DIST**ribution command except that, for the transport equations, it specifies the retardation, rather than the distribution, coefficient. Mode 2 of the **DIST**ribution command **can not be used with the RET**ardation command. For the pressure and the temperature variables, this command is identical to the **DIST**ribution command.

EXAMPLES_____

See the **DIST**ribution command

COMMAND **ROCK**

PURPOSE To specify the rock type and its physical properties.

COMMENTS _____

This command is identical to the **MATE**rial command in all respects.

EXAMPLES _____

See the **MATE**rial command

COMMAND	SAVE
PURPOSE	To write the archival and post-processing file for restart, plotting and archiving purposes.
MODE 1:	Archival and Post-Processing Data by Field Format
SYNTAX	SAVE [Φ] [fname] [fmt] [GEOM COMP] [DATA] [REPL] [V _{freq}] [TIME] [NOW OFF ON]
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. If no symbols are specified then a suitable set is automatically selected based on the nature of the problem being solved.
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 'acr_SAVE.TMP' unless a name was specified on a previous SAVE command.
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 Φ above). If GEOMetry modifier is present, then only the problem geometry and grid connectivity information is written to the file; the Φ variables are omitted.
COMP	If the COMPact modifier is present, then the problem geometry and grid connectivity information is not written to the file; only the Φ variables are written.
DATA	This modifier is effective only if the COMPact modifier is also present. By default a standard ACRi file header containing information about the problem and nature of data appears at the start of the file. If the DATA modifier is present then the file header information is omitted; only the data fields are written to the file.
REPL	If multiple records are written, then by default the new record is appended to the existing file. If this modifier is present, then the old records are replaced by the new record.
V _{freq}	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, the V _{freq} is interpreted to be the frequency in terms of number of steps. If this modifier is present, then V _{freq} is interpreted to be time interval between successive outputs.
NOW	Archive output is produced immediately
OFF	Archive output is completely suppressed.
ON	Archive output is reactivated if it was previously suppressed.

COMMENTS

If this command is omitted or if V_{freq} is equal to 0, output for the active variables is automatically produced at the end of simulations. The active variables consist of all variables for which the equations are solved, plus other derived variables that are of interest depending on the nature of the problem being solved. Successive commands may be used to accommodate changing output requirements.

EXAMPLES

SAVE every 20 steps

SAVE U, V, W on file 'DEMO.PLT' in UNFormatted mode

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 only GEOMetry data on the file 'GEOM.SAV'

SAVE in COMPact form on file 'VARIABLES.SAV'

SAVE in COMPact DATA form on file 'DATAONLY.SAV'

SAVE OFF for all variables

MODE 2: Archival and Post-Processing Data in Tabular Format**SYNTAX** **SAVE** [Φ] {**TABL**} [**subrgn**] [**fname**] [**REPL**] [**V_{frq}**] [**TIME**] [**NOW** | **OFF** | **ON**] Φ See Mode 1 Specification.**TABL** Specifies that the data file is to be written in a tabular format. See Section 6.9 for further details.**subrgn** The subregion for which the output is required. 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 to which the output is directed. See Section 7.2.2 for additional information. The default file name is 'acr_SAVE_TABLE.TMP' unless a name was specified on a previous **SAVE** command with **TABL**e modifier.**REPL** See Mode 1 Specification.**V_{frq}** See Mode 1 Specification.**TIME** See Mode 1 Specification.**NOW** See Mode 1 Specification.**OFF** See Mode 1 Specification.**ON** See Mode 1 Specification.**EXAMPLES****SAVE** **TABL**e of U, V, W on file 'TABLE.ARC'**SAVE** U, V, P, K and L in **TABL**e format every 100 steps**SAVE** in **TABL**e format at **TIME** interval of 1.25 years**SAVE** in **TABL**e format at **TIME** interval of 1.25 years and **REPL**ace every time**SAVE** **TABL**e of U, V, W on file 'DEMO.TBL'**SAVE** **TABL**e of U, V, P for subregion on most recent **LOC**ate command**SAVE** U in **TABL**e for subregion **ID=DMN**Table at **TIME** interval = 0.2 hours**SAVE** W in **TABL**e format **NOW****SAVE** **TABL**e of T **NOW** at **TIME** interval of 20 years**SAVE** **OFF** for **TABL**e option**SAVE** **ON** for **TABL**e option

MODE 3: Output of Properties and Coefficients for the Differential Equations

SYNTAX **SAVE** { ϕ } {DIFF | STOR | SOUR | MATR | RESI} [*fname*] [REPL] [*V_{freq}*] [TIME] [NOW]

- ϕ One or more of the symbols that denote the dependent variable(s) 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.
- DIFF** The diffusion coefficient for the variable is written to the archive file. If the diffusion coefficient is a tensor, then the output is produced for the diagonal component in each direction.
- STOR** The storage coefficient for the variable is written to the archive file.
- SOUR** The net source (algebraic sum of source and sink) for the variable is written to the archive file.
- MATR** The matrix coefficients for the variable are written to the archive file. The coefficients are written 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.
- fname* The name of the archive file. See Mode 1 Specification.
- REPL** See Mode 1 Specification.
- V_{freq}* See Mode 1 Specification.
- TIME** See Mode 1 Specification.
- NOW** The output is produced at the next step following the command.

COMMENTS

Each command may specify only one type of output. However, multiple commands may be used in succession to obtain different types of output.

EXAMPLES

- SAVE:** for variables U and T: DIFFusion coefficients at the final stage
SAVE: for T:, U , V SOURce coefficients NOW (at the next step)
SAVE: for T:STORage coefficients at step number 52
SAVE: for U and T the: MATRix at step number 52 on file 'MATRIX.T'
SAVE: RESIdue, for T and P at step number 75
SAVE: RESIdue for U, T and P at the end

MODE 4: Output of Balance of Convective and Diffusive Flux for each Element

SYNTAX **SAVE** { ϕ } {FLUX} [CONV | DIFF] [AREA] [fname] [REPL] [V_{frq}] [TIME] [NOW]

ϕ	One or more of the symbols that denote the dependent variable(s) 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 written to the archive file. 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 written to the archive file.
DIFF	The diffusive flux for each face of the elements of the computational domain is written to the archive file.
AREA	The computed flux is divided by the projected area of the face for output.
fname	The name of the archive file. See Mode 1 Specification.
REPL	See Mode 1 Specification.
V_{frq}	See Mode 1 Specification.
TIME	See Mode 1 Specification.
NOW	The output is produced at the next step following the command.

COMMENTS

Each command may specify only one type of output; however multiple commands may be specified to obtain output of different types.

EXAMPLES

SAVE: FLUX for U, V, T for each element at the final stage

SAVE CONVective FLUX for T NOW (at the next step)

SAVE: DIFFusive FLUX for T at step number 52

SAVE: DIFFusive FLUX divided by AREA for T at step number 52

MODE 5: Restart File for Later Continuation of Simulations**SYNTAX** SAVE { REST } [fname] [V_{freq}] [TIME]

REST A special file is generated at the end of the simulations 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 only be read 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. **RESTART** 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 Sections 6.4 and 7.2.2 for more information. If no file name is specified, then the output is directed to the file named ‘acr_RESTART.TMP’.

V_{freq} 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. The data on the file is replaced each time the file is written.

TIME By default, the V_{freq} is interpreted to be the frequency in terms of number of steps. If this modifier is present, then V_{freq} is interpreted to be time interval between successive outputs.

EXAMPLES**SAVE** REST**SAVE** REStart file for later use**SAVE** REStart file as ‘STAGE_1.FILE’

COMMAND **SCALE****PURPOSE** To allow internal scaling of the specified input according to the equation:

$$Q_{in} = a_1 * Q + a_2 ,$$

where Q_{in} is the internal representation of a quantity, Q is the value specified by the user, and a_1 and a_2 are user-specified constants.

SYNTAX **SCAL** { N1 } [N2]**N1** Multiplier, a_1 , in the scaling equation; automatically set to 1 at the end of each application.
The default value is 1.**N2** Addend, a_2 , in the scaling equation; automatically set to 0 at the end of each application.
The default value is 0.**COMMENTS**

This command must be used immediately before the command that contains the data to be scaled. It can be used for internal scaling in conjunction with the **COORD**inate, **SET** and **SOUR**ce 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 **SCRE** [OFF | ON] [WIDE]

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

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

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

COMMENTS

This command allows the user to monitor progress of the numerical simulation on the CRT device. It does not affect the diagnostic output to the standard output device.

EXAMPLES

SCREen echo for diagnostic output to be turned OFF

SCREen ON ! This is the default option

SCREen in WIDE format

SCREen ON in WIDE format ! Turn on previously off screen output

COMMAND **SELECT**

PURPOSE To locate and identify a subregion or material type in the domain of computation for later input by one of the other commands.

COMMENTS _____

This command is identical in all respects to the **LOCAte** command.

EXAMPLES _____

See **LOCAte** command

COMMAND **SET**

PURPOSE To set the value of a field variable as a function of space, time or other variables.

MODE 1: **Specification of Field Variable**

SYNTAX **SET** { Φ } [func [ξ]] [option] [mod] [ALWA] [subrgn] [FIEL] [dir] {N1} [fname | N2 ..., Nn] [STAC]

Φ **One, and only one**, symbol that denotes the variable for which the values are specified. The valid symbols include those listed in Table 6.8.1 plus the user defined variables (**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 7.2.4, which denotes the functional form of the dependent variable. *If no function is specified, the value is assumed to be constant.*

ξ One of the independent variables listed in Table 7.2.5. *If no variable is specified, the independent variable is assumed to be time.*

option Option selected for the operation to be performed.

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

mod The modifier for function evaluation

mod	INTERPRETATION
ABS or ABSO	The absolute value for the computed function is taken.
POSI	The negative values for the computed function are set of zero
NEGA	The positive values for the computed function are set of zero

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

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

FIEL Unless the **subrgn** was specified by a **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. 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 Φ is computed from the value of ξ at the same location. If this modifier is present, then Φ is computed from ξ at a location previously specified by the **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.

MODE 2: Specification of a Variable as a Linear Sum of other Variables

SYNTAX **SET** { Φ } {SUM | LINE } [MASS] { ξ_1, \dots, ξ_n } [a_1, \dots, a_n] [a_0] [option] [mod] [ALWA] [subrgn] [FIEL]

Φ See Mode 1 specification.

SUM The variable Φ is computed from:

$$\Phi = \sum_n \xi_n$$

LINE The variable Φ is computed from:

$$\Phi = \sum_n a_n \xi_n + a_0$$

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

ξ_1, \dots, ξ_n Symbols to denote the variables, or their masses, which contribute to the linear sum as in the above equation. The valid symbols include those listed in Table 6.8.1 plus the variables defined by the user (**ALLOCATE** command) or the real type variables defined in the initialization file (xxxINIT.ACR). There is no default value; appropriate symbols must be specified. No more than 10 symbols can be specified.

a_1, \dots, a_n The numerical constants and coefficients for the linear function. These values must be specified if the **LINEAR** modifier is present. With the **SUM** modifier these values must not be specified. There is 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 ID=T1DOMAIN

SET T as LINEAR function 1. * X + 2. * Y -0.5 * U + 5. ALWAYS for ID=T1DOMAIN

MODE 3: Specification of a Variable as a Square or Square Root Sum of other Variables

SYNTAX **SET** { Φ } {SQUA | ROOT} { ξ_1, \dots, ξ_n } [a₁, ..., a_n] [a₀] [option] [mod] [ALWA] [subrgn] [FIEL]

Φ See Mode 1 specification.

SQUA The variable Φ is computed from:

$$\Phi = \sum_n \xi_n^2 \quad \text{or} \quad \Phi = \sum_n a_n \xi_n^2 + a_0$$

ROOT The variable Φ is computed from:

$$\Phi = \sqrt{\sum_n \xi_n^2} \quad \text{or} \quad \Phi = \sqrt{\max\left[\left(\sum_n a_n \xi_n^2 + a_0\right), 0\right]}$$

ξ_1, \dots, ξ_n Symbols to denote the variables that contribute to the linear sum as in the above equation. The valid symbols include those listed in Table 6.8.1 plus the variables defined by the user (**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, ..., a_n The numerical constants and coefficients for the function. Either all coefficients must be omitted (in which case the function without coefficients is used) or all must be specified.

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 4: Specification of Multiple Variables on a Node-by-Node Basis from a Table

SYNTAX SET {NODE} { Φ_1, \dots, Φ_n } { V_1, \dots, V_{nm} | fname} [subrgn] [FIEL] [dir]

NODE The Φ_n variables are set individually for each of the nodes of a subregion. It is assumed that the input is in a tabular format with n columns (1 for each variable) and m rows (1 for each node of the subregion).

Φ_1, \dots, Φ_n Symbols to denote the variables that are to be set by the command. The symbols include all those listed in Table 6.8.1 plus the user defined variables (**ALLOCATE** command) and the real type variables defined in the initialization file. **In addition a special symbol "SKIP" can be specified to by-pass the input of any of the columns. Any symbol may occur multiple times. There is no default value; appropriate symbols must be specified.**

V_1, \dots, V_{nm} The table of values that pertains to the variable values at each node selected by the **subrgn** and **dir** modifiers. If n symbols are specified on the command and there are m nodes in the subregion, then n times m values must be specified in n columns and m rows.

fname The name of the file (see Section 7.2.2) from which V_1 through V_{nm} are read.

subrgn See Mode 1 specification.

dir See Mode 1 specification.

FIEL See Mode 1 specification.

EXAMPLES

SET by NODE following variables for ID=SUBRGN1 ! There are 3 nodes in this region)

U	V	T	
10	-5	100	
-10	+5	150	
-10	+10	200	

SET by NODE following variables for ID=SUBRGN1 ! There are 3 nodes in this region)

SKIP	SKIP	U	V	SKIP	T
0.2	0.2	10	-5	10.20	100
3.5	0.6	-10	+5	12.50	150
10	1.0	-10	+10	2.00	200

SET by NODE for ID=MIDDLE at X- boundary ! 3 boundary nodes

(SKIP, SKIP, SKIP)	U	V	T
(0.2, 0.2, 0.2)	10	-5	100
(3.5, 0.6, 1.0)	-10	+5	150
(10., 1.0, 2.0)	-10	+10	200

SET by NODE ID=MIDDLE T, U, V SKIP from file 'SPATIAL' ! File with 4 x #of nodes values

MODE 5: Specification of Variables as Distance-based Interpolated Function from a Table

SYNTAX **SET** {DIST} [SQUA | LINE] {Φ₁, ..., Φ_n} {Nsets} {V₁...V_m | fname} [option] [mod] [ALWA] [subrgn] [FIEL]

DIST The Φ_n variables are computed from a inverse distance based interpolation function (here ξ_i is ith value in the specified table for the corresponding variable and x_{k,i} are its coordinates):

$$\Phi = \sum_i \frac{\xi_i / r_i}{1/r_i} \quad \text{or} \quad \Phi = \sum_i \frac{\xi_i / r_i^2}{1/r_i^2} ; \quad r_i^2 = \sum_k (x_k - x_{k,i})^2$$

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

LINE The inverse distance based interpolation is used.

