USGS
science for a changing world

Rec'd 2/3/00 (Kim/GHG)

User's Guide to PHREEQC (Version 2)—A Computer Program for Speciation, Batch-Reaction, One-Dimensional Transport, and Inverse Geochemical Calculations

Water-Resources Investigations Report 99-4259

U.S. DEPARTMENT OF THE INTERIOR
U.S. GEOLOGICAL SURVEY

U.S. DEPARTMENT OF THE INTERIOR
U.S. GEOLOGICAL SURVEY

USER'S GUIDE TO PHREEQC (VERSION 2)— A COMPUTER PROGRAM FOR SPECIATION, BATCH-REACTION, ONE-DIMENSIONAL TRANSPORT, AND INVERSE GEOCHEMICAL CALCULATIONS

By **David L. Parkhurst *and* C.A.J. Appelo**[1]

Water-Resources Investigations Report 99-4259

[1]Hydrochemical Consultant
Valeriusstraat 11
1071 MB Amsterdam, NL
appt@xs4all.nl
http://www.xs4all.nl/~appt/index.html

Denver, Colorado
1999

U.S. DEPARTMENT OF THE INTERIOR

BRUCE BABBITT, Secretary

U.S. GEOLOGICAL SURVEY

Charles G. Groat, Director

For additional information write to:

Chief, Branch of Regional Research
U.S. Geological Survey
Box 25046, MS 418
Denver Federal Center
Denver, CO 80225

Copies of this report can be purchased from:

U.S. Geological Survey
Earth Science Information Center
Open-File Reports Section
Box 25286, MS 517
Denver Federal Center
Denver, CO 80225

CONTENTS

FIGURES

TABLES

ABBREVIATIONS OF UNITS

The following abbreviations are used in this report:

atmosphere	atm
calorie	cal
Coulomb	C
degree Celsius	°C
equivalent	eq
gram	g
Joule	J
degree Kelvin	K
kilocalorie	kcal
kilogram	kg
kilogram of water	kgw
kilojoule	kJ
liter	L
meter	m
milliequivalent	meq
millimole	mmol
micromole	µmol
mole	mol
parts per million	ppm
parts per billion	ppb
square meter	m^2
Volt	V

Degree Celsius (°C) may be converted to degree Fahrenheit (°F) by using the following equation:

$$°F = 9/5\ (°C) + 32.$$

Degree Fahrenheit (°F) may be converted to degree Celsius (°C) by using the following equation:

$$°C = 5/9\ (°F-32).$$

Other conversions:

Absolute temperature °K = 273.15 + (°C).

1 joule = 0.239 calorie

Some constants:

Avogadro's comstant *N*	6.022e23 /mol
Faraday *F*	96,485 C/mol
Gas constant *R*	8.314 J/mol/°K
Molar volume, ideal gas, 1 atm, 25°C	24.465 dm^3/mol

User's Guide to PHREEQC (Version 2)—A Computer Program for Speciation, Batch-Reaction, One-Dimensional Transport, and Inverse Geochemical Calculations

By David L. Parkhurst *and* C.A.J. Appelo

Abstract

PHREEQC version 2 is a computer program written in the C programming language that is designed to perform a wide variety of low-temperature aqueous geochemical calculations. PHREEQC is based on an ion-association aqueous model and has capabilities for (1) speciation and saturation-index calculations; (2) batch-reaction and one-dimensional (1D) transport calculations involving reversible reactions, which include aqueous, mineral, gas, solid-solution, surface-complexation, and ion-exchange equilibria, and irreversible reactions, which include specified mole transfers of reactants, kinetically controlled reactions, mixing of solutions, and temperature changes; and (3) inverse modeling, which finds sets of mineral and gas mole transfers that account for differences in composition between waters, within specified compositional uncertainty limits.

New features in PHREEQC version 2 relative to version 1 include capabilities to simulate dispersion (or diffusion) and stagnant zones in 1D-transport calculations, to model kinetic reactions with user-defined rate expressions, to model the formation or dissolution of ideal, multicomponent or nonideal, binary solid solutions, to model fixed-volume gas phases in addition to fixed-pressure gas phases, to allow the number of surface or exchange sites to vary with the dissolution or precipitation of minerals or kinetic reactants, to include isotope mole balances in inverse modeling calculations, to automatically use multiple sets of convergence parameters, to print user-defined quantities to the primary output file and (or) to a file suitable for importation into a spreadsheet, and to define solution compositions in a format more compatible with spreadsheet programs.

This report presents the equations that are the basis for chemical equilibrium, kinetic, transport, and inverse-modeling calculations in PHREEQC; describes the input for the program; and presents examples that demonstrate most of the program's capabilities.

INTRODUCTION

PHREEQC version 2 is a computer program for simulating chemical reactions and transport processes in natural or polluted water. The program is based on equilibrium chemistry of aqueous solutions interacting with minerals, gases, solid solutions, exchangers, and sorption surfaces, but also includes the capability to model kinetic reactions with rate equations that are completely user-specified in the form of Basic statements. Kinetic and equilibrium reactants can be interconnected, for example by linking the number of surface sites to the amount of a kinetic reactant that is consumed (or produced) during the course of a model period. A 1D transport algorithm comprises dispersion, diffusion, and various options for dual porosity media. A powerful inverse modeling capability allows

identification of reactions that account for observed water compositions along a flowline or in the time course of an experiment. An extensible chemical data base allows application of the reaction, transport, and inverse-modeling capabilities to almost any chemical reaction that is recognized to influence rain-, soil-, ground- and surface water quality.

PHREEQC is based on the Fortran program PHREEQE (Parkhurst and others, 1980). PHREEQE was capable of simulating a variety of geochemical reactions for a system including:

- Mixing of waters,
- Addition of net irreversible reactions to solution,
- Dissolving and precipitating phases to achieve equilibrium with the aqueous phase, and
- Effects of changing temperature.