Φ₁, ..., Φ_n Symbols to denote the variables that are to be computed from the distance-based interpolation. **There is no default value; appropriate symbols must be specified. No more than 7 symbols can be specified.**

Nsets The number of sets of values in the table. Each set must consist of (x, y; for 2D) or (x, y, z; for 3D) and as many values as the symbols specified above. **See also below. There is no default value.**

V₁...V_m The table of values that pertains to the coordinates of the locations and the variable values at those locations. If the table is viewed to consist of a number of columns and rows, then each row must contain, in order, (x, y; for 2D) or (x, y, z; for 3D) and **n** values for Φ₁, through Φ_n. The number of rows of the table must equal **Nsets**.

fname The name of the file (see Section 7.2.2) from which V₁ through V_n are read.

option See Mode 1 specification.

mod See Mode 1 specification.

ALWA See Mode 1 specification.

subrgn See Mode 1 specification.

FIEL See Mode 1 specification.

EXAMPLES

SET by DISTANCE interpolation from specified values at 3 stations (Two-dimensional)

(X, Y)	U	V	T
(0.2, 0.2)	10	-5	100
(3.5, 0.6)	-10	+5	150
(10, 1.0)	-10	+10	200

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

(X, Y, Z)	U	V	T
(0.2, 0.2, 0.2)	10	-5	100
(3.5, 0.6, 1.0)	-10	+5	150
(10., 1.0, 2.0)	-10	+10	200

SET by DISTANCE ID=MIDDLE 3 sets of U V T file 'SPATIAL' ALWAYS ADD POSITIVE to existing

MODE 6: Specification of Values for a Variable from a File**SYNTAX** **SET** { ϕ } {subrgn} [FIEL] {fname}

ϕ A symbol to denote the variable for which the field values are specified. The valid symbols include those listed in Table 6.8.1 plus the supplementary integer or real type variables that are defined in the initialization file (xxxINIT.ACR). There is no default value; a symbol must 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.

fname The name of a file from which numerical values are read. See Section 7.2.2 for additional information.

FIEL See Mode 1 specification.

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 7: **Specification of Values from Total Inventory**

SYNTAX **SET** { ϕ } {INVE} [VOLUME | UNIF | SCAL] {subrgn} [FIEL] {N1}

ϕ A symbol that denotes the variable for which the values are specified. This input mode can only be used for those variables for which a differential equation is solved. The valid symbols include those listed in Table 6.7.1 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 of the property for the subregion is specified.

VOLUME The inventory is distributed so that the amount in each element is proportional to the volume of the element. This is the default option.

UNIF The inventory is distributed so that ϕ is uniform and constant.

SCAL The existing value of ϕ is scaled such that the total amount is equal to the specified inventory.

subrgn See Mode 1 specification.

FIEL See Mode 1 specification.

N1 The total inventory of the property ϕ which is defined as:

$$N1 = \sum_{\text{subrgn}} \alpha \phi \delta V ,$$

where α is the accumulation coefficient for the transport equation, δV is the volume of an element and the summation is taken over all elements of the subregion.

COMMENTS

This command can be used to set the values only in the interior of the computational domain. Also, the specification for **P** is meaningful only for variable density flows; in this case, the density of the fluid is set so that the total amount of fluid is equal to the specified inventory.

EXAMPLES

- SET** INVENTORY of FU to 10 kg
- SET** INVENTORY of FF to 10 kg; distribute UNIFormly over SELEcted region
- SET** T INVENTORY in ID=RGN1 is 1.E6 units; SCALe existing value

MODE 8: Computation of Gradient of a Variable

SYNTAX SET {GRAD} {Ψ} {Φ} {ξ} [fnc] [BLOC] [FIEL] [ALWA | OFF]

GRAD The Gradient of the specified variable is computed as:

$$\Psi = \frac{\partial\Phi}{\partial\xi} \quad \text{or} \quad \Psi = \text{function}\left(\frac{\partial\Phi}{\partial\xi}\right)$$

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

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

Φ Symbol for one of the previously defined field variables listed in Table 7.2.5. There is no default value; a variable must be specified.

ξ One of the symbols: X, Y, r or Z that denotes the coordinate with respect to which the gradient is computed. The symbol r can only be used if the specified coordinate system is cylindrical. There is no default value; a value must be specified.

fnc The modifier for function evaluation

fnc	INTERPRETATION
ABS or ABSO	The absolute value for the computed gradient is taken.
POSI	The negative values for the computed gradient are set of zero
NEGA	The positive values for the computed gradient are set of zero
SQUA	The function is computed as the square of the gradient
ROOT	The function is computed as the square root of the gradient; negative values are set to 0.

BLOC If this modifier is present then the value of the gradient at the blocked nodes (see **BLOCK** command) is set to zero. Otherwise the value is computed from the existing variable values in the neighborhood.

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

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

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

EXAMPLES

- SET DTDX = GRAD of T with respect to X
- SET DTDR = ABSOLUTE value of GRAD of T with respect to R compute ALWAYS
- SET PHIR = GRAD of T with respect to R only once
- SET DTDY = GRAD of T with respect to Y for FIELd nodes only with BLOC value set to zero
- SET DTDY = POSItive values of GRAD of T with Y with BLOC = 0
- SET DTDY = SQUAre ROOT of GRAD of T with Y with BLOC = 0

SET DTDY OFF

MODE 10: Disable Previously Specified SET Commands

SYNTAX **SET** { Φ } {OFF} [subrgn]

Φ See Mode 1 specification.

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

subrgn Identifier for the subregion that appeared previously in a **SET** command for the current variable, Φ .

EXAMPLES

SET commands for T for OFF for currently SELEcted subregion

SET commands OFF for T for ID=MIDDLE

COMMAND	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
SYNTAX	SOLV [Φ] [MANU AUTO] [PASS] [N1, N2, N3, N4, N5, N6, N7]
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.
AUTO	By default the time step is determined according to the user-specified input of N2 through N4 . If this modifier is present, then the time step is automatically determined within user-specified constraints (N2 through N7). The time step is increased if the number of iterations to converge is less than 1/2 of the maximum number of iterations specified on the CONVergence command with FLOW modifier and decreased if it is larger than 4/5 th of the maximum number of iterations.
PASS	With the AUTO modifier, the time step may be adjusted either dynamically or passively. This modifier disables the dynamic mode; only passive adjustment takes place. In the dynamic adjustment mode, if the solution fails to converge to a specified tolerance in the specified number of iterations (see above), the time step is decreased and the solution process is repeated from the previous time. In the passive adjustment mode, the time step is adjusted only from the next time step; the currently computed values are retained.
N1	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 in the transient mode. Each successive time step will be multiplied by this value until a maximum value specified by N4 is reached. The default value is 1.01 if the AUTO modifier is present; otherwise, it is 1.
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 using the AUTO modifier. 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. This input is useful in conjunction with the AUTO modifier to limit the amount of computer resources allowed. The default value is 999999.

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 50 years in steps of 2

SOLVE for 50 hours, initial step=0.2 hr, increase by 1.1, max=10 hr

SOLVE 1.E6 yrs, DT=1, fac=1.1, max=1000, min=1, dfac=1.1, max steps=1000

SOLVE for 10 days 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 10 days, **MANUal** mode from now on

SOLVE for T only for 10 days in steps of 0.2

SOLVE for T only for 10 days in **AUTO** mode

SOLVE for P2, T and C3 only for 10 days in steps of 0.2

MODE 2: **Steady State Solution Mode**

SYNTAX **SOLV {STEA} [Φ] [N1, N2]**

STEA The equations are solved in their steady state mode. That is, the storage or accumulation term in the governing transport equation is set to zero.

Φ By default, an appropriate set of governing equations is solved based on the user-specified input. However, the user may override this default by explicitly specifying one or more symbols to select the corresponding dependent variable for which the governing equations are solved. The valid symbols are listed in Table 6.7.1.

N1 The maximum number of iterative steps for solution of the matrix. The default value in this instance is 999999.

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 3: **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 4: **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} [TOTAL] [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.

TOTAL 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 7.2.4 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 7.2.5. If no independent variable is specified, the variable is assumed to be time.

option Options selected for implementation of the source.

option	INTERPRETATION
VOLU	In the absence of the TOTAL modifier, the source for each element is computed as: $Q = q \delta V$. Here q is the amount specified by the user and δV is the volume of the element. The q , in turn, is computed from $func(\xi)$ 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.
AREA	In the absence of the TOTAL modifier, the source for each element is computed as: $Q = q \delta A$, where δA is the area of the element boundary indicated by dir. 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.
INTE	By default, if dir points to a boundary wall, then any special treatment for the wall is deactivated and the diffusive flux at the wall is set to zero. If the INTERNAL modifier is present then the wall treatment and wall diffusive flux are retained.
NORM	In the absence of the TOTAL modifier, the source, Q , is computed as: $Q = q \sum_i A_i \cdot V_i$ where A_i is the i^{th} direction component of the element boundary area specified by 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, ρ_B , on the command. If this value is omitted, then the boundary value at the node indicated by the dir direction is used if the AREA or 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, V_i , if the NORMAL modifier is present. Two values must be specified for 2D and 3 for 3D flows. There are no default values for this input.
ρ_B	The density value that multiplies the computed source. It can only be specified if the DENSITY modifier is present and then it must be the last value on the command. If no value is specified but the modifier DENSITY is present, then default value is the boundary value at the node indicated by the dir direction if the AREA or NORMAL modifier is present, otherwise the default value is the local density for the element.

EXAMPLES

Generic examples for this command are given in Section 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] [ρ_B]

- FLOW** The source (or sink) for a variable is created due to direct fluid injection or withdrawal. The amount of the property of interest (momentum, heat, chemical species etc.) of the injected (or withdrawn) fluid acts as the source or sink for each of the relevant properties.
- TOTA** See Mode 1 specification
- func** See Mode 1 specification.
- option** See Mode 1 specification.
- subrgn** See Mode 1 specification.
- dir** See Mode 1 specification.
- fname** See Mode 1 specification.
- φ** The symbols that denote the dependent variables injected with the fluid. If no dependent variable is specified, all properties in the injected fluid are assumed to be zero. This specification is ignored if the fluid is being withdrawn because the amount of property withdrawn is equal to the local value of that property.
- N1, .., Nn** See Mode 1 specification.
- Nn+1, .. Nm,** Each of these represents the value of injected variable denoted by the symbol immediately preceding the value.
- Nm+1, .., Nk** The normalizing vector components, V_i , if the **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_B** 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.

EXAMPLES

All examples cited for Mode 1 are applicable provided that the modifier **FLOW** is added. Some illustrative examples of the use of attributes specific to this mode are given below.

SOURCE with **FLOW** injection: amount = 0.001 with T=50, C=1.

SOURCE with **FLOW** per unit **AREA** of X- face: = 0.001 with T=50, C=1.

SOURCE with **FLOW** withdrawal: amount =-0.001 per second

SOURCE FLOW: TABLE 3 sets SELEcted (0, 0), (100, 1), (200, 0) U=1, V=0.1, W=0., T=10, K=0.001, L=1

SOURCE for **FLOW**: EXPONential series with TIME 7 sets from 'SOURCE' T=100, C=0.

SOURCE FLOW q=10 X- direction T=100, U=20. NORMAlized velocity 0., 1.5, 2.5 ID=VSOURCE;

SOURCE FLOW q= -10 in X- dir with NORMAlized vel 1., 1.5, 2.5 and DENSity for ID=VSOURCE

SOURCE FLOW q=10 X- dir variable values: U=10, V=0, W=-20, T=100, K=0.03, L=0.5 INTErnal with NORMAlized vel 1., 1.5, 2.5 and DENSity for ID=VSOURCE:

SOURCE FLOW q=10 per unit **VOLUME** injected variables: U=10, V=0, W=-20, T=100, K=0.03, L=0.5 multiply by DENSity = 5 for SELEcted region

MODE 3: Flow Injection with Fixed Variable Values and Computed Momentum Components

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

MOME The source (or sink) for a variable is created due to direct fluid injection or withdrawal. The amount of the property of interest of the injected (or withdrawn) fluid acts as the source or sink for each of the relevant properties. In addition, the velocity components of the injected flow are computed from the source flow rate for the element, Q, as:

$$V_j = \frac{Q}{\rho_B A} n_j,$$

where V_j are the computed velocity components of the injected flow in the j^{th} direction, ρ_B is the density, A is the area of the element boundary specified by the **dir** modifier, and n_j is a normalizing vector. If the **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 NORMAL modifier is present, it is assumed that the AREA modifier is in effect.** Any specification of the **VOLUME** modifier is ignored. All other modifiers can be used as for Mode 1 Specification.
- 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 **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_B** The density value to compute the velocity components. It can only be specified if the **DENSITY** modifier is present and then it must be the last value on the command. **If no value is specified, then boundary value at the node indicated by the dir is used.**

COMMENTS

The **SOURCE MOMENTUM** command is similar to the **SOURCE FLOW** command except that the momentum (or velocity) of the incoming flow is computed on the assumption that the direction of injected mass is normal to the boundary and that the mass is uniformly distributed at the boundary. All other details of its implementation are identical to those for the **SOURCE FLOW** command.

EXAMPLES

All the examples cited for Mode 2, except those with **VOLUME** modifier, are applicable provided that the modifier **MOMEntum** is added. Some illustrative examples specific to this mode are given below.

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

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

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

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

SOURce MOMEntum: TABLE 2 sets SELEct (0, 0), (100, 1) T=10, K=0.001, L=1

SOURce MOMEntum EXPOntial SERles TIME 7 sets 'SOURCE' T=100, C=0. NORMal 1. -1 SELEcte

SOURce MOMEntum q=10 X- dir INTERnal for ID=VSOURce: injected variables: T=100, K=0.03, L=0.5
NORMalized vel 1., 1.5, -0.7 and DENSity as exists

MODE 4: Flow Injection at Fixed Spherical Angles with Computed Momentum Components

SYNTAX **SOUR** {**ANGL**} [**func**[ξ]] [**TOTA**] [**subrgn**] {**dir**} {**N1**}[**fname**|**N2**,...,**Nn**] [**φ**=**Nn+1**,...,**φ**=**Nm**] [**Nm+1**, ..., **Nm+5**] [**DENS** | **SPEE**] [**ρ_B**| **V_s**]

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

$$U_i = \frac{Q}{\rho_B A} n_i \quad \text{or} \quad U_i = V_s n_i$$

where ρ_B is the density at the neighboring element and A is the area of the element boundary specified by the **dir** modifier. The local direction vector, n_i , for each segment of the surface is obtained from the user input of two spherical angles and a reference axis.

- func** See Mode 1 specification.
- TOTA** See Mode 1 specification
- subrgn** See Mode 1 specification.
- dir** See Mode 1 specification. **This modifier must be present for this mode of command.**
- fname** See Mode 1 specification.
- φ** See Mode 1 specification.
- N1, ..., Nn** See Mode 1 specification.
- Nn+1, .. Nm,** These values represent the value of injected variable denoted by the symbol immediately preceding the value.
- Nm+1** The spherical angle, Θ in degrees measured as the deflection of the injection vector from the face normal. The magnitude of the angle must be less than 180 degrees.
- Nm+2** The spherical angle, ϕ in degrees measured as the rotation of the injection vector with respect to the projection of the axis vector onto the face plane.
- Nm+3, Nm+5** The direction cosines of the axis vector used to measure the angle ϕ ; 3 values must be specified since this option is only available for 3D flows.
- DENS** The density, ρ_B , is specified as the last value on the command.
- ρ_B** See Mode 3 specification.
- SPEE** The injection speed, V_s , is specified as the last value on the command.
- V_s** The value V_s if the **SPEED** modifier is present. **There is no default value for this input.**

EXAMPLES

All examples cited for Mode 1 are applicable provided that the modifier **ANGLE** is added and appropriate input for spherical angles is appended. Some illustrative examples of the use of attributes specific to this mode are given below.

SOURce q=10., X- direction with ID=VSOURce; T=100, K=0.05, L=0.02, Theta=45, phi=20, axis=1., 0, 0
SOURce q=10., X- ID=VSOURce; T=100, K=0.05, L=0.02, Theta=45, phi=20, axis=1., 0, 0, SPEEd=120

SOURce q=10., X- ID=VSOURce; T=100, K=0.05, L=0.02, Theta=45, phi=20, axis=1., 0. 0, DENsity=2.5

MODE 5: Flow Injection with Multiple Tabular Functions

SYNTAX **SOUR** {FLOW} { Φ } {TABL} {MULT} [option] [subrgn] [fname] [N1, ..., Nn]

- FLOW** The source (or sink) for a variable is created due to direct fluid injection or withdrawal. The amount of the property of interest (momentum, heat, chemical species etc.) of the injected (or withdrawn) fluid acts as the source or sink for each of the relevant properties.
- Φ** **One or more** of the symbols that denote the dependent variables which are specified as functions of time. **There is no default value; at least one symbol must be specified.**
- TABL** The variables are specified as tabular functions of time. **This is the only available option in this mode.**
- MULT** Multiple variables are tabular functions of time.
- option** See Mode 1 specification.
- subrgn** See Mode 1 specification.
- fname** See Mode 1 specification.
- N1** The number of sets of data for the tabular functions.
- N2, ..Nn** The N1 sets of data for the flow rate and variables as tabular functions of time. Each set must consist of time, flow rate, and one value for each variable specified by the symbol Φ on the command, in that order. Thus if 4 variables are selected, then each data set must consist of 6 values. **There is no default value; the correct number of values must be specified.** If the flow rate is negative (withdrawal of fluid), then specified value of the property is ignored since it is assumed that the fluid is being withdrawn with an amount of property equal to the local value of that property.

EXAMPLES

SOURce with MULTiple TABLe functions: 4 sets

Time	Flow	U	V	T
0.	0.001	1.00	0.02	100
1.	0.002	0.50	0.01	200
2.	0.004	2.00	0.01	500
5.	0.010	5.00	0.02	600

SOURce with FLOW MULTiple TABLe: 4 sets per unit VOLUME

Time	Flow	U	V	T
0.		0.001	1.00	0.02 100
1.		0.002	0.50	0.01 200
2.		0.004	2.00	0.01 500
5.		0.010	5.00	0.02 600

MODE 6: Solubility-Limited Source for a Chemical Species**SYNTAX** **SOUR** { Φ } {SOLU} [func [ξ]] [subrgn] [fname] {N1,..., Nn} {Nn+1} [Nn+2] Φ A symbol that denotes the dependent variable for one of the chemical species. A symbol must be specified.**SOLU** The source is solubility-limited. That is, the source is specified as the total initial mass of a chemical species and the solubility of the species in the fluid phase is limited to a maximum saturation value that is defined by Equation 3.10.3.**func** The function that specifies the solubility limit, C_s , for the species. See Mode 1 specification. ξ See Mode 1 specification.**subrgn** See Mode 1 specification.**fname** See Mode 1 specification.**N1, ..., Nn** See Mode 1 specification.**Nn+1** Total amount of material to be dissolved. The default value is 0.**Nn+2** Time at which release begins. The default value is 0.**EXAMPLES**

All examples cited for Mode 1 are applicable provided that the modifier "SOLU" is added and the values for the variables (Nn+1 and Nn+2) are appended at the end of the command, as necessary. Some illustrative examples of the use of attributes specific to this mode are given below.

SOURce for FF: SOLUbility limited: $C_s=0.05$, $S=100$ kg, $t=0$ for selected zone.

SOURce for FU: SOLUbility: 75 sets file 'SOURCE.DAT', $S=75.$, $t_{start} = 20$.

MODE 7: Radiation Type of Source For a Variable

SYNTAX **SOUR** { Φ } {RADI} [COEF] [VARI] {H, Φ_{eq} } [POSI | NEGA] [option] [subrgn]

Φ A symbol to denote the dependent variable for which the source is specified. A symbol must be specified; otherwise, a fatal error will occur.

RADI The radiation type of source is specified. This source is defined as:

$$Q = H (\Phi_{eq} - \Phi) .$$

In the above equation, Q is the source for Φ in appropriate units, H is a transfer coefficient and Φ_{eq} is an equilibrium value of Φ . The H and Φ_{eq} may be constants or symbols for previously defined variables.

COEF By default it is assumed that h is a constant (with its value specified as a numerical constant on the command). However, if this modifier, along with the **VARI** modifier, is present, then h must be a symbol for a previously defined variable.

VARI This modifier must be present with the **COEF** modifier if H is specified as a symbol for a variable. Otherwise it is ignored.

H The transfer coefficient, H, of the radiation source. A numerical value must be specified unless the modifiers **COEF** and **VARI** are present. In this case, a valid symbol for a previously defined variable must be specified.

Φ_{eq} The equilibrium value for the radiation source. A numerical value or a valid symbol for a previously defined variable must be specified.

POSI The source is implemented only if $Q > 0$.

POSI The source is implemented only if $Q < 0$.

option See Mode 1 specification. **NORMAL** and **DENSITY** options are not available with this type of source

subrgn See Mode 1 specification.

EXAMPLES

SOURCE for T: RADlation tcoef 0.001, equilibrium value = 15 degrees.

SOURCE for T: RADlation .hcoef = 0.001 value from variable EQVALu !EQVA is a symbol

SOURCE for T: RADlation VARI COEF as function HVALu , value=15 !HVAL is a symbol

SOURCE for T: RADI VARI COEF HCOEF & HVALu !HCOE & EQVA are symbols

MODE 8: Source Term as a Linear Decay or Half-Life.

SYNTAX SOUR {Φ} {DECA} [HALF] [subrgn] {N1}

Φ A symbol to denote the variable for which the source is specified.

DECA The decay rate of the variable is specified; it is equivalent to introducing a source term in the corresponding governing equation for Φ as:

$$Q = - \lambda \rho \Phi,$$

In this equation, Q is the source rate for Φ, λ is the decay rate and ρ is the fluid density if the variable Φ is defined in terms of mass units and unity if it is defined in terms of volumetric units.

LIFE Half-life of the property Φ is specified. The decay rate is computed from the half life, σ:

$$\lambda = - \frac{\log(0.5)}{\sigma}$$

subrgn See Mode 1 specification.

N1 The decay rate, λ, or half-life, λ, for the variable Φ.

EXAMPLES

SOURce DECAy rate for C = 0.001

SOURce DECAy rate for C = 0.001 in the SELEcted subregion

SOURce DECAy rate for C = 0.001 in the subregion ID=DRGN

SOURce DECAy rate for half LIFE of 29 years in subregion ID=DRGN

MODE 9: Source as a Linear Function of User-Defined Reaction Rates**SYNTAX** **SOUR** { ϕ } {**REAC**} {id_1, ..., id_n} [**subrgn**] {N1, ..., Nn} ϕ A symbol to denote the chemical species for which the source is specified.**REAC** Modifier specifying the source term as a linear combination of reaction rates for a given chemical species according to the equation:

$$Q = \sum_n C_n R_n ,$$

In this equation, Q is the source for ϕ , C_n are the scaling constants and R_n are the previously specified reactions.

id_1,..., id_n Identifiers for the reactions, R_n , described by Mode 2 of **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} { ϕ } {VALU | DIFF | GRAD} [AREA | NORM | VOLU] {dir} [subrgn] [func ξ] [N1] [fname | N2, ..,Nn] [Nn+1, Nk] [DENS] [ρ_B]

ϕ A symbol to denote the dependent variable for which the flux transfer is specified. A symbol must be specified.

FLUX The flux, Q, is transferred from a donor to a receptor element across an interface between the two elements according to the relation:

$$Q = q F_{\phi} F_V F_{\rho}$$

Here, q is the amount specified by the user (func (ξ) and N1 through Nn). The factors F_{ϕ} , F_V and F_{ρ} are functions of other modifiers as described below. The quantity Q is algebraically added to the source for the receptor cell and subtracted from the source term of the donor cell. **There is no net source for the system.**

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

DIFF $F_{\phi} = (\phi_B - \phi_F)$, where ϕ_B and ϕ_F are the values of ϕ in the “boundary” and “field” cells, respectively. The field cell is that defined by the subrgn modifier; the boundary cell is selected in reference to the field cell by the dir modifier. The source acts as a diffusive flux and enhances equilibrium between neighboring elements. For this type of source, the boundary cell acts as the donor cell and the field cell as the receptor cell. This option can be used to incorporate supplementary equilibrium, diffusive or radiative processes.

GRAD $F_{\phi} = (\phi_B - \phi_F) / L$, where L is the Eulerian distance between the donor and receptor cell nodes. This source is essentially the same as that specified by the DIFF modifier.

AREA The factor F_V is set equal to the area of the receptor cell boundary indicated by the dir modifier. If the AREA, NORM and VOLU modifiers are all absent, then F_V is set to unity.

NORM $F_V = \sum A_i V_i$, where A_i is the projected area of the interface in the direction of the i^{th} coordinate denoted by dir modifier, and V_i are the user inputs, Nn+1 through Nk. The summation is over 2 terms for 2D, and 3 for 3D simulations.

VOLU The factor F_V is set equal to the volume of the receptor cell.

subrgn See Mode 1 specification.

func See Mode 1 specification.

ξ See Mode 1 specification.

fname See Mode 1 specification.

N1, .., Nn See Mode 1 specification.

Nn+1, .., Nk See Mode 1 specification.

DENS F_{ρ} is equal to ρ_B . In the absence of this modifier, F_{ρ} is equal to unity.

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

the default value is the local density for the element.

EXAMPLES

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

SOURCE FLUX type for T: $q=0.1$ for X- direction of currently active subregion.

SOURCE FLUX type for C1: $q=1.5$ multiply by AREA of X- direction interface for ID=RGN1.

SOURCE FLUX for C2: $q=1.5$ NORMALized velocities 0., 0., 0.25 for X- direction of ID=RGN1.

SOURCE FLUX C2: $q=1.5$ NORM 0.12., 0., 0.25 multiply by DENSITY; X- direction ID=RGN1.

SOURCE FLUX C2: DIFFerence. $q=1.5$ NORM 0.12., 0., 0.25 DENSITY = 5; X- ID=RGN1.

SOURCE FLUX C2: GRADient. $q=1.5$ VOLUmetric DENS; X- ID=RGN1.

MODE 11: Disable Previously Specified SOURCE commands

SYNTAX **SOUR** {OFF} { Φ } {subrgn}

Φ See Mode 1 specification.

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

subrgn See Mode 1 specification.