PHREEQE calculated concentrations of elements, molalities and activities of aqueous species, pH, pe, saturation indices, and mole transfers of phases to achieve equilibrium as a function of specified reversible and irreversible geochemical reactions.

PHREEQC version 1 (Parkhurst, 1995) was a completely new program written in the C programming language that implemented all of the capabilities of PHREEQE and added many capabilities that were not available in PHREEQE, including:

- Ion-exchange equilibria,
- Surface-complexation equilibria,
- Fixed-pressure gas-phase equilibria, and
- Advective transport.

Other improvements relative to PHREEQE included complete accounting for elements in solids and the aqueous and gas phase, mole balance on hydrogen and oxygen to account for the mass of water in the aqueous phase, identification of the stable phase assemblage from a list of candidate phases, use of redox couples for definition of redox state in speciation calculations, and a more robust non-linear equation solver.

PHREEQC version 2 is a modification of PHREEQC version 1. All of the capabilities and most of the code for version 1 are retained in version 2 and several new capabilities have been added, including:

- Kinetically controlled reactions,
- Solid-solution equilibria,
- Fixed-volume gas-phase equilibria,
- Variation of the number of exchange or surface sites in proportion to a mineral or kinetic reactant,
- Diffusion or dispersion in 1D transport,
- 1D transport coupled with diffusion into stagnant zones, and
- Isotope mole balance in inverse modeling.

The numerical method has been modified to use several sets of convergence parameters in an attempt to avoid convergence problems. User-defined quantities can be written to the primary output file and (or) to a file suitable for importation into a spreadsheet, and solution compositions can be defined in a format that is more compatible with spreadsheet programs.

Program Capabilities

PHREEQC can be used as a speciation program to calculate saturation indices and the distribution of aqueous species. Analytical data for mole balances can be defined for any valence state or combination of valence states for an element. Distribution of redox elements among their valence states can be based on a specified pe or any redox couple for which data are available. PHREEQC allows the concentration of an element to be adjusted to obtain equilibrium (or a specified saturation index or gas partial pressure) with a specified phase. Solution compositions can be specified with a variety of concentration units.

In batch-reaction calculations, PHREEQC is oriented toward system equilibrium rather than just aqueous equilibrium. For a purely equilibrium calculation, all of the moles of each element in the system are distributed among the aqueous phase, pure phases, solid solutions, gas phase, exchange sites, and surface sites to attain system equilibrium. Non-equilibrium reactions can also be modeled, including aqueous-phase mixing, user-specified changes in the elemental totals of the system, kinetically controlled solid-liquid heterogeneous reactions, and to a limited extent kinetically controlled aqueous homogeneous reactions. Mole balances on hydrogen and oxygen allow the calculation of pe and the mass of water in the aqueous phase, which allows water-producing or -consuming reactions to be modeled correctly.

The generalized two-layer model of Dzombak and Morel (1990), a model with an explicitly calculated diffuse layer (Borkovec and Westall, 1983), and a non-electrostatic model (Davis and Kent, 1990) have been incorporated for modeling surface-complexation reactions. Surface complexation constants for two of the databases distributed with the program (*phreeqc.dat* and *wateq4f.dat*) are taken from Dzombak and Morel (1990); surface complexation constants for the other database distributed with the program (*minteq.dat*) are taken from MINTEQA2 (Allison and others, 1990). Ion-exchange reactions are modeled with the Gaines-Thomas convention and equilibrium constants derived from Appelo and Postma (1993) are included in two of the databases distributed with the program (*phreeqc.dat* and *wateq4f.dat*).

New modeling capabilities in version 2 include kinetically controlled reactions, solid solutions, and fixed-volume gases. Kinetically controlled reactions can be defined in a general way by using an embedded Basic interpreter. Rate expressions written in the Basic language are included in the input file, and the program uses the Basic interpreter to calculate rates. Formulations for ideal, multicomponent and nonideal, binary solid solutions have been added. The program is capable of determining the equilibrium compositions of nonideal, binary solid solutions even if miscibility gaps exist and of determining the equilibrium composition of ideal solid solutions that have two or more components. It is possible to precipitate solid solutions from supersaturated conditions with no pre-existing solid, and to dissolve solid solutions completely. In addition to the fixed-pressure gas phase of version 1 (fixed-pressure gas bubbles), version 2 allows for a fixed-volume gas phase.

It is possible to define independently any number of solution compositions, gas phases, or pure-phase, solid-solution, exchange, or surface-complexation assemblages. Batch reactions allow any combination of solution (or mixture of solutions), gas phase, and assemblages to be brought together, any irreversible reactions are added, and the resulting system is brought to equilibrium. If kinetic reactions are defined, then the kinetic reactions are integrated with an automatic time-stepping algorithm and system equilibrium is calculated after each time step.

The capability to define multiple solutions and multiple assemblages combined with the capability to determine the stable phase assemblage, provides a framework for 1D transport modeling. PHREEQC provides a numerically efficient method for simulating the movement of solutions through a column or 1D flow path with or

without the effects of dispersion. The initial composition of the aqueous, gas, and solid phases within the column are specified and the changes in composition due to advection and dispersion (Appelo and Postma, 1993) coupled with reversible and irreversible chemical reactions within the column can be modeled. A very simple advective-reactive transport simulation option with reversible and irreversible chemical reactions is retained from version 1.

Inverse modeling attempts to account for the chemical changes that occur as a water evolves along a flow path. Assuming two water analyses which represent starting and ending water compositions along a flow path, inverse modeling is used to calculate the moles of minerals and gases that must enter or leave solution to account for the differences in composition. Inverse models that mix two or more waters to form a final water can also be calculated. PHREEQC allows uncertainty limits to be defined for all analytical data, such that inverse models are constrained to satisfy mole balance for each element and valence state as well as charge balance for each solution, but only within these specified uncertainty limits. With version 2, isotope mole-balance equations with associated uncertainty limits can be specified, but Rayleigh fractionation processes can not be modeled explicitly.