EXAMPLES

SOURCE OFF for T for most recentlu SELEcted region

SOURCE OFF for T for ID=MIDDLE

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 $\mathbf{x}_i = \mathbf{T}_{ji} \mathbf{X}_j$. The index **j** varies the fastest. Each row of the transformation matrix is interpreted as a unit vector directed along the \mathbf{x}_i direction. This is the default option.
- PLAN** The plane of transformation is specified by the coordinates of a point on the plane and the direction cosines normal to the plane. The transformation is then computed by taking \mathbf{x}_1 to be the unit normal to the plane, \mathbf{x}_2 is computed by intersecting the plane with the bounding box of the domain, and \mathbf{x}_3 is computed as the cross product of \mathbf{x}_1 and \mathbf{x}_2 . The user has no control over the directions of \mathbf{x}_1 and \mathbf{x}_2 . However, they are guaranteed to be in the specified plane and mutually perpendicular.
- POIN** The plane of transformation is specified by the coordinates of 2 points. The plane is assumed to pass through the 2nd point and the normal points from the 1st to the 2nd point. The transformation is computed in the same way as the previous option.
- N1, ..., Nn** The 1st through 3rd components of the transformation matrix if the **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 1st point if the **POINT** modifier is present. There is no default value.
- Nn+1, ..., Nm** The 4th through 6th 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 2nd point (on the plane of transformation) if the **POINT** modifier is present. There is no default value.
- N7, ..., N9** The 7th through 9th 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      0      0
0      0     -1
0      1      0
    
```

STACK TRANSformation between local & global grid is:

```

1.0          0.0          0.0
0.0          0.9950040    -0.0998334
0.0          0.0998334    0.9950040
    