The input to PHREEQC is completely free format and is based on chemical symbolism. Balanced equations, written in chemical symbols, are used to define aqueous species, exchange species, surface-complexation species, solid solutions, and pure phases, which eliminates all use of index numbers to identify elements or species. At present, a graphical user interface is available for version 1 (Charlton and others, 1997), and a graphical user interface with charting options has been released for version 2 (PHREEQC for Windows, V.E.A. Post, written commun., 1999, *http://www.geo.vu.nl/users/posv/phreeqc.html*). The free-format structure of the data, the use of order-independent keyword data blocks, and the relatively simple syntax facilitate the generation of input data sets with a standard editor. The C programing language allows dynamic allocation of computer memory, so there are very few limitations on array sizes, string lengths, or numbers of entities, such as solutions, phases, sets of phases, exchangers, solid solutions, or surfaces that can be defined to the program.

Program Limitations

PHREEQC is a general geochemical program and is applicable to many hydrogeochemical environments. However, several limitations need to be considered.

Aqueous Model

PHREEQC uses ion-association and Debye Hückel expressions to account for the non-ideality of aqueous solutions. This type of aqueous model is adequate at low ionic strength but may break down at higher ionic strengths (in the range of seawater and above). An attempt has been made to extend the range of applicability of the aqueous model through the use of an ionic-strength term in the Debye Hückel expressions. These terms have been fit for the major ions using chloride mean-salt activity-coefficient data (Truesdell and Jones, 1974). Thus, in sodium chloride dominated systems, the model may be reliable at higher ionic strengths. For high ionic strength waters, the specific interaction approach to thermodynamic properties of aqueous solutions should be used (for example, Pitzer, 1979, Harvie and Weare, 1980, Harvie and others, 1984, Plummer and others, 1988), but this approach is not incorporated in PHREEQC.

The other limitation of the aqueous model is lack of internal consistency in the data in the databases. Two of the databases distributed with the code, *phreeqc.dat* and *wateq4f.dat*, are consistent with the aqueous model of WATEQ4F (Ball and Nordstrom, 1991) and the compilation of Nordstrom and others (1990), the other database,

minteq.dat, is taken from MINTEQA2 (Allison and others, 1990). However, in these compendia, the log K's and enthalpies of reaction have been taken from various literature sources. No systematic attempt has been made to determine the aqueous model that was used to develop the individual log K's or whether the aqueous models defined by the current database files are consistent with the original experimental data. The database files provided with the program should be considered to be preliminary. Careful selection of aqueous species and thermodynamic data is left to the users of the program.

Ion Exchange

The ion-exchange model assumes that the thermodynamic activity of an exchange species is equal to its equivalent fraction. Optionally, the equivalent fraction can be multiplied by a Debye-Hückel activity coefficient to define the activity of an exchange species (Appelo, 1994a). Other formulations use other definitions of activity (mole fraction instead of equivalent fraction, for example) and may be included in the database with appropriate rewriting of species or solid solutions. No attempt has been made to include other or more complicated exchange models. In many field studies, ion-exchange modeling requires experimental data on material from the study site for appropriate model application.

Surface Complexation

PHREEQC incorporates the Dzombak and Morel (1990) generalized two-layer model, a two-layer model that explicitly calculates the diffuse-layer composition (Borkovec and Westall, 1983), and a non-electrostatic surface-complexation model (Davis and Kent, 1990). Other models, including triple- and quadruple-layer models have not been implemented in PHREEQC. Sorption according to Langmuir or Freundlich isotherms can be modeled as special cases of the non-electrostatic model.

Davis and Kent (1990) reviewed surface-complexation modeling and note theoretical problems with the use of molarity as the standard state for sorbed species. PHREEQC version 2 uses mole fraction for the activity of surface species instead of molarity. This change in standard state has no effect on monodentate surface species, but does affect multidentate species significantly. Other uncertainties occur in determining the number of sites, the surface area, the composition of sorbed species, and the appropriate log K's. In many field studies, surface-complexation modeling requires experimental data on material from the study site for appropriate model application.

The capability of PHREEQC to calculate explicitly the composition of the diffuse layer (**-diffuse_layer** option **SURFACE** data block) is *ad hoc* and should be used only as a preliminary sensitivity analysis.

Solid Solutions

PHREEQC uses a Guggenheim approach for determining activities of components in nonideal, binary solid solutions (Glynn and Reardon, 1990). Ternary nonideal solid solutions are not implemented. It is possible to model two or more component solid solutions by assuming ideality. However, the assumption of ideality is usually an oversimplification except possibly for isotopes of the same element.

Transport Modeling

An explicit finite difference algorithm is included for calculations of 1D advective-dispersive transport and optionally diffusion in stagnant zones. The algorithm may show numerical dispersion when the grid is coarse. The magnitude of numerical dispersion also depends on the nature of the modeled reactions; numerical dispersion may be large in the many cases--linear exchange, surface complexation, diffusion into stagnant zones, among others--but may be small when chemical reactions counteract the effects of dispersion. It is recommended that modeling be performed stepwise, starting with a coarse grid to obtain results rapidly and to investigate the hydrochemical reactions, and finishing with a finer grid to assess the effects of numerical dispersion on both reactive and conservative species.

Convergence Problems

PHREEQC tries to identify input errors, but it is not capable of detecting some physical impossibilities in the chemical system that is modeled. For example, PHREEQC allows a solution to be charge balanced by addition or removal of an element. If this element has no charged species or if charge imbalance remains even after the concentration of the element has been reduced to zero, then the numerical method will fail to converge. Other physical impossibilities that have been encountered are (1) when a base is added to attain a fixed pH, but in fact an acid is needed (or *vice versa*) and (2) when noncarbonate alkalinity exceeds the total alkalinity given on input.

At present, the numerical method has proved to be relatively robust. All known convergence problems--cases when the numerical method fails to find a solution to the non-linear algebraic equations--have been resolved. Occasionally it has been necessary to use the scaling features of the **KNOBS** keyword. The scaling features appear to be necessary when total dissolved concentrations fall below approximately 10^{-15} mol/kgw (moles per kilogram of water).

Inverse Modeling

Inclusion of uncertainties in the process of identifying inverse models is a major advance over previous inverse modeling programs. However, the numerical method has shown some inconsistencies in results due to the way the solver handles small numbers. The option to change the tolerance used by the solver (**-tol** in **INVERSE_MODELING** data block) is an attempt to remedy this problem. In addition, the inability to make Rayleigh fractionation calculations for isotopes in precipitating minerals is a limitation.

How to Obtain the Software and Manual

Win32 and Unix versions of the software described in this report, including this manual in Postscript and PDF formats, can be obtained from the web site *http://water.usgs.gov/software/*. The most current version and additional bibliographies and information files are available at *http://wwwbrr.cr.usgs.gov/projects/GWC_coupled.*

An interactive user interface for Windows can be obtained from the web site *http://www.geo.vu.nl/users/posv/phreeqc.html.* An expanded interactive user interface for Windows is currently under development (1999) and will be made available through the web sites noted in the previous paragraph.

Win32 and Unix versions of PHREEQC, may be obtained by anonymous ftp from the Internet address: brrcrftp.cr.usgs.GOV. For ftp access, the files reside in directories */geochem/pc/phreeqc* and

/geochem/unix/phreeqc. Be sure to use "type binary" for transferring the tar file by ftp. A typical anonymous ftp session follows:

```
% ftp brrcrftp.cr.usgs.GOV
Name: anonymous
Password: userid@computer          (replaced with your userid and computer name)
ftp> cd geochem/pc/phreeqc         (change directory)
ftp> ls                            (list files in directory)
phrqc2xx.exe                       (xx represents version number)
ftp> type binary                   (eliminate any ascii translation for binary files)
ftp> get phrqc2xx.exe              (transfer the file, xx represents version number)
ftp> quit                          (quit ftp)
```

Alternatively, the documentation and Win32 or Unix versions of the software can be ordered from the following address:

U.S. Geological Survey
NWIS Program Office
437 National Center
Reston, VA 22092
(703) 648-5695

Additional copies of this report are available from:

U.S. Geological Survey
Branch of Information Services
Box 25286
Denver Federal Center
Denver, CO 80225-0286

For additional information, write to the address on page ii of this report.

Installation and Execution of the Win32 Version

The Win32 version of PHREEQC requires Windows 9x or Windows NT. The installation file *phrqc2xx.exe* (where *xx* is the version number), obtained by downloading from a web page or anonymous ftp, will install the necessary files to run PHREEQC. Optionally, the installation program will modify the user's path to allow PHREEQC to run from any directory. The installation will install all of the files of the program distribution into a user-supplied directory name (default *c:\usgs\phreeqc*). The executable file (*phreeqc.exe*) will be installed in the subdirectory *src\Release* relative to the installation directory. A batch file (*phreeqc.bat*) will be installed in the installation directory along with the database files (*phreeqc.dat*, *wateq4f.dat*, and *minteq.dat*). The source code is installed in the subdirectory *src* relative to the installation directory. An example file for each problem described in the "Examples" section of this manual is copied into the subdirectory *examples* relative to the installation directory.

The version of PHREEQC described here is a batch-oriented program that requires an input file that describes the calculations to be made, an output file name to store results, and a database file. To run any of the input files in the *examples* subdirectory, change directory to the *examples* directory in a DOS window. Note that example 14 requires the database file *wateq4f.dat*, which is in the installation directory, and example 15 requires the database file *ex15.dat*, which is in the *examples* subdirectory. All other examples can be run with the database file *phreeqc.dat*. Invoke PHREEQC with any of the following commands.

phreeqc	(The program will query for each of the needed files.)
phreeqc *input*	(The input file is named *input*, the output file will be named *input.out* and the default database file will be used.)
phreeqc *input output*	(The input file is named *input*, the output file is named *output*, and the default database file will be used.)
phreeqc *input output database*	(All file names are specified explicitly.)
phreeqc *input output database screen_output*	(All file names are specified explicitly, and screen output is directed to the file *screen_output*.)

The environmental variable **PHREEQC_DATABASE** can be used to specify the default database for a DOS window session. This environmental variable can be set with the command:

set PHREEQC_DATABASE=*c:\mydirectory\myproject\mydata.dat*

If the environmental variable is not set, the default database file is *phreeqc.dat* in the installation directory. If PHREEQC is invoked with at least three arguments, the third argument is the database file and it supersedes any of the default databases. When specifying the database file, it may be necessary to give a relative pathname. For instance, example 14 requires the database *wateq4f.dat* and this example could be run from the *examples* subdirectory with the following command:

phreeqc ex14 ex14.out ..\wateq4f.dat

The results of this calculation will be found in the file *ex14.out* in the *examples* directory.

Installation and Execution of the Unix Version

The Unix source code is almost identical to the Win32 source code, the only difference being one statement in the file *global.h* ("#define DOS" is absent). A script to run the program in batch mode and a makefile are included in the Unix distribution. The following steps should be used to transfer, compile, and install the program on a Unix computer. Compiled executables are available at the web sites for SunOS and Linux operating systems, which eliminates the need for the compilation step on these operating systems.

(1) Transfer the compressed tar files to your home computer with a browser or by ftp. Be sure to use "type binary" for transferring the tar file by ftp.

(2) Uncompress the compressed tar file and extract the files with tar. The files will automatically extract into subdirectories named *bin*, *database*, *doc*, *src*, and *examples*. Here, "2.x" represents a version number.