```

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

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

STACK TRANSformation between POINTS (0.12, 0.55, 0.012) (0.15, 0.47, 0.012)

MODE 3: Store Scaling Factors to Normalize Output of Variables

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

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

$$\Phi_{\text{out}} = a \Phi + b$$

Φ A symbol that denotes the variable for which the output is required. The valid symbols include those listed in Table 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 **STAC 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] [V_{frq}] [TIME]

Φ A symbol for the dependent variable for which the statistics are required. The computed statistics consist of the minimum, maximum, mean and standard deviation, and the location of the minimum and the maximum. The valid symbols are listed in Table 6.8.1. One and only one character string must be specified for each command.

subrgn The subregion for computation of statistics. If no subregion is specified, the entire domain is selected.

OFF Computation of statistics for the specified variable and subregion is deactivated.

fname The file name for output. The default file name is 'acr_STATS.TMP'. A different output file may be specified for each command. If no file name is specified then the output is directed to the most recent previously specified file. If no previous file was specified, then the output is directed to the default file. The total number of open files can not exceed 64. A summary of output is also printed to the standard output device at the end of simulations.

NO This modifier is effective only in the presence of the TABL modifier. It disables some default features of the command.

TABL A tabulated summary of the computed statistics appears on the standard output file at the end of simulations. If the NO and the TABL modifiers are present, then this output is suppressed.

V_{frq} The frequency (step or time interval) of output. See Section 7.2.11 for further details. The default value is set so that output is obtained at every step of simulations.

TIME By default, V_{frq} is interpreted to be the frequency of computations in terms of number of steps. If this modifier is present, then V_{frq} is interpreted to be the time interval between successive computations.

COMMENTS

The user should perform these computations only at the required frequency since frequent computations will result in increased computation time.

EXAMPLES

STATistics for U for the entire domain

STATistics for T for SELEcted region every 20 steps

STATistics for T for SELEcted region every 20 steps with NO TABLes of output

STATistics for T for subregion defined by ID=VAULt every TIME=0.50 units

STATistics for T for ID=VAULt OFF !!! Switch off previously specified STAT command

STATistics for U to 'FLUX.OUT' for SELEcted region at TIME interval of 0.4

COMMAND **STORAGE**

PURPOSE To specify the storage coefficient for the governing differential equations.

COMMENTS

This command is identical to the **DISTribution command.** The term “distribution coefficient” is normally applied only to the transport of chemical species whereas the term “storage coefficient” applies to any of the governing transport equations.

EXAMPLES

See the **DIS**Tribution command

COMMAND	SWIRL
PURPOSE	To specify the a transformation of the velocity components from an induced swirl.
MODE 1:	Swirl induced by Vanes at Fixed Angles or by Solid Body Rotation
SYNTAX	SWIR { ANGL SOLI } [LOCA FLUX SOUR] [subrgn] [dir] {N1} {N2, ..,Nn} [Nn+1]
ANGL	Swirl is induced in the manner of the flow passing vanes at a fixed angle. With α as the vane angle, the tangential (swirling) velocity, V_T , at the swirl plane is computed as: $V_T = V_N \tan(\alpha),$ <p>where V_N is the normal component of velocity at the surface.</p>
SOLI	Swirl is induced in the manner of solid body rotation. With ω as the angular velocity, the tangential velocity (swirl component), V_T , at the swirl plane is computed as: $V_T = 2 \pi \omega r,$ <p>where r is the radius from the specified center of rotation.</p>
LOCA	The normal component of velocity at the swirl plane, V_N , is computed as: $V_N = \sum_i n_i U_i$ <p>where n_i is the normal vector and U_i, the velocity vector just upstream of the swirl plane. This is the default option if none is specified.</p>
FLUX	The normal component of velocity at the swirl plane, V_N , is computed as: $V_N = \frac{q}{\rho A}$ <p>where ρ is the fluid density, q is the flow rate and A is the area of the surface.</p>
SOUR	The normal component of velocity at the swirl plane, V_N , is computed as: $V_N = \frac{q}{\rho A}$ <p>where ρ is the fluid density, q is the source and A is the area of the surface. If this modifier is present, then a TRANSFER command with SOURCE modifier must have been previously specified for the same subrgn.</p>
subrgn	The subregion that defines the surface at which the velocity transformation occurs. If the subregion was previously defined by a LOCATE MATCH command, then the transformation is applied only to the 2 nd surface. A subregion must be defined; there is no default value.
dir	The orientation index for the element boundary associated with the input. See Section 7.2.5 for available choices. There is no default value for this input. A value must be supplied unless the subrgn was defined by a LOCATE PAIR or LOCATE MATCH command.

- N1 The vane angle, α , in degrees if the **ANGLE** modifier is specified or the angular velocity of rotation, ω , if the **SOLID** modifier is specified. The numerical value of α must be less than 89.9. **There is no default value for this input.**
- N2, Nn The (x,y) or (x,y,z) coordinates of the center of the swirl around which the velocity vector is rotated. The center must lie in the plane of the surface defined **subrgn** and **dir**. The transformed tangential velocity is assumed to lie in this plane. Two values are needed for 2D and 3 for 3D geometry. **There is no default value for this input.**
- Nn+1 The radius of the swirler. This is radius of the circle (in the plane of the surface) within which the velocity components are transformed. Any velocity components in the surface outside this radius are not transformed. **The default value 10^{30} .**

COMMENTS

The local Cartesian components of velocity at the swirl plane are computed from the normal and tangential velocity components by transforming from cylindrical to cartesian components. The global Cartesian components of velocity are obtained by transforming the local components as:

$$U_j = T_{ij} u_i$$

where T_{ij} is the transformation vector and u_i is the velocity vector.

The transformation is applied only if the computed $V_N > 0$; otherwise the command is ignored. The computed Cartesian velocity components are set at a location just upstream of the swirl surface. A **BLOCK or **FIX** command must be used to ensure that these values are not overwritten by values computed from the governing momentum equations.**

EXAMPLES

- SWIRL** velocity with a vane **ANGLE** of 50 degrees center at (0.14, 0.22, 0.0) ID=Q_TRANSFER
- SWIRL** vane **ANGLE** = 50 degrees VN from LOCAL velocity; center at (0.14, 0.22, 0.0) ID=Q_TRANSFER
- SWIRL** vane **ANGLE** = 50 degrees VN from FLUX; center at (0.14, 0.22, 0.0) ID=Q_TRANSFER
- SWIRL** vane **ANGLE** = 50 degrees VN from SOURce; center at (0.14, 0.22, 0.0) ID=Q_TRANSFER
- SWIRL** velocity **ANGLE** of 50 deg. Center: (0.14, 0.22, 0.0) within radius of 0.25 meters ID=Q_TRANSFER
- SWIRL** with angular speed of 200 radian/second center at (0.14, 0.22, 0.0) ID=Q_TRANSFER
- SWIRL** with 200 radians/second ; center: (0.14, 0.22, 0.0) within radius of 0.25 meters ID=Q_TRANSFER

MODE 2: Swirl from User Specified Value of Function

SYNTAX **SWIR** [LOCA| FLUX| SOUR] [subrgn] [dir] {N1} [fname| N2, ..Nn] [Nn+1,..,Nm] [Nm+1]

LOCA See Mode 1 Specification.

FLUX See Mode 1 Specification.

SOUR See Mode 1 Specification.

subrgn See Mode 1 Specification.

dir See Mode 1 Specification.

fname The name of the file from which numerical values **N2** through **Nn** are read. This option can be used only if the selected function is a table or one of the series functions. See Section 7.2.2 for further information.

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

Nn+1, ...,Nm The (x,y) or (x,y,z) coordinates of the center of the swirl around which the velocity vector is rotated. The center must lie in the plane of the surface defined **subrgn** and **dir**. The transformed tangential velocity is assumed to lie in this plane. Two values are needed for 2D and 3 for 3D geometry. *There is no default value for this input.*

Nm+1 The radius of the swirler. This is radius of the circle (in the plane of the surface) within which the velocity components are transformed. Any velocity components in the surface outside this radius are not transformed. *The default value 10^{30} .*

EXAMPLES

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

SWIRL tangential velocity 0.25; center at (0.14, 0.22, 0.0) ID=SWIRL

SWIRL tangential velocity 0.25; normal from LOCAL; at (0.14, 0.22, 0.0) ID=SWIRL

SWIRL tangential velocity 0.25; normal from FLUX; at (0.14, 0.22, 0.0) ID=SWIRL

SWIRL tangential velocity 0.25; normal from SOUR; at (0.14, 0.22, 0.0) ID=SWIRL

SWIRL factor of 0.25 ; center: (0.14, 0.22, 0.0) within radius of 0.25 meters ID=SWIRL

MODE 3:	Swirl Profile or Mapping from a data file
SYNTAX	SWIR [PROFile] [RADial] {subrgn} {dir} {N1,N2,N3,N4} {fname} [SCALE [NONE] N5,N6] [AXIS N7,N8,N9] [ALWAYS] [FUEL = N10] [DIAGnostics] ($\Phi_1.. \Phi_N$)
PROF	Implies that a profile of the velocity components and other specified variables be read in from a data file.
RADI	Implies a radial swirler. The default is an axial swirler. <i>For an axial swirler, the axial velocity component is scaled to match the mass flow.</i> The other velocity components are then scaled by the same factor. If the turbulent kinetic energy is listed as one of the dependent variables on this command, then it is scaled by the square of the scale factor. <i>The velocity profile data is assumed to be a function of the radius.</i> <i>For a radial swirler, the radial velocity component is scaled to match the mass flow.</i> The other velocity components are scaled by the same factor. If the turbulent kinetic energy is listed as one of the dependent variables on this command, then it is scaled by the square of the scale factor. <i>The velocity profile data is assumed to be a function of the axial distance.</i>
subrgn	The subregion that defines the surface at which the velocity transformation occurs and also defined to be an inlet or open or dirichlet boundary. A subregion must be defined; there is no default value.
dir	The orientation index for the element boundary associated with the input. See Section 7.2.5 for available choices. There is no default value for this input. A value must be supplied unless the subrgn was defined by a LOCATE PAIR
fname	The name of the file from which profile data are read in. This file is mandatory. The file consists of an optional header block ending with a “ End Header ” string starting from the first column, followed by columns of data, one per variable. The first four data columns are the <i>independent variable</i> (axial distance for radial swirler / radial distance for axial swirler), <i>axial velocity</i> , <i>radial velocity</i> and the <i>swirl velocity</i> . It may also contain additional columns of data for the other variables specified on the command (in the same order). Extra columns of data are ignored.
N1	The mass flow through the subregion. This subregion should be defined as an INLET or an OPEN or DIRICHLET boundary. There are no default values for this input.
N2, N3, N4	The (X,Y,Z) coordinates of the origin of the local cylindrical system. This input is mandatory for both axial and radial swirlers. There are no default values for this input.
SCALE	Scale factors are specified to normalize the transformed coordinate prior to interpolating the dependent variables from the profile data file. The min and max values of the independent variable should be specified. <i>In the absence of this modifier, the min value is set to zero and the max value to the actual <u>maximum radial extent</u> of the sub-region for an <u>axial swirler</u>, and to the actual min and max values of the <u>axial extent</u> of the sub-region for a <u>radial swirler</u>.</i> The independent variable is thus scaled to lie between 0 and 1.
SCALE NONE	Turns off all scaling. The independent variable is not scaled prior to doing the interpolation.
N5, N6	The min and max scale factors for the transformed coordinate for the SCALE modifier. The coordinate is transformed as $r = (R - N5) / (N6 - N5)$. Must be specified if the SCALE modifier is present without the NONE modifier. In the absence of the SCALE modifier, these values are computed internally as described above.

- AXIS** The axis of the transformation between the global 3D Cartesian system and the local cylindrical system applicable to the defined subregion. **For an axial swirler, this modifier and the associated data may be omitted**, in which case it is computed as the normal pointing into the mesh from a suitable boundary node. **For a radial swirler this input and the associated data is mandatory.**
- N7, N8, N9** The three components of direction vector of the axis of the local cylindrical coordinate system. **There are no default values for a radial swirler.** For an axial swirler, in the absence of the **AXIS** modifier, these are computed as the normal pointing into the mesh from a suitable boundary node.
- FUEL** This optional input specifies a fuel mass flow rate. The presence of this modifier causes the Fuel Mass Fraction, **FU**, to be scaled to match the specified fuel mass flow rate.
- N10** The fuel mass flow rate if the **FUEL** modifier is specified. **There is no default.**
- $\Phi_1.. \Phi_N$** Optional list of scalar variables also to be interpolated from the profile data. (except pressure)
- ALWA** This modifier causes the command to be executed at every iteration/time step. The default is to execute only once in the startup phase of the next **SOLVe** command. This behavior is analogous to the **SET** vs. **SET ALWays** command. This modifier may be used if the density is a function of the dependent variables and an iterative procedure is needed to determine the final boundary conditions.
- DIAG** This modifier causes detailed diagnostics to be output. In the absence of the **DIAGnostics** modifier, only a brief summary is output for each swirler.

COMMENTS

This command only applies to a 3D Cartesian framework. A global 3D Cartesian to local cylindrical transformation is computed from the center and the axis. Any requested scaling is carried out. The velocity components and other variables are then interpolated from the profile data. The velocity components are then scaled to match the specified mass flux. These velocity components and other variables are then applied to the boundary specified by the sub-region.

If the turbulent kinetic energy, **K**, is also present on the command line, then it is also scaled as the square of the scaling for the velocity variables. However, the turbulent length scale, **L**, is not scaled.

If the fuel mass flow rate is specified, the fuel mass fraction, **FU**, is adjusted to match the specification.

A positive mass flow specification causes the flow to enter the domain whereas a negative mass flow specification causes the flow to leave the domain. If the data in the velocity profile file is not consistent with the mass flow specification, then a negative scale factor is computed. This has the effect of inverting the profile and reversing the sense of the swirl component.

EXAMPLES

SWIRler **PROF**line mdot=50.0 kg/s center=(0.4,0.5,1.0)
 FUEL mdotf=5 kg/s file='swirl3.dat'
 additional variables in file: T K L FU at ID=RGN3 Z+ **ALWA**ys detailed **DIAG**nostics on.

SWIRler **PROF**ile mdot=10.0 kg/s center=(1.0,0.5,0.5)
 SCALing (3.53553E-02,0.1060660) file='swirl3.dat'
 additional variables in file: T K L at ID=RGN2

The contents of file 'swirl3.dat':

Swirler mapping file for axial swirler:

r	u	Ur	Ut	T	K	L	FU
0	-0.1	0	0.1	300	0.001	0.1	0.01
0.1	-0.2	0	0.2	350	0.002	0.15	0.02
0.2	-0.3	0	0.3	400	0.003	0.2	0.03
0.3	-0.4	0	0.4	450	0.004	0.25	0.04
0.4	0.0	0	0.5	500	0.0001	0.3	0.05
0.5	0.6	0	0.6	550	0.006	0.35	0.06
0.6	0.7	0	0.7	600	0.007	0.4	0.05
0.7	0.8	0	0.8	400	0.008	0.45	0.04
0.8	0.9	0	0.9	300	0.009	0.5	0.03
0.9	1.0	0	1.0	300	0.01	0.55	0.02
1.0	1.0	0	0.0	300	0.01	0.6	0.01

SWIRler **PROF**ile **RAD**ial mdot=50.0 kg/s center=(0.5,0.5,0.5)
 AXIS=(1,0,0) file='swirl2-r.dat'
 additional variables in file: T K L at ID=RGN1

SWIRler **PROF**ile **RAD**ial mdot=50.0 kg/s center=(0.5,0.5,0.5) AXIS=(1,0,0)
 FUEL mdotf=5 kg/s file='swirl3-r.dat'
 additional variables in file: T K L FU at ID=RGN1 **ALWA**ys detailed **DIAG**nostics on.

The contents of file 'swirl3-r.dat':

Swirler mapping file for radial swirler:

X	U	Ur	Ut	T	K	L	FU
0	-0.5	-0.0	0.0	300	0.005	0.1	0.01
0.1	-0.4	-0.1	0.1	350	0.004	0.15	0.02
0.2	-0.3	-0.2	0.2	400	0.003	0.2	0.03
0.3	-0.2	-0.3	0.3	450	0.002	0.25	0.04
0.4	-0.1	-0.4	0.4	500	0.001	0.3	0.05
0.5	0.0	-0.5	0.5	550	0.0001	0.35	0.06
0.6	0.1	-0.4	0.4	600	0.001	0.4	0.05
0.7	0.2	-0.3	0.3	650	0.002	0.45	0.04
0.8	0.3	-0.2	0.2	700	0.003	0.5	0.03
0.9	0.4	-0.1	0.1	750	0.004	0.55	0.02
1.0	0.5	-0.0	0.0	800	0.005	0.6	0.01

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 and Table 7.2.3 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**SYMM**etry at Y- for ID=B_SYMMETRY**SYMM**etry OFF for Y- and ID=B_SYMMETRY

COMMAND **THERMAL**

PURPOSE To specify thermal properties of the porous matrix, or of the planar or linear features.

SYNTAX **THER** [N1, N2, N3, N4]

N1 Specific heat of the solid, c_s of Equation 2.2.2 (≥ 0). However, if a **PROPerTy** command with **T** and **EFFEctive** modifiers is specified, the input is assumed to be the effective specific heat for the matrix, α_h of Equation 2.2.6 (≥ 0). **The default value is 1.**

N2 Thermal conductivity of the dry solid, k_s of Equation 2.2.8 (≥ 0). However, if a **PROPerTy** command with **T** and **EFFEctive** modifiers is specified, the input is the effective thermal conductivity of the matrix which is equal to the sum of the first two terms on the right side of Equation 2.2.8 (≥ 0). **The default value is 0.**

N3 Longitudinal dispersivity, α_L of Equation 3.9.1. **The default value is 0.**

N4 Transverse dispersivity, α_T of Equation 3.9.1. **The default value is 0.**

EXAMPLES

THERmal properties cs = 26, kt = 45

THERmal props cs = 26, ks = 45., alfa1 = 0., alfa2 = 0.

THERmal properties: ce = 26, ke = 45, alpha1 = 10, alfa2 = 2

COMMAND **TIME**

PURPOSE To set the initial simulation time for a problem.

SYNTAX **TIME** { N1 }

N1 The starting time (≥ 0) for simulations. The default value is 0.

EXAMPLES

TIME = 50 years at start of simulations

COMMAND **TITLE**

PURPOSE To specify the problem title.

SYNTAX **TITL** followed by character information

COMMENTS

The specification must be restricted to one 80 character record. Title specification is included in all output files generated by the ACRi Software.

EXAMPLES

TITLe ILLUSTRATIVE PROBLEM - DEFAULT SET UP - 07/01/93:ACRi/akr

COMMAND TRACK

PURPOSE To compute particle tracks and their corresponding elapsed time in the flow field.

MODE 1: Particle Locations and Tracking Options

SYNTAX TRAC [fname] [TABL] [STOP] [option] {N1, ..Nn} [Nn+1, Nn+2, Nn+3]

fname The file name for output. The default file name is 'acr_TRACK.TMP'. A different output file may be specified for each command. If no file name is specified then the output is directed to the most recent previously specified file. If no previous file was specified, then the output is directed to the default file. The total number of open files can not exceed 64. A summary of output is also printed to the standard output device at the end of simulations.

TABL The particle track data are printed in a tabular form at the end of simulations.