% **uncompress phreeqc.2.x.tar.Z**

% **tar -xvof phreeqc.2.x.tar**

(3) Versions for Linux and Sun are available with precompiled executables. If the program is to be used on another type of computer, then change directory into *src* and compile the programs using make. By default the makefile (named *src/Makefile*) uses gcc as the compiler. Change the variables "CC" and "CCFLAGS" in the makefile to be consistent with the C compiler on your system if necessary. The following commands will create an executable file named, *../bin/phreeqc*.

% **cd src**

% **make**

(4) Install the script to run PHREEQC. The makefile edits a template of the script (*bin/phreeqc.orig*) to contain the complete pathname to the installation directory and places the edited script in the installation directory. A symbolic link that points to the script is then placed in a directory specified by the user (frequently */usr/bin*). The directory in which the symbolic link is installed is assumed to be included in your PATH environmental variable, so that the PHREEQC will run regardless of the directory from which it is invoked. The default directory in which the symbolic link is installed is *$(HOME)/bin*.

The following command installs a symbolic link in *$(HOME)/bin*:

% **make install**

The following command installs the script in the specified directory:

% **make install BINDIR=***/usr/local/bin*

(5) The environmental variable **PHREEQC_DATABASE** can be used to specify the default database. In the shell *csh*, this variable can be set with the command:

% **setenv PHREEQC_DATABASE** */home/jdoe/local/project/mydata.dat*

In the Bourne or Korn shell, this variable can be set with the command:

% **export PHREEQC_DATABASE=***/home/jdoe/local/project/mydata.dat*

The environmental variable can be set permanently by including the appropriate command in a file that is read when the shell is initiated, frequently *$(HOME)/.login* or *$(HOME)/.profile*. If this environmental variable is not set, the default database is set in the script (in the installation directory) to *database/phreeqc.dat* relative to the installation directory. It is possible to specify a different default database by editing the script.

After PHREEQC is installed, it can be executed from any directory with any of the commands described in the Win32 installation section with the understanding that Unix is case sensitive and that most Unix commands and file names are lower case. The examples from this manual can be run from the subdirectory *examples* in the installation directory. Note that example 14 requires the database file *wateq4f.dat,* which is *database/wateq4f.dat* in the installation directory, and example 15 requires the database file *ex15.dat*, which is in the *examples* subdirectory.

Purpose and Scope

The purpose of this report is to describe the theory and operation of the program PHREEQC. The report includes the definition of the constituent equations, explanation of the transformation of these equations into a numerical method, description of the organization of the computer code that implements the numerical method, description of the input for the program, and presentation of a series of examples of input data sets and model results that demonstrate many of the capabilities of the program.

EQUATIONS FOR SPECIATION AND FORWARD MODELING

In this section of the report, the algebraic equations used to define thermodynamic activities of aqueous species, ion-exchange species, surface-complexation species, gas-phase components, solid solutions, and pure phases are presented. First, thermodynamic activities and mass-action equations are described for aqueous, exchange, and surface species. Then, a set of functions, denoted f, are defined that must be solved simultaneously to determine equilibrium for a given set of conditions. Many of these functions are derived from mole-balance equations for each element or element valence state, exchange site, and surface site or from mass-action equations

for pure phases and solid solutions. Additional functions are derived for alkalinity, activity of water, aqueous charge balance, gas-phase equilibria, ionic strength, and surface-complexation equilibria. Each function is reduced to contain a minimum number of variables, such that the number of functions equals the number of variables. The program uses a modified Newton-Raphson method to solve the simultaneous nonlinear equations. This method uses the residuals of the functions and an array of partial derivatives of each function with respect to the set of master unknowns or master unknowns. For clarity, the set of variables used in partial differentiation are referred to as "master unknowns". The total derivatives of each function, f, will be presented without derivation. In the following equations, lack of a subscript or the subscript "(aq)" will refer to entities in the aqueous phase, "(e)" refers to exchangers, "(g)" refers to gases, "(s)" refers to surfaces, "(ss)" refers to solid solutions, and "(p)" refers to phases.

Activities and Mass-Action Equations

In this section the activities of aqueous, exchange, and surface species are defined and the mass-action relations for each species are presented. Equations are derived from the mass-action expression for the moles of each species in the chemical system in terms of the master unknowns. These equations are then differentiated with respect to the master unknowns. Later, these equations for the moles of a species and the partial derivatives will be substituted into the constituent mole-balance, charge-balance, and phase-equilibria functions.

Aqueous Species

PHREEQC allows speciation or equilibration with respect to a single aqueous phase. However, multiple aqueous phases may be defined in the course of a run and an aqueous phase may be defined as a mixture of one or more aqueous phases (see **MIX** keyword in "Description of Data Input"). The dissolved species in the aqueous phase are assumed to be in thermodynamic equilibrium, with one exception; in initial solution calculations, disequilibrium among valence states of redox elements is allowed. The unknowns for each aqueous species i are the activity, a_i, activity coefficient, γ_i, molality, m_i, and moles in solution, n_i.

PHREEQC rewrites all chemical equations in terms of master species. There is one master aqueous species associated with each element (for example, Ca^{+2} for calcium) or element valence state (for example, Fe^{+3} for ferric iron) plus the activity of the hydrogen ion, the activity of the aqueous electron, and the activity of water. Some programs, for example MINTEQA2 (Allison and others, 1990) and MINEQL$^+$ (Schecher and McAvoy, 1991) use the term "component" for these species, but that terminology is not used here because of confusion with the definition of component for the Gibbs' phase rule. For PHREEQC, the identity of each aqueous master species is defined with **SOLUTION_MASTER_SPECIES** data block (see "Description of Data Input"). The numerical method reduces the number of unknowns to be a minimum number of master unknowns, and iteratively refines the values of these master unknowns until a solution to the set of algebraic equations is found. The master unknowns for aqueous solutions are the natural log of the activities of master species, the natural log of the activity of water, a_{H_2O}, the ionic strength, μ, and the mass of solvent water in an aqueous solution, W_{aq}.