STOP The particle stopping criterion is explicitly specified as one of the options listed in the table below. If no stopping criterion is specified then the particle is tracked to the end of simulations or till it reaches or crosses a boundary element.

option The stopping option for the particle; the modifier STOP must be present for one of these options to be effective.

option	INTERPRETATION
X	Particle tracking stops when its x-coordinate exceeds the Nn+1 th numerical value.
Y or R	Particle tracking stops when its y-coordinate exceeds the Nn+1 th numerical value.
Z or THEta	Particle tracking stops when its z-coordinate exceeds the Nn+1 th numerical value.
DIST	Particle tracking stops when its distance from the point of release exceeds the Nn+1 th numerical value. This is default option if STOP modifier is specified.
TIME	Particle tracking stops when the time exceeds the Nn+1 th numerical value.
ELAP	Particle tracking stops when the elapsed time from its moment of release exceeds the Nn+1 th numerical value.
FREE	Particle tracking stops when the particle reaches a free surface or zone of saturation.

N1, ..., Nn The coordinates of the starting location of the particle. Two values are required for the 2D and 3 for 3D input modes.

Nn+1 The numerical value pertaining to the stopping criterion as described under options listed in the table above. There is no default value. A dummy numerical value must be specified if any numerical input follows for the time or frequency input below.

Nn+2 In the transient solution mode, the starting time for the particle; the default value is 0. In the steady state solution mode, the time interval for the particle computations; the default value is 1.

Nn+3 The frequency index for providing the output in the particle track file. The output is obtained every Nn+3 (≥1) steps. For example, a specification of 10 result in output at the 10th, 20th, 30th, etc. step. If no input is specified, the frequency is assumed to be 1.

EXAMPLES

TRACks for particle start at: (35.0, 5.27)

TRACks for particle start at: (1.22, 10.0, 19.3) print **TABLe**s also

TRACk particle at: (1.22, 10.0) print **TABLe**s and save on file 'TRACK.SAV'

TRACk particle at: (1.22, 10.0, 19.3) **STOP** at X=100. starting time = 20 years

TRACk at: (1.2, 10.0) stop_distance 150 m; delta_t 0.1 yrs (steady state mode)

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 ELAP**sed **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**TRAC**king FACTor: is 0.5 in the currently SELEcted subregion**TRAC**king FACTor is 2.5 in the subregion with ID=FAST**TRAC**king FACTor: DIVIde particle velocity by 2.0 !! to account for retardation

COMMAND **TRANSPORT**

PURPOSE To specify the transport properties of the host porous matrix, or of planar or linear features.

SYNTAX **TRAN** [Φ] [*phase* | **SOLI**] [**WET**] [**EXPO**] {**N1**} [**N2**, **N3**, **N4**]

Φ A symbol to denote the dependent variable for which the transport properties are specified. The valid symbols are those pertaining to transport equations of Table 6.7.1. *By default, the specification is assumed to be for the 1st transport equation or chemical species.*

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.*

SOLI The transport equation for the solid phase of the porous matrix (Equation 2.5.1) is solved.

WET *The modifier is effective only if the **SOLI**d modifier is also present on the command.* By default, the accumulation term for the solid phase transport assumes that the total amount of solid is available for transport of the species. If this modifier is present, then only the wetted part of the solid participates in transport. That is the accumulation term of Equation 2.5.1 is further multiplied by the saturation, *S*, of the first fluid phase.

EXPO The diffusion coefficient is computed from the exponential formulation of Equation 2.4.11.

N1 The partition coefficient, k_d of Equation 2.4.5 (≥ 0). However, if a **PROPerTy** command with appropriate symbol and **EFFEctive** modifier is present, the coefficient is the retardation factor, R_d of Equation 2.4.7 (> 0). *The default value is 0 for k_d and 1 for R_d .*

N2 Molecular diffusivity, D_M of Equation 2.4.9 (≥ 0), for the chemical species. *The default value is 0.*

N3 The coefficient α_1 of Equation 2.4.11 or the longitudinal dispersivity, α_L of Equation 3.9.1 (≥ 0). *The default value is 0.*

N4 The coefficient α_2 of Equation 2.4.11 or the transverse dispersivity, α_T of Equation 3.9.1 (≥ 0). *The default value is 0.*

EXAMPLES

TRANsport properties: $k_d=1$, $D_m=5.24$, $\alpha_L=10$, $\alpha_T=1$
TRANsport properties for C2: $k_d=1$, $D_m=5.24$, $\alpha_L=10$, $\alpha_T=1$
TRAN: $k_d=0$, $d_m=1.E-3$, $L=10$, $T=1$
TRANsport $k_d=1.1E-3$, $d_m=0.$, $L=10$, $T=1$
TRANsport $R_d=21$, $d_m=0.$, $L=10$, $T=1$
TRANsport for C2 in **SOLI**d phase
TRANsport for C2 in **SOLI**d phase $k_d=0$, $d_m=1.E-5$
TRANsport for C2 only in **WET** part of **SOLI**d phase.
TRANsport **EXPO**ntial form: $k_d=1.1E-3$, $d_m=1.2$, $a_1=10$, $a_2 = 5$

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 **VELO**

PURPOSE To specify the direct computation of Darcy velocity from pressure field.

SYNTAX **VELO** {**PRES**}

PRES By default the representative Darcy velocity at a grid node is computed by dividing the arithmetic average of the flux by the average directed area of the faces. For example, for the hexahedral element shown in Figure 7.2.1, the U, V and W velocity components are computed as the averages of flux at each of the faces divided by the total face area and multiplied by the directed area in the x, y and z direction, respectively. If **PRES**surre modifier is present, then the velocity components are directly computed from equation 2.2.3.

EXAMPLES

VELOcity directly from **PRES**SURE gradient

COMMAND **VISCOSITY****PURPOSE** To specify the viscosity and to select associated options.**MODE 1:** **Generic Functional Form for Viscosity****SYNTAX** **VISC** {func [ξ]} [phase | EFFE] [ADD] [subrgn] {fname | N1 ..., Nn}

func One of the modifiers listed in Table 7.2.4 that denotes the functional form of the input. For this input, the function specifies the value of the viscosity for the corresponding phase. If no function is specified then the value is assumed to be constant.

ξ One of the independent variables listed in Table 7.2.5. If no variable is specified, then the independent variable is assumed to be time.

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

EFFE The input is applied to the effective or mixture viscosity of the fluid. By default, the specified viscosity is assumed to be molecular viscosity of the fluid. For a turbulent flow, the effective mixture viscosity is computed as the sum of the molecular and turbulent viscosity. If this modifier is present, then the specified values replace any internally computed values.

ADD The input is added to the current value of viscosity. This modifier is effective only if the EFFECtive modifier is also present.

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

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

COMMENTS

This mode of the VISCosity command is currently available only for ANSWER™ Software Tool

The effective viscosity for the fluid (combined molecular and turbulent components) may also be set by the **SET** Command with **MU** modifier. In this case the modifier **ALWAYS** must be specified if the viscosity is to be set as the specified function throughout the solution process. Otherwise the values will be set only at the time of specification and may be subsequently overwritten by other computations. The **VISCosity** command is recommended as the preferred mode of input.

The effective viscosity may also be set by the **CONDUCTIVITY** command with **U**, **V** or **W** modifiers. This option allows the specification of anisotropic viscosity; that is the **coefficient of viscosity can be specified as a vector** so that the components in the 3 principal directions are different from each other. In this mode, the viscosity vector for each velocity component can be independently specified for each velocity component.

The **VISCOSITY**, **SET** and **CONDUCTIVITY** commands may be specified in combination. In this case, the **CONDUCTIVITY** commands will take precedence over all other commands.

EXAMPLES

VISCosity molecular value = 2.E-5

Generic examples for this command are given in Section 7.2.8. The command keyword (VISC) must

replace the keyword used in these examples.

MODE 2: Pre-Defined Viscosity Functions

SYNTAX **VISC** { [CONS | POLY | EXPO | USER] | [NODE] } [FIEL] [phase] [N1, N2, N3, N4]

CONS The fluid viscosity is constant. This is the default option.

POLY polynomial variation in fluid viscosity according to Equation 3.2.2.

EXPO Exponential variation in fluid viscosity according to Equation 3.2.1.

USER The viscosity is specified by a user specified function.

NODE The viscosity is specified for each node by a separate command such as **INITial**, **READ** or **SET** command. In such a case, this command can be used to modify the specified value according to the one of the relations specified on this command.

FIEL By default the viscosity at the boundaries is computed from the specified relation or option selected by the user. If this modifier is present, then the boundary value of viscosity is set equal to the field value at the nearest neighbor.

phase The fluid phase for which the input is specified. See Section 7.2.10 for available options. If no phase modifier exists, the input is assumed to be for the 1st phase of the fluid.

N1 Reference viscosity, μ^* of Equations 3.2.2 and 3.2.3. The reference value is used only to determine the K_j of the 2nd and 3rd phases from the K^* specified for the primary phase by the HYDRaulic command. For the primary phase, only the viscosity ratio, rather than the absolute viscosity value, is required. The default value is 1 for all phases.

N2 Coefficient a_1 of Equations 3.2.2 and 3.2.3. The default value is 1436 K if the modifier EXPOntial is present; otherwise, it is zero.

N3, N4 Coefficients a_2 and a_3 , respectively, of Equation 3.2.2. The default value is 0 for both.

COMMENTS

This mode of the **VISCosity** command is currently available only for PORFLOW™ Software Tool

EXAMPLES

VISCosity UNIFORM for primary fluid: $\mu^* = 1.002E-3$ (water at 20 deg C)

VISCosity UNIFORM SECONdary phase: $\mu^* = 8E-2$ (Engine oil at 20 deg C)

VISCosity EXPOntial option: $\mu^* = 0.544E-3$, $b = 1450$ K for FIELD values

VISCosity POLYnomial option: $\mu^* = 8.0E-4$, $a=1.E-4$, $b=5.E-5$, $c=0$.

VISCosity NODE basis and multiply by POLYnomial: $\mu^* = 8.0E-4$, $a=1.E-4$, $b=5.E-5$, $c=0$.

VISCosity UNIFORM for THIRd phase: $\mu^* = 1.E-2$

COMMAND **WRITE**

PURPOSE To generate output of selected variables on demand.

MODE 1: **Write Selected Variables to Standard Output File or User-Specified File**

SYNTAX **WRIT** { Φ } [**STAC**] [**fname**] [**subrgn**]

Φ One or more of the symbols that represent the variables for which output is desired. The valid symbols are listed in Tables 6.8.1 and 6.8.2. **There is no default value.**

STAC The output variable is scaled by coefficients **a** and **b** specified on a previous **STAC WRITE** command according to:

$$\Phi_{\text{out}} = a \Phi_{\text{computed}} + b$$

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.

option	INTERPRETATION
INLE	Only the boundaries specified by the INLEt command are selected.
OUTL	Only the boundaries specified by the OUTLEt command are selected.
OPEN	Only the boundaries specified by the OPEN command are selected.
IO	All boundaries specified by INLEt , OUTLEt or OPEN command are selected.
BOUN	All external boundaries of the specified (or default) subregion are selected.
SYMM	Only the boundaries specified by the SYMMETRY command are selected.
WALL	Only the walls are selected.

walltype The type of wall to be selected for output if the **WALL** modifier is present.

walltype	INTERPRETATION
ALL	All walls are selected. This is the default option.
EXTE	Only the exterior walls of the computational domain are selected.
INTE	Only the walls located in the interior of the computational domain are selected.
BLOC	Only the walls of the blocks (BLOCK command) are selected.

SUMM By default, the output generated by this command includes the detailed information for each element of the boundary as well as an overall summary for the boundary as a whole. If this modifier is present, then only the summary information is generated; the element by element details are suppressed.

STAC See Mode 1 Specification.

subrgn The subregion for computations. If no subregion is specified, the entire domain is selected.

dir The orientation index for the boundary. See Section 7.2.5 for available choices. If no input is given, then the output is obtained for all boundaries of the selected type.

fname The file name for output. By default the output is printed only to the standard output device. If a file name is specified, then the output is printed to the named file.

EXAMPLES

WRITE T and RHO for all INLEt boundaries

WRITE T and RHO only for INLEt boundaries; scale output with STACK coefficients

WRITE T and RHO only for INLEt boundaries in X- direction to file 'INLETX.FIL'

WRITE T and RHO only for INLEt boundaries defined by ID=INLEt in X- direction.

WRITE SUMMery for U, T and RHO for INLEt boundaries defined by ID=INLEt in X- direction.

WRITE U, V, P, T and RHO for all OUTLEt boundaries in X+ direction to 'OUTLET.VAL' file

WRITE T and RHO only for WALLs to the file 'WALLS.PRB'

WRITE T and RHO only for EXTERior WALLs to the file 'WALL_EXT.PRB'

WRITE T and RHO only for axis of SYMMetry in Y- direction.

WRITE T for all external BOUNDaries.

MODE 3: Write a Profile of Values at Selected Locations

SYNTAX **WRIT** { Φ } [INTE] [method] [STAC] [subrgn] [fname]

Φ One or more of the symbols that represent the variables for which output is desired. The valid symbols are those listed in Table 6.8.1 and Table 6.8.2. There is no default value.

INTE The variable values s at the selected locations are computed by linear or inverse distance squared interpolation from the computed values at the nearest neighboring nodes.

method The method of interpolation for computing values at the specified (x, y, z) locations. Only one method may be selected for each command.

method	INTERPRETATION
LINE	The values are computed by inverse linear distance interpolation from the computed values at nearest neighbors. This is the default option.
SQUA	The values are computed by inverse distance squared interpolation from the computed values at nearest neighbors.
AVER	The values are computed by arithmetic average of computed values at nearest neighbors.
NEAR	The value is set equal to that at the nearest neighbor.
X	Same as LINE except that the distance is set equal to the separation in the x-coordinate.
Y	Same as LINE except that distance is set equal to the separation in the y-coordinate.
Z	Same as LINE except that the distance is equal to the separation in the z-coordinate.

STAC The output variable is scaled by coefficients **a** and **b** specified on a previous **STAC WRITE** command according to:

$$\Phi_{out} = a \Phi_{computed} + b$$

The **STAC** modifier is effective only for **real** variables. It is ignored if no previous **STACK WRITE** command was specified.

subrgn The sub region for output. The sub region must have been previously specified by a **LOCATE LIST COORDINATE** command. Any other sub region specification will lead to an error condition. There is no default value.

fname The file name for output. By default the output is printed only to the standard output device. If a file name is specified, then the output is printed to the named file.

EXAMPLES

- WRITE** U, V, P and T by **INTE**erpolation for ID=L_PROFILE
- WRITE** U, V, P and T by **SQUA**red **INTE**erpolation for ID=L_PROFILE
- WRITE** U, V, P and T by **INTE**erpolation for ID=L_PROFILE
- WRITE** U by **AVERAGE** **INTE**erpolation for ID=L_PROFILE on file: 'PROFILE.U'
- WRITE** U, V, P and T by **X** direction **INTE**erpolation for ID=L_PROFILE on file: 'PROFILEX.VAR'

MODE 4: Write Selected Variables to User-Specified File in Block Mode**SYNTAX** **WRIT** { Φ } {BLOC} [HEAD] [FIEL] {fname}

- Φ One or more of the symbols that represent the variables for which output is desired. The valid symbols are those listed in Table 6.8.1, and the node or element-based variables listed in Table 6.8.2. There is no default value.
- BLOC** The variables are written in the block format. One record is written for each variable. If the grid is structured then the variable is written in the manner of the FORTRAN DO loop (over the grid indices I, J, K). If the grid is unstructured, then the record for the variable is sequential over element numbers starting with the 1st element.
- HEAD** By default only the numeric values for the selected variable(s) are written to the file without any header information. If the **HEADer** modifier is present, then a two line header in the standard **ACRI SAVE** file format appears before each set of variable values.
- FIEL** Only the inner field nodes or elements are written to the output file. The boundary nodes are omitted. This modifier is active only if the **BLOCK** modifier is simultaneously present. By default both the field and the boundary elements are written to the record.
- fname** The name of the file to which the output is directed. See Section 7.2.2 for additional information. There is no default file name; a file name must be specified.

EXAMPLES

WRITe T in BLOCK format to file named 'value.T'**WRITe** U, V and W in BLOCK format to file 'value.UVW'**WRITe** U, V, W, T, C, FU in BLOCK format to file named 'mixed.val'**WRITe** FIELd values of U and X in BLOCK format 'UandX.val'**WRITe** in BLOCK format values of X Y Z and MYTP to 'geometry.val'

MODE 5: Write Selected Variables to User-Specified File in Block Mode**SYNTAX** **WRIT** { Φ } {ACRI | SAVE} [GEOM|COMP] [DATA] {fname} [fmt]

- Φ One or more of the symbols that represent the variables for which output is desired. The valid symbols are those listed in Table 6.8.1, and the node or element-based variables listed in Table 6.8.2. There is no default value.
- ACRI** The variables are written in the standard **ACRi** format (same as the **SAVE** command). This file is written immediately and closed.
- SAVE** Same as **ACRI**.
- GEOM** By default the file contains problem geometry and grid connectivity variables plus variables specified by the user or active default variables (see Φ above). If **GEOMetry** modifier is present, then **only** the problem geometry and grid connectivity information is written to the file; the Φ variables are omitted.
- COMP** If the **COMPact** modifier is present, then the problem geometry and grid connectivity information is not written to the file; **only** the Φ variables are written.
- DATA** This modifier is effective only if the **COMPact** modifier is also present. By default a standard **ACRi** file header containing information about the problem and nature of data appears at the start of the file. If the **DATA** modifier is present then the file header information is omitted; only the data fields are written to the file.
- 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.
- fmt** The modifier "**FORMatted**" or "**UNFORmatted**", which defines the nature of the data in the archive file. By default, the file is formatted.

EXAMPLES**WRITe** T in ACRi format to file named 'value.T'**WRITe** U, V and W in COMPACT ACRi format to file 'value.UVW'**WRITe** U, V and W in ACRi COMPACT DATA format without file header to file 'data.UVW'**WRITe** GEOMetry only in SAVE format to file 'GEOM.SAV'**WRITe** U, V and W SAVE file in UNFORatted mode to file 'data.UVW'

MODE 6: Write Vertex Coordinates to Standard Output File or User-Specified File

SYNTAX WRIT {VERT} {fname}

VERT A file containing the (x, y, z) coordinates of the vertices is generated. The file contains a header 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, NVRTX
    WRITE(IFILE,*) N, ( XV(K), K=1, N23D)
ENDDO
```