The following relationships apply to all aqueous species (except aqueous electrons and water itself): $a_i = \gamma_i m_i$ and $n_i = m_i W_{aq}$. Equilibrium among aqueous species in an ion-association model requires that all mass-action equations for aqueous species are satisfied. For example, the association reaction for the aqueous

species $CaSO_4^0$ is $Ca^{2+} + SO_4^{2-} = CaSO_4^0$. The log K for this reaction at 25°C is 2.3, which results in the mass-action equation:

$$10^{2.3} = \frac{a_{CaSO_4^0}}{a_{Ca^{2+}} a_{SO_4^{2-}}}. \tag{1}$$

In general, mass-action equations can be written as

$$K_i = a_i \prod_m^{M_{aq}} a_m^{-c_{m,i}}, \tag{2}$$

where K_i is a temperature-dependent equilibrium constant, $c_{m,i}$ is the stoichiometric coefficient of master species m in species i and M_{aq} is the total number of aqueous master species. The values of $c_{m,i}$ may be positive or negative. For PHREEQC, terms on the right-hand side of an association reaction are assigned negative coefficients and terms on the left-hand side are assigned positive coefficients. The same formalism applies to master species, where the mass-action equation is simply $1 = \frac{a_m}{a_m}$.

The total moles of an aqueous species i can be derived from the mass-action expression:

$$n_i = m_i W_{aq} = K_i W_{aq} \frac{\prod_m^{M_{aq}} a_m^{c_{m,i}}}{\gamma_i}. \tag{3}$$

The Newton-Raphson method uses the total derivative of moles with respect to the master unknowns. The total derivative is

$$dn_i = n_i \left[d\ln(W_{aq}) + \sum_m^{M_{aq}} c_{m,i} d\ln(a_m) - \frac{\partial}{\partial \mu} \ln(\gamma_i) d\mu \right]. \tag{4}$$

Activity coefficients of aqueous species are defined with the Davies equation:

$$\log \gamma_i = -A z_i^2 \left(\frac{\sqrt{\mu}}{1 + \sqrt{\mu}} - 0.3\mu \right), \tag{5}$$

or the extended or WATEQ Debye-Hückel equation:

$$\log \gamma_i = -\frac{A z_i^2 \sqrt{\mu}}{1 + B a_i^o \sqrt{\mu}} + b_i \mu, \tag{6}$$

where z_i is the ionic charge of aqueous species i, and A and B are constants dependent only on temperature. Equation 6 is the extended Debye-Hückel equation, if b_i is zero, or the WATEQ Debye-Hückel equation (see Truesdell and Jones, 1974), if b_i is not equal to zero. In the extended Debye-Hückel equation, a_i^o is the ion-size parameter, whereas in the WATEQ Debye-Hückel equation a_i^o and b_i are ion-specific parameters fitted from mean-salt activity-coefficient data. Unless otherwise specified in the database file or the input data set, the Davies equation is used for charged species. For uncharged species, the first term of the activity coefficient equation is zero, and the WATEQ Debye-Hückel equation reduces to the Setchenow equation ($\ln \gamma_i = b_i \mu$) (see Langmuir, 1997 for discussion). Unless otherwise specified, b_i is assumed to be 0.1 for all uncharged species.

The partial derivatives of these activity coefficient equations with respect to ionic strength are

$$\frac{\partial}{\partial \mu} \ln \gamma_i = -\ln(10)\left[A z_i^2 \left(\frac{1}{2\sqrt{\mu}(\sqrt{\mu}+1)^2} - 0.3 \right) \right], \quad (7)$$

for the Davies equation and

$$\frac{\partial}{\partial \mu} \ln \gamma_i = -\ln(10)\left(\frac{A z_i^2}{2\sqrt{\mu}(B a_i^o \sqrt{\mu}+1)^2} + b_i \right), \quad (8)$$

for the extended or WATEQ Debye-Hückel equation.

For data input to PHREEQC, the chemical equation for the mole-balance and mass-action expressions, the log K and its temperature dependence, and the activity coefficient parameters for each aqueous species are defined through the **SOLUTION_SPECIES** data block. Master species for elements and element valence states are defined with the **SOLUTION_MASTER_SPECIES** data block. Composition of a solution is defined with the **SOLUTION** or **SOLUTION_SPREAD** data block (see "Description of Data Input").

Exchange Species

Ion-exchange equilibria are included in the model through heterogeneous mass-action equations and mole-balance equations for exchange sites. PHREEQC allows multiple exchangers, termed an "exchange assemblage", to exist in equilibrium with the aqueous phase. The approach uses mass-action expressions based on half-reactions between aqueous species and a fictive unoccupied exchange site (Appelo and Postma, 1993) for each exchanger. This unoccupied exchange site is the master species for the exchanger and the log of its activity is an additional master unknown. Its identity is defined with **EXCHANGE_MASTER_SPECIES** data block (see "Description of Data Input"). However, the master species is not included in the mole-balance equation for the exchanger, forcing its physical concentration to be zero. Its activity is also physically meaningless, but is such that all of the exchange sites are filled by other exchange species.

The unknowns for exchange calculations are the activity, a_{i_e}, which is defined to be the equivalent fraction in PHREEQC times an activity coefficient, γ_{i_e}, and the moles, n_{i_e}, of each exchange species, i_e, of exchanger e. The equivalent fraction is the moles of sites occupied by an exchange species divided by the total number of exchange sites. The activity of an exchange species is $a_{i_e} = \gamma_{i_e} \frac{b_{e,i_e} n_{i_e}}{T_e}$, where b_{e,i_e} is the number of equivalents of exchanger, e, occupied by the exchange species i_e, and T_e is the total number of exchange sites for the exchanger, in equivalents. Note that T_e is the total number of equivalents of the exchanger in the system, which is not necessarily equal to the number of equivalents per kilogram of water (eq/kgw) because the mass of water in the system may be more or less than 1 kg. By default, the activity coefficient for an exchange species is 1.0, but optionally, a Davies, extended Debye-Hückel, or WATEQ Debye-Hückel activity coefficient can be used, which is based on the aqueous ionic strength and the number of equivalents of exchange sites occupied by the exchange species.