where IFILE is an internally assigned file unit number, NVRTX is the total number of vertices, 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 7: Write Vertex Coordinates to Standard Output File or User-Specified File

SYNTAX WRIT {CORN}

CORN A file containing the (x, y, z) coordinates of the vertices for each element is generated. The name of the file is 'acr_XYZ_CRNR.TMP'. **The command must be given before the first SOLVe command.** The file contains a header followed by a table of vertex numbers and the corresponding grid coordinates for each element. 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
```

where IFILE is an internally assigned file unit number, Total_Elements is total number of elements in the computational domain, Vertices_for_this_Element is the number of vertices for the current element, N is the current vertex 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.

EXAMPLES

WRITE CORNers of the vertices

WRITE vertex CORNers to the default file

COMMAND **ZONE**

PURPOSE To specify the material or zone type for the soil or rock formation.

MODE 1: **Specification of Material Type**

SYNTAX **ZONE** [FRAC | BORE] [COORD] {N1} {N2, ..., Nn} [Nn+1, Nn+2]

MODE 2: **Material Type Data Input from a File**

SYNTAX **ZONE** {fname}

COMMENTS

This command is retained primarily to provide compatibility for data sets prepared with previous versions of **PORFLOW™**. Mode 1 and Mode 2 of the **MATER**ial, **ROCK** or **SOIL** commands have replaced its function. These latter commands should be used whenever possible.

EXAMPLES

See Mode 1 of the **MATER**ial command.

(This page left intentionally blank)

REFERENCES

- Bear, J., 1972.** *Dynamics of Fluids in Porous Media*. American Elsevier, New York, New York.
- Bixler, N. E., 1985.** NORIA-A Finite Element Computer Program for Analyzing Water, Vapor, Air, and Energy Transport in Porous Media, Report No. SAND84-2057, UC-70, Sandia National Laboratories, Albuquerque, New Mexico 87185.
- Brooks, R.H. and A.T. Corey, 1966.** Properties of porous media affecting fluid flow. J. Irrig. Drain. Div., Am. Soc. Civil Eng. 92 (IR2), pp. 61-88.
- Burdine, N.T., 1953.** Relative permeability calculations from pore-size distribution data. Petr. Tans., Am. Inst. Mining Metall. Eng., 198, pp. 71-77.
- Freeze, R.A., and J.A. Cherry, 1979.** *Groundwater*, Prentice-Hall, Englewood Cliffs, New Jersey.
- Harten, A., 1983.** High Resolution Schemes for Hyperbolic Conservation Laws. *Journal of Computational Physics*, Vol. 49, pp. 357-393.
- Kemper, W. O., and J. C. van Schaik, 1966.** Diffusion of Salts in Clay-Water Systems. *Soil Sc. Am. Proc.* 30:534-540.
- Leonard, B.P., 1979.** A Stable and Accurate Convective Modeling Procedure based on Quadratic Upstream Interpolation. *Comp. Methods Appl. Mech. Eng.*, 19,59-98.
- Mualem, Y., 1976.** A new model for predicting the hydraulic conductivity of unsaturated porous media. *Water Resources Res.* 12, 3, pp. 513-522.
- Perry, R.H. and C.H. Chilton, 1973.** *Chemical Engineers' Handbook*, 5th Edition, McGraw Hill, New York.
- Roache, P.J., 1972.** *Computational Fluid Dynamics*, Hermosa Publishers, Albuquerque, New Mexico.
- Runchal, A.K., 1972.** Convergence and Accuracy of Three Finite Difference Schemes for a Two-Dimensional Conduction and Convection Problem, *Int. J. Num. Meth. Eng.*, 4, pp. 541-550.
- Runchal, A.K., 1977.** Comparative Criteria for Finite Difference Formulations for Problems of Fluid Flow, *Int. J. Num. Methods in Eng.*, Volume 11, pp. 1667-1679.
- Runchal, A.K., 1987a.** Theory and Application of the PORFLOW Model for Analysis of Coupled Fluid Flow, Heat and Radionuclide Transport in Porous Media, in *Coupled Processes Associated with Nuclear Waste Repositories*, pp. 495-516, Academic Press, 1987, Ed. C-F. Tsang.
- Runchal, A. K., 1987b.** CONDIF: A Modified Central-Difference Scheme for Convective Flows, *Int'l J. Num. Methods in Engg.*, 24. 1593-1608.
- Scheidegger, A.E., 1961.** General Theory of Dispersion in Porous Media, *J. Geophys. Res.*, 66, pp. 3273-3278.
- Spalding, D.B., 1972.** A Novel Finite-Difference Formulation for Differential Expressions Involving Both First and Second Derivatives, *Int. J. Num, Methods Eng.*, 4, pp. 551-559
- van Genuchten, R., 1978.** Calculating the unsaturated hydraulic conductivity with a new closed-form analytical model, Report 78-WR-08, Water Resources Program, Department of Civil Engineering, Princeton University, Princeton, NJ 08540.
- Varga, R.S., 1962.** *Matrix Iterative Analysis*, Prentice-Hall, Englewood Cliffs, New Jersey.
- Vargaftik, N.B., 1975.** *Tables of Thermophysical Properties of Liquids and Gases*, 2nd Ed., John Wiley & Sons, New York, NY.
- Wheeler, J. A., 1973.** Simulation of heat transfer from a warm pipeline buried in permafrost, Paper 27b, 74th National Meeting, American Institute of Chemical Engineers, New Orleans, Louisiana, 11-15 March.

APPENDIX A

PARTIAL LIST OF PUBLICATIONS

PORFLOW™ has been extensively used over the last 20 years. More than 100 publications and project reports on the benchmarking, verification and application of **PORFLOW™** are currently available. This appendix presents a partial list of these publications.

- C.K. Ho, R.G. Baca, S.H. Conrad, G.A. Smith, L-J. Shyr & T.A. Wheeler, 1999. Stochastic Parameter Development for PORFLOW Simulations of the Hanford AX Tank Farm. SAND98-2880 Sandia National Laboratories.
- Jacobs Engineering Group, Inc. (US Department of Energy), 1998. Supplemental Performance Evaluation Methodology for the AX Tank Farm. DOE/RI-98-72, Richland Operations Office, Hanford, WA, September 1998.
- US Department of Energy, 1998. Supplement Analysis for the Tank Waste Remediation System. DOE/EIS-0189-SA2. Richland Operations Office, Hanford, WA, January 1998.
- Kline, N.W., 1997. Initial Acceptance Testing of PORFLOW Version 3.0. HNF-SD-WM-CSWD-085, Rev. 0, Lockheed Martin Services, Inc., Hanford, WA, December 1997.
- Kline, N.W., 1997. Vadose Zone Flow Model: Validation with Experimental Data. American Nuclear Soc. 1997 Annual meeting, Orlando, FL, June 1-6.
- U.S. Army Corps of Engineers, 1996. Final Remedial Design Submittal Contaminated Groundwater Remediation Explosives Washout Lagoons. Seattle District, U.S. Army Corps of Engineers Seattle, WA, January 1996.
- Kline, N.W., 1996. Unit Cell Modeling in Support of Interim Performance Assessment of Low Level Tank Waste Disposal. WHC-SD-WM-RPT-225, Westinghouse Hanford Company, Richland, WA
- Marivoet, J., G. Volckaert, J. Wubin, I. Wemaere, 1995. Sensitivity Analysis of a Geologic Repository for radioactive Waste Disposal in Clay. 5th Int. Conf. Radioactive Waste Management and Env. Remediation, ICM '95, ASME, Sep 3-5, 1995, Berlin, Germany.
- Kline, N.W. 1995. Development of Unit Cell Model for Interim Performance Assessment of Vitrified Low Level Waste Disposal. WHC-SD-WM-RPT-189, Westinghouse Hanford Company, Richland, WA.
- Mann, F.M., C.R. Eiholzer, N.W. Kline, B.P. McGrail and M.G. Piepho, 1995. Impacts of Disposal System Design Options on Low-Level Glass Waste Disposal System Performance. WHC-EP-0810, Westinghouse Hanford Company, Richland, WA.
- Kline, N.W., 1994. Certification of Version 1.2 of the PORFLOW-3 Code for the WHC Scientific and Computer Center. WHC-SD-WM-CSWD-069, Westinghouse Hanford Company, Richland, WA.
- Piepho, M.G., 1994. Grout Performance Assessment Results of Benchmark, Base, Sensitivity and Degradation Cases. WHC-SD-WM-TI-561, Westinghouse Hanford Company, Richland, WA.
- ACRi, 1994. PORFLOW: A Software Tool for Multiphase Fluid Flow, Heat and Mass Transport in Fractured Porous Media - Validation, Version 2.50. Analytic & Computational Research, Inc, Los Angeles, CA.
- Yu, A.D., J.R. Cook, P.S. Lam, E.L. Wilhite and C.S. Smith, 1994. Radiological Performance Assessment for the E-Area Vaults at the Savannah River Site - Part I: Simulation of Radionuclide Migration in the Near-Field. Proc. Simulation Multi-Conferences, Society for Computer Simulation, San Diego, CA, April.
- Hiergesell, R.A., A.D. Yu, J.R. Cook, L.M. McDowell-Boyer, and P.M. Kearl, 1994. Radiological Performance Assessment for the E-Area Vaults at the Savannah River Site -Part II: Simulation of Contaminant Migration in the Far-Field. Proc. Simulation Multi-Conferences, Society for Computer Simulation, San Diego, CA, April.
- Stetson-Harza, 1994. Preliminary Design Report - Whitestown Landfill Remediation, New York State Department of Environment. NYSDEC Site # 633013, February 9, 1994. Stetson-Harza, Utica, NY 13501.
- ACRi, 1994. PORFLOW: A Software Tool for Multiphase Fluid Flow, Heat and Mass Transport in Fractured Porous Media - Validation, Version 2.50. Analytic & Computational Research, Inc, Los Angeles, CA.
- Singleton, K.M. and K.A. Lindsey, 1994. Groundwater Impact Assessment Report for the 216-U-14 Ditch.

WHC-EP-0698, Westinghouse Hanford Company, Richland, WA.

Kline, N.W., 1993. Certification of Version 1.2 of the PORFLOW-3 Code for the Hanford Cray Computer, WHC-SD-ER-CSWD-003, Westinghouse Hanford Company, Richland, WA.

Converse Consultants NW and Pacific Groundwater Group, 1993. Southwest Harbor Cleanup and Redevelopment Project Sediment Dredge Disposal Containment Model Study. Seattle, Washington.

Ranganathan, Vishnu, 1993. The Maintenance of High Salt Concentrations in Interstitial Waters Above the New Albany Shale of the Illinois Basin. *Water Resources Research*, 29, 11, 3659-3670.

Reidel, S.P., V.G. Johnson, N.W. Kline, 1993. Groundwater Impact Assessment for the 216-U-17 Crib, 200 West Area. WHC-EP-0664, Westinghouse Hanford Company, Richland, WA.

Lowe, S.S., W.C. Carlos, J.J. Irwin, R. Khaleel, N.W. Kline, J.D. Ludowise, R.M. Marusich, P.D. Rittman, 1993. Engineering Study of Tank Leaks Related to Hydraulic Retrieval of Sludge from Tank 241-C-106. WHC-SD-WM-ES-218, Westinghouse Hanford Company, Richland, WA.

Williams, Mark D., 1993. Two and Three Dimensional Numerical Simulations of Continuous and Episodic Basin Dewatering Around a Salt Dome with the Formation of Thermal and Brine Plumes. Thesis, Master of Science, Dept of Geological Sciences, Indiana University

U.S. Department of Energy, 1993. Hanford Site Risk Assessment Methodology. DOE/RL-91-45 Rev. 2, US DOE, P.O. Box 550, Richland, WA 99352.

Runchal, A. K. and B. Sagar, 1993. PORFLOW: A Multifluid Multiphase Model for Simulating Flow, Heat Transfer and Mass Transport in Fractured Porous Media, User's Manual - Version 2.41. NUREG/CR-5991; CNWRA 92-003, U.S. Nuclear Regulatory Commission, Washington, DC.

Green, R.T., A.C. Bagtzoglou, G.W. Wittmeyer, B. Sagar, and R.G. Baca, 1992. Computational Analyses of Groundwater Travel Time - A Preliminary Study. Prepared for the Nuclear Regulatory Commission, Contract NRC-02-88-005, CNWRA, SwRI, San Antonio, TX.

Ahola, Mikko and Budhi Sagar, 1992. Regional groundwater modeling of the saturated zone in the vicinity of Yucca Mountain, Nevada. NUREG/CR-5890; CNWRA92-001, U.S. Nuclear Regulatory Commission, Washington, DC.

Magnuson, Swen O. and Andrew D. Yu, 1992. Mechanisms of Migration from Grouted Waste. US Department of Energy National Low Level Waste Conf., LLW 92, Phoenix, AZ.

Martin Marietta Energy Systems, Inc., EG&G Idaho, Inc., Westinghouse Hanford Company and Westinghouse Savannah River Company, 1992. Radiological Performance Assessment for the Z-Area Saltstone Disposal Facility. WSRC-RP-92-1360, Westinghouse Savannah River Company, Aiken, SC.

Stephens, Mark, 1992. Effluent Disposal Assessment Fedhaven-Nalcrest Subdivision for Lake Wales Utilities Company. Job # 2286, Atlanta Testing & Engineering, Imperial Lakes Crown Centre, Suite 218, P.O. Box 5527, Lakeland, Florida 33807, June 5.

Baca, R.G., S.O. Magnuson, H.D. Nguyen and P. Martian, 1992. A Modeling Study of Water Flow in the Vadose Zone beneath the Radioactive Waste Management Complex. EGG-GEO-10068, Idaho National Engineering Lab., Idaho Falls, ID.

Runchal, A. K. and B. Sagar, 1992. PORFLOW: A Multifluid Multiphase Model for Simulating Flow, Heat Transfer and Mass Transport in Fractured Porous Media, User's Manual - Version 2.41. NUREG/CR-; CNWRA 92-003, U.S. Nuclear Regulatory Commission, Washington, DC.

Rockhold, M.L. and Wurstner, S.K., 1991. Simulation of Unsaturated Flow and Solute Transport at the Las Cruces Trench Site Using the PORFLO-3 Computer Code. PNL-7562/UC-702, Pacific Northwest Laboratory, Richland, WA 99352.

- Piepho, M.G. and A.K. Runchal, 1991. A Comparison of Three Methods for Solving Flow Equations of Two-Immiscible Fluids in Variably-saturated Media. WHC-SA-1289-FP, Westinghouse Hanford Company, Richland, WA.
- Walton, J.C. and R. Seitz, 1991. Performance of Intact and Partially Degraded Concrete Barriers in Limiting Fluid Flow. NUREG-CR-5614, U.S. Nuclear Regulatory Commission.
- Principia, S.A., 1991. Impacto Hidrogeologico de la Ampliacion del Metro de Valencia. Informe no. 189/91, Proyecto P-80, Principia, S.A., Velazquez, 94, 28006 Madrid, Spain.
- Magnuson, S.O., R.G. Baca and A.J. Sondrup, 1990. Independent Verification and Benchmark Testing of the PORFLO-3 Computer Code, Version 1.0. EGG-BG-9175, Idaho National Engineering Lab., Idaho Falls, ID.
- Golder Associates, Inc., 1990. Phase 1 Remedial Investigation Report for the Hanford Site - 1100-EM-1 Operable Unit, Volume I. DOE-R-90-8, U.S. Department of Energy, Richland, WA.
- Sagar, B. and Runchal, A. K., 1990. PORFLO-3: A mathematical model for fluid flow, heat and mass transport in variably saturated geologic media -Theory and Numerical Methods, Version 1.0. WHC-EP-0042, Westinghouse Hanford Company, Richland, WA.
- Walton, J.C. and B. Sagar, 1990. Fluid flow through small flaws in impermeable liners: Importance of isolation for hazardous waste. Environmental Science & Technology, Vol 24, 6, p 920-924.
- Walton, J.C. 1990. Mathematical Modeling of Mass Transport and Chemical Reaction in Crevice and Pitting Corrosion. Corrosion Science, Vol 30, 8/9, p. 915-928.
- Walton, J.C., R.G. Baca, J.B. Sisson and T.R. Wood, 1990. Application of Soil Venting at a Large Scale: A Data and Modeling Analysis. Fourth Nat'l Outdoor Action Conference on Aquifer Restoration, National Water Well Association, Dublin, Ohio, p 619-632.
- Smoots, J.L., B. Sagar, 1990. Three-Dimensional Plume Dynamics in the Vadose Zone: PORFLO-3 Modeling of the Defense Waste Leak at Hanford. 1st Int'l High Level Radioactive Waste Management Conf., Las Vegas, April 8-12, p 537-543.
- Rockhold, M.L., B. Sagar and M.P. Connelly, 1990. Multi-Dimensional Modeling of Unsaturated Flow in the Vicinity of Exploratory Shafts and Fault Zones at Yucca Mountain, Nevada. 1st Int'l High Level Radioactive Waste Management Conf., Las Vegas, April 8-12, p 1192-1199.
- Walton, J.C. 1990. Theoretical Modeling of Crevice and Pitting Corrosion Processes in Relation to Corrosion of Radioactive Waste Containers. Materials Research Society, Vol 176, p 509-516.
- Walton, J.C. and M.D. Otis, 1990. Concrete Barrier Performance in Radioactive Waste Disposal in the Unsaturated Zone. Materials Research Society, Vol 176, p 711-717.
- Smoots, J.L., B. Sagar, 1990. Three-Dimensional Contaminant Plume Dynamics in the Vadose Zone: Simulation of the 241-T-106 Single-Shell Tank Leak at Hanford. PNL-7221, Battelle Pacific Northwest Lab, Richland, WA.
- Smoots, J.L., J.E. Szecsody, B. Sagar, G.W. Gee and C.T. Kincaid, 1989. Simulations of Infiltration of Meteoric Water and Contaminant Plume Movement in the Vadose Zone at Single-Shell Tank 241-T-106 at the Hanford Site. WHC-EP-0332, Westinghouse Hanford Company, Richland, WA.
- Runchal, A. K. and B. Sagar, 1989. PORFLO-3: A mathematical model for fluid flow, heat and mass transport in variably saturated geologic media -User's Manual, Version 1.0. WHC-EP-0041, Westinghouse Hanford Company, Richland, WA.
- Sagar, B., M.P. Connelly and P.E. Long, 1989. Modeling of hydrologic perturbations during reverse circulation drilling. J. Ground Water, American Assoc. of Ground Water Hydrologists.

- Baca, R.G., J.C. Walton, A.S. Rood and M.D. Otis, 1988. Organic contaminant release from a mixed waste disposal site: A computer simulation study of transport through the vadose zone and site remediation. *Proc. Tenth Annual DOE Low-level Waste Management Conf.*
- Hembise, O. and P. Loiseau, 1988. Etude tridimensionnelle d'une fuite de gaz: Gaz de France. *Principia Recherche Developpement, Report 88.11.279b.*
- Hembise, O. and P. Loiseau, 1988. Etude bidimensionnelle d'une fuite de gaz: Gaz de France. *Principia Recherche Developpement, Report 88.11.279.*
- Loiseau, P., 1988. PORFLOW: Un code de calcul pour simuler les transferts en milieu poreux. *M.Sc. Thesis, Institut National Polytechnique de Toulouse, Ecole Nationale Supérieure, Toulouse, France.*
- Walton, J.C. and B. Sagar, 1988. Modeling performance of steel containers in high-level waste repository environments: Implications for waste isolation. *Radioactive Waste Management and the Nuclear Fuel Cycle, Vol 9 (4), pp. 323-347, Harwood Academic Publishers.*
- Walton, J.C. and B. Sagar, 1988. Mathematical modeling of Copper Container Corrosion: the Transport Limited Approach. Ed. R.G. Post, *Proc. Symposium on Waste Management '88, p. 711-717, Univ. of Arizona, Tucson, Arizona.*
- Gough, J.M., 1988. Validation Test of the Well Simulation Program PORFLOW3. *Principia Mechanical Ltd., London, U.K.*
- Aimo, N.J., 1987. Preliminary numerical study of the effects of fluctuating river level on solute transport within riverbank aquifers. *M.Sc. Thesis, Washington State University.*
- Runchal, A.K., 1987. Theory and application of the PORFLOW model for analysis of coupled fluid flow, heat and radionuclide transport in porous media, in *Coupled Processes Associated with Nuclear Waste Repositories*, pp. 495-516, Academic Press, 1987, Ed. C-F. Tsang.
- Walton, J.C. and B. Sagar, 1987. A Corrosion model for nuclear waste containers. *Materials Research Society, Symp. Proc. Vol 84, p 271-282.*
- Bardey, P., V. Perrin and G. Barrot, 1987. Etude de Taux de Fuite d'une Enceinte de Confinement: CEA. *Principia Recherche Developpement, Report 86.02.074.*
- Runchal, A. K., B. Sagar, R. G. Baca, and N. W. Kline, 1985. PORFLO - A continuum model for fluid flow, heat transfer, and mass transport in porous media: model theory, numerical methods and computational tests. *RHO-BW-CR-150P, Rockwell Hanford Operations, Richland, WA.*
- Eyler, L.L. and M.J. Budden, 1984. Verification and Benchmarking of PORFLO: An Equivalent Porous Continuum Code for Repository Scale Analysis. *Report # PNL-5044, UC-70, Battelle Pacific Northwest Lab., Richland, WA.*
- Fujioka, M.R. and A.K. Runchal, 1983. Cooling water supply disposal in coastal aquifer. *J. Energy Eng., Vol. 109, No. 2, pp. 88-102, June 1983.*
- Kline, N.W., A.K. Runchal and R.G. Baca, 1983. PORFLO computer code: Users guide. *RHO-BW-CR-138P, Rockwell Hanford Operations, Richland, WA.*
- Runchal, A.K., 1982. PORFLOW: A mathematical model for coupled ground water flow, heat transfer and radionuclide transport in porous media. *ACRI/TN-006.*
- Runchal, A.K., 1982. PORFREEZ: A general purpose ground water flow, heat and mass transport model with freezing, thawing and surface water interaction. *ACRI/TN-005*
- Runchal, A.K., P. Bardey, B. Sagar and R.G. Baca, 1982. An equivalent continuum model for flow and transport in basalt medium: Volume I - Theory and Benchmarking. *Rockwell Hanford Operations, Basalt Waste Isolation Project, Richland, Washington.*

Sagar, B., 1982. Estimation of drawdown due to dewatering of an open pit mine. ACRi Job # 001-05-01, Kaman Tempo, Denver, CO.

Sagar, B. and A.K. Runchal, 1982. GEOFORM: A mathematical model for flow, heat transfer and mass transport in consolidating porous media under conditions of sedimentation and erosion. ACRi/REP/013 & ACRi Job # 001-03-01, Exxon Production Research Company, Houston, Texas.

Runchal, A.K., 1981. Numerical modeling of some geotechnical considerations associated with underground isolation of nuclear wastes at the Savannah River Plant, South Carolina. National Academy of Sciences, Washington, D.C., ACRi Job #008-01-01.

Runchal, A. K. and G. Hocking, 1981. An equivalent continuum model for fluid flow, heat and mass transport in geologic materials. ASME/AIChE National Heat Transfer Conf., Milwaukee, Aug. 2-5, 1981, ASME publication 81-HT-54.

Runchal, A.K., B. Sagar, J. Treger, R.G. Baca and Arnett, R.C., 1981. Theoretical analysis of waste isolation in a basalt media: an evaluation of postulated release scenarios. ACRi Job # 003-01-01, Rockwell Hanford Operations, Richland, Washington.

Runchal, A.K., 1980. A porous media fluid flow, heat & mass transport model with rock stress coupling. Proc. Workshop on Numerical Modeling of Thermohydrological Flow in Fractured Rock Masses, Lawrence Berkeley Laboratory Report# 11566.

Runchal, A.K., 1980. Mathematical modeling for ground water flow with heat transfer, freezing, thawing and atmospheric heat exchange. ACRi Job # 004-02-01, Northern Technical Services, Anchorage, AK.

Runchal, A.K., 1980. A description of panther computer program for transient analysis of heat transfer & fluid flow with chemical reaction in porous media. ACRi/REP-001, British Nuclear Fuels Limited, Risley, U.K.

Runchal, A.K. and T. Maini, 1980. The impact of a high level nuclear waste repository on the regional ground water flow. Int. J. of Rock Mechanics and Mining Science Oct.

Hocking, G. and A.K. Runchal, 1979. Numerical modeling of rock stresses within a basaltic nuclear waste repository. Dames & Moore Job #10731-001/002-86, Rockwell Hanford Operations, Richland, WA.

Runchal, A.K., 1979. Modelling of processes occurring in the magnox swarf silos. Dames & Moore Job # 11334-003-60, British Nuclear Fuels Limited, Risley, U.K.

Runchal, A.K. and T. Maini, 1979. Regional ground water flow near a high-level radioactive waste repository. Engineering Bulletin 50, Dames & Moore, December 1979, pp. 11-20.

Runchal, A.K., G.S. Segal and A. Mills, 1979. Proceedings of the workshop on ground water flow model GWTHERM. Dames & Moore ATG/TN-LA-45.

Runchal, A.K., J. Treger and G. Segal, 1979. Two-dimensional fluid flow, heat and mass transport in porous media. Dames & Moore ATG/TN-LA-34.

Runchal, A.K., 1978. Ground water movement and nuclide transport. Technical support for geis: radioactive waste isolation in geologic formations. Y/OWI/TM-36/21, U.S. Dept. of Energy Contract W-7405 Eng 26, Union Carbide Subcontract 62Y-45701C, Dames & Moore Job # 00822-107-23, April, 1978. Office of Waste Isolation, Union Carbide, Oak Ridge, TN.

Runchal, A.K. and T. Maini, 1978. The impact of a high level radioactive waste repository on the regional ground water flow. Dames & Moore ATG/TN-LA-30, October 16, 1978, Abstract Appeared in 1978 Joint Annual Meeting GSA, GAC and MAC, Toronto, Ontario, Canada, October 23-26.

Runchal, A.K. and G.S. Segal, 1978. Two-dimensional fluid flow analysis in porous media with temperature and density variations. Dames & Moore, ATG/TN-LA-36.

2 January, 1999

(This page intentionally left 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.

CONTENTS

B.1 THE FREEFORM™ INPUT RECORDS

- B.1.1 The Keyword Record
- B.1.2 The Continuation Record
- B.1.3 The Comment Record

B.2 ELEMENTS OF INPUT RECORD

- B.2.1 The Keyword
- B.2.2 The Modifier
- B.2.3 The Numeric Field
- B.2.4 The Separator Field
- B.2.5 The Terminator
- B.2.6 The Comment Field
- B.2.7 The Prompt

B.1 THE FREEFORM™ INPUT RECORDS

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

B.1.1 The Keyword Record

Function To specify the numeric and character data.

Structure

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

B.1.2 The Continuation Record

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

Structure

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

B.1.3 The Comment Record

Function To enhance the clarity and readability of the input.

Structure

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

B.2 ELEMENTS OF INPUT RECORD

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

B.2.1 *The Keyword*

Function The keyword identifies the input group.

Structure

- ◆ The keyword may consist of any characters except separator (Section B.2.4) or terminator (Section B.2.5) characters. However, the first character of a keyword must be alphabetic ('A' through 'Z' or 'a' through 'z'). To this extent, the concept of a keyword is similar to that of a variable name in FORTRAN.
- ◆ The keyword may be in upper or lower case.
- ◆ A keyword must begin in the first column of a record.
- ◆ The keyword is terminated with the first occurrence of a valid separator or terminator character.
- ◆ The keyword may consist of 1 to 256 characters. However, if there are more than four characters, only the first four are machine-identifiable.

EXAMPLES

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

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

Note that a specification of ABC) or ABC', ABC\$, although valid, is equivalent to that of ABC because the last character in all of these examples is either a separator (Section B.2.5) or a terminator (Section B.2.6).

B.2.2 The Modifier

Any character information on an input record following a keyword, except that embedded in a numeric or comment field (see Sections B.2.3 and B.2.6), is treated as modifier(s).

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

Structure

- ◆ A modifier in any input group, if present, must follow the keyword.
- ◆ The modifier is identical to the keyword in its structure. It may consist of any characters, except separator and terminator characters, of which the first character must be alphabetic.
- ◆ A modifier must not start in the first column of a record. It can be from 1 to 79 characters in length; however, if it is longer than four characters, only the first four are significant.
- ◆ The modifier must be separated from the keyword, other modifiers and numeric data by a valid separator, terminator or comment field.

EXAMPLES

The structure of a modifier is identical to that of a keyword except that it must not start in the first column of a record. Examples are given in Section B.2.1

B.2.3 The Numeric Field

Any numeric characters on a keyword or continuation record following a keyword, except those embedded in a keyword, modifier or comment field (see Sections B.2.1, B.2.2 and B.2.6), are treated as numeric data.

Function A numeric field contains numeric data for input variables.

Structure

- ◆ A numeric field is a continuous string of characters that must begin with the numeric character set. In this context, the numeric character set consists of the numerals (0-9), the decimal point (.), and the plus (+) and minus (-) operators.
- ◆ A numeric field must consist only of the numeric character set defined above, the asterisk (*), and the exponent in lower (e) or upper (E) case. It must not contain any other character.
- ◆ The plus (+) or minus (-) sign, if present, must immediately precede the numerical value without any intervening blank or other characters.
- ◆ The asterisk (*) or the exponent (E or e), if present, must be embedded; the numeric field must not begin or end with one of these characters.
- ◆ A numeric field must be separated from the keyword, modifiers and other numeric fields by a valid separator, terminator or comment field.
- ◆ Numeric field may be located anywhere on a keyword or continuation record.
- ◆ The numeric values may be specified in any of the following formats:
 - Integer, (e.g., 999),
 - Real (e.g., 999.0, 999.)
 - Exponent (e.g., 9.99E2, 99.9E1)
- ◆ Successive, repetitive, identical numeric values may be specified by the asterisk (*) option. Thus, (30., 30., 30.) may be represented as (3*30. or 3*3.0E+1); embedded separators or non-numeric characters must not appear in such specification.

EXAMPLES

The input character strings, 1, 0.1234, .567, +123., -1.0005, 1.2e00, 1.35E0, and 3*1.2 are all valid examples of a numerical field. Input specifications of 123, 123., 1.23e02, +0.123E+3, 1.23E2, 1*123, 1*1.23E02 are all equivalent.

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

Note that a specification of 1.2)2. or 1.2=2, although valid, will be equivalent to a specification of two numeric fields, 1.2 and 2 because of the embedded separator (Section B.2.4) in both cases. A specification of 1.2\$2 is equivalent to a specification of 1.2 because the 2 following the \$ will be ignored (Section B.2.5).

B.2.4 The Separator Field

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

Structure

- ◆ Any continuous 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 preferred separator characters are the comma (,), the space (), the equal sign (=), the colon (:), the semi-colon (;), the pound sign (#), the apostrophe ('), the vertical line (|), the left parenthesis "(", and the right parenthesis ")" characters.

EXAMPLES

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

B.2.5 The Terminator

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

Structure

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

EXAMPLES

The character sequences:

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

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

B.2.6 The Comment Field

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

Structure

- ◆ A comment field may be in the form of an embedded comment or a comment record.
- ◆ An embedded comment field is one which occurs on a keyword or continuation record. It must begin with a terminator (\$ or !) character; any combination of characters may follow the terminator. The comment field is terminated at the end of the 256th character in that record (Section B.2.5).
- ◆ A comment field on a comment record may consist of any combination of characters. In this case, the comment field begins with the back slash (/), asterisk (*), or a terminator (\$ or !) character as the first character and terminates with the 256th character.

EXAMPLES

In the input record:

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

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

As examples of comment records, the character strings:

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

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

B.2.7 The Prompt

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

Structure

- ◆ The user may insert a '?' in place of either an entire input record or numerical or character string in the input itself. The command interpreter will pause and prompt the user for input at this stage.
- ◆ The input obtained by the prompt may be comprised of one or more of the elements of an input record described in Sections B.2.1 through B.2.6.
- ◆ Any input specified by the prompt is inserted in the input command string starting with the location of the question mark symbol.
- ◆ All input in the original command string following the prompt in that record will be disregarded.
- ◆ The input for the prompt may consist of 1 to 256 characters depending on where the question mark symbol is located. The prompt will pause and display a message telling the user the maximum number of characters allowed.
- ◆ It is possible to avoid the message, referred to above, from being displayed by placing two question mark symbols in succession (eg. '??'). The command interpreter will still pause and wait for input, however it will not display a message.

EXAMPLES

In the input record:

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

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

Alternatively the user may choose to write:

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

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

(This page left intentionally blank.)

APPENDIX C

EXAMPLES OF PORFLOW™ INPUT AND OUTPUT

Illustrative examples of PORFLOW™ input command files, and output obtained from them, are supplied under separate cover.

(This page left intentionally blank.)