Equilibrium among aqueous and exchange species requires that all mass-action equations for the exchange species are satisfied. The association reaction for the exchange species CaX_2 is $Ca^{2+} + 2X^- = CaX_2$, where X^- is the exchange master species for the default database. The use of equivalent fractions for activities and this form for the chemical reaction is known as the Gaines-Thomas convention (Gaines and Thomas, 1953) and is the

convention used in the databases *phreeqc.dat* and *wateq4f.dat*, which are distributed with PHREEQC. [It is also possible to use the Gapon convention in PHREEQC, which also uses equivalent fraction, but writes the exchange reaction as $0.5Ca^{2+} + X^- = Ca_{0.5}X$. See Appelo and Postma (1993) for more discussion.] The log K for calcium exchange in the default database file is 0.8, which results in the following mass-action equation:

$$10^{0.8} = \frac{a_{CaX_2}}{a_{Ca^{2+}} a_{X^-}^2}. \tag{9}$$

In general, mass-action equations can be written as

$$K_{i_e} = a_{i_e} \prod_m^M a_m^{-c_{m,i_e}}, \tag{10}$$

where m varies over all master species, including exchange master species, c_{m,i_e} is the stoichiometric coefficient of master species, m, in the association half-reaction for exchange species i_e, and K_{i_e} is a half-reaction selectivity constant. The values of c_{m,i_e} may be positive or negative. For PHREEQC, terms on the right-hand side of an association reaction are assigned negative coefficients and terms on the left-hand side are assigned positive coefficients.

For an exchange species, the equation for the total moles of species i_e is

$$n_{i_e} = K_{i_e} \frac{\prod_m^M a_m^{c_{m,i_e}}}{\gamma_{i_e}\left(\frac{b_{e,i_e}}{T_e}\right)}. \tag{11}$$

The natural log of the activity of the master species of the exchanger is a master unknown in the numerical method. The total derivative of the moles of species i_e with respect to the master unknowns is

$$dn_{i_e} = n_{i_e}\left(\sum_m^M c_{m,i_e} d\ln(a_m) - \frac{\partial}{\partial \mu}\ln(\gamma_{i_e}) d\mu\right). \tag{12}$$

For data input to PHREEQC, the chemical equation for the mole-balance and mass-action expressions, the log K and its temperature dependence, and, optionally, the activity-coefficient expression for each exchange species are defined through the **EXCHANGE_SPECIES** data block. Exchange master species are defined with the **EXCHANGE_MASTER_SPECIES** data block. The number of exchange sites and exchanger composition are defined with the **EXCHANGE** data block (see "Description of Data Input").

Surface Species

Surface-complexation processes are included in the model through heterogeneous mass-action equations, mole-balance equations for surface sites, and charge-potential relations for each surface. PHREEQC allows multiple surfaces and surface-site types, termed a "surface assemblage", to exist in equilibrium with the aqueous phase. Two formulations of the mass-action equations for surface species are available in PHREEQC: (1) one that includes electrostatic potential terms and (2) another that excludes all electrostatic potential terms. If the Dzombak and Morel (1990) model, which includes electrostatic potential terms, is used, additional equations and mass-action terms become operational because of surface charge and surface electrostatic potential.

The two principle differences between the formulation of exchange reactions and surface reactions are that exchange reactions are formulated as half-reactions, which causes the master species not to appear in any mole-balance equations, and the exchange species are expected to be neutral. Surface reactions are not half-reactions, so the master species is a physically real species and appears in mole-balance equations, and surface species may be anionic, cationic, or neutral.

The basic theory for surface-complexation reactions including electrostatic potentials is presented in Dzombak and Morel (1990). The theory assumes that the number of active sites, T_s (eq), the specific area, A_s (m^2/g), and the mass, S_s (g), of the surface are known. The two additional master unknowns are (1) the quantity, $\ln a_{\Psi_s} = \ln\left(e^{\frac{F\Psi_s}{2RT}}\right) = \frac{F\Psi_s}{2RT}$, where F is the Faraday constant (96493.5 J V^{-1} eq^{-1}), Ψ_s is the potential at surfaces (volts), R is the gas constant (8.3147 J mol^{-1} K^{-1}), and T is temperature (Kelvin) and (2) the natural log of the activity of the master surface species. Note that the quantity $\ln a_{\Psi_s}$ is defined with a 2 in the denominator of the term on the right-hand side. This is a different master unknown than that used in Dzombak and Morel (1990), but produces the same results as their model because all equations are written to be consistent with this master unknown.

The activity of a surface species is assumed to be equal to the mole fraction of a given surface-site type that is occupied. In other words, a surface species is in the standard state (has activity of 1) when it completely covers a given kind of surface site. This convention differs from Dzombak and Morel (1990) who assumed that activity of a surface species (conceptually in the solid phase) is numerically equal to molarity (concentration in solution). If only monodentate complexes are considered (as is done by Dzombak and Morel, 1990), terms cancel in the mass-action equation and identical numerical results are obtained irrespective of the convention for standard state. However, a notable difference in surface site concentration exists when the molarity convention is used for multidentate complexes (bidentate, tridentate, and others, cf. Appelo and Postma, 1999). If a vessel contains a solution in equilibrium with a surface containing multidentate species, and more of exactly the same solution is added, the composition of solution and surface would change with the molarity convention. The molarity convention is clearly not correct in this case.

"*Hfo*" (*H*ydrous *f*erric *o*xide) is used in the default database files with "_w", which indicates a low affinity or weak site and "_s", which indicates a high affinity or strong site. "*Hfo_wOH*" is used to represent a neutral surface species at a weak site and the association reaction for the formation of a negatively charged weak site (it is an association reaction in the sense that the defined species is on the right hand side of the equation) can be written as

$$Hfo\_wOH \rightarrow Hfo\_wO^- + H^+ . \qquad (13)$$

The mass-action expression, which includes the electrostatic potential term, is

$$K^{int}_{Hfo\_wO^-} = \frac{a_{Hfo\_wO^-} a_{H^+}}{a_{Hfo\_wOH}} e^{-\frac{F\Psi_s}{RT}}, \qquad (14)$$

where $K^{int}_{Hfo\_wO^-}$ is the intrinsic equilibrium constant for the reaction, and $e^{-\frac{F\Psi_s}{RT}}$ is a factor that accounts for the work involved in moving a charged species (H^+) away from a charged surface. In general, the mass-action equation for surface species i_{s_k} is

$$K_{i_{(s_k)}}^{int} = \left(a_{i_{(s_k)}} \prod_m^M a_m^{-c_{m,i_{(s_k)}}} \right) e^{\frac{F\Psi_s}{RT}\Delta z_{i_{(s_k)}}} \tag{15}$$

where $K_{i_{(s_k)}}^{int}$, is the intrinsic equilibrium constant; $i_{(s_k)}$ is the i^{th} surface species for surface-site type k (weak or strong in Dzombak and Morell, 1990) in surface s; m varies over all master species, M, including surface master species; c_{m,i_s} is the stoichiometric coefficient of master species, m, in the association reaction for surface species $i_{(s_k)}$, and $\Delta z_{i_{(s_k)}}$ is the net change in surface charge due to the formation of the surface species. The values of $c_{m,i_{(s_k)}}$ may be positive or negative. For PHREEQC, terms on the right-hand side of an association reaction are assigned negative coefficients and terms on the left-hand side are assigned positive coefficients.

For a surface species, the equation for the total moles of species $i_{(s_k)}$ is

$$\begin{aligned} n_{i_{(s_k)}} &= a_{i_{(s_k)}} \frac{T_{s_k}}{b_{i_{(s_k)}}} = K_{i_{(s_k)}} T_{s_k} e^{\left(-\frac{F\Psi_s}{RT}\Delta z_{i_{(s_k)}}\right)} \prod_m^M a_m^{c_{m,i_{(s_k)}}} \\ &= K_{i_{(s_k)}} \frac{T_{s_k}}{b_{i_{(s_k)}}} a_{\Psi_s}^{-2\Delta z_{i_{(s_k)}}} \prod_m^M a_m^{c_{m,i_{(s_k)}}} \end{aligned}, \tag{16}$$

where T_{s_k} is the total number of a type of surface site, and $b_{i_{(s_k)}}$ is the number of surface sites bounded to the species. The total derivative of the moles of species $i_{(s_k)}$ with respect to the master unknowns is

$$dn_{i_{(s_k)}} = n_{i_{(s_k)}} \left[\sum_m^M c_{m,i_{(s_k)}} d\ln a_m - 2\Delta z_{i_{(s_k)}} d\ln a_{\Psi_s} \right]. \tag{17}$$

The second formulation of mass-action equations for surface species excludes the electrostatic potential term in the mass-action expression (**-no_edl** identifier in the **SURFACE** data block). The equation for the moles of a surface species is the same as equation 16, except the factor involving a_{Ψ_s} does not appear. Likewise, the total derivative of the moles is the same as equation 17, except the final term is absent.

For data input to PHREEQC, the chemical equation for the mole-balance and mass-action expressions and the log K and its temperature dependence of surface species are defined through the **SURFACE_SPECIES** data block. Surface master species or types of surface sites are defined with the **SURFACE_MASTER_SPECIES** data block. The identity of the surfaces and the number of equivalents of each site type, the composition of the surface, the specific surface area, and the mass of the surface are defined with the **SURFACE** data block (see "Description of Data Input").

Gas-Phase Components

Equilibrium between a multicomponent gas phase and the aqueous phase is modeled with heterogeneous mass-action equations and an equation for total pressure (fixed-pressure gas phase only). Only one gas phase can exist in equilibrium with the aqueous phase, but the gas phase may contain multiple components. All gas components are assumed to behave ideally and the gas phase is assumed to be an ideal mixture of gas components.

If a gas phase is specified to have a fixed volume, then the pressure in the gas volume will vary with reaction extent, but each gas component will always be present in the gas phase. For a fixed-volume gas phase, no additional

master unknowns are needed, and the moles of a component in the gas phase can be calculated from the activities of the aqueous master species.

If a gas phase is specified to have a fixed pressure, the gas phase is a fixed-pressure bubble that will vary in volume with reaction extent. If the sum of the partial pressures of the component gases is less than the specified total pressure, the fixed-pressure gas phase will not exist and none of the gas components will be present in the gas phase. For a fixed-pressure gas phase, one additional master unknown is included in the equations, which is the total moles of gas components in the gas phase, N_{gas}.

By the assumption of ideality, the fugacity (activity) of a gas component is equal to its partial pressure. PHREEQC uses dissolution equations, in the sense that the gas component is assumed to be on the left-hand side of the chemical reaction. For carbon dioxide, the dissolution reaction may be written as

$$CO_{2(g)} = CO_{2(aq)}. \tag{18}$$

The Henry's law constant relates the partial pressure of the gas component (numerically equal to fugacity for ideal gases) to the activity of aqueous species. For carbon dioxide, the Henry's law constant is $10^{-1.468}$ [following the ideal gas assumption, units are atmospheres (atm)], and the following mass-action equation applies at equilibrium:

$$P_{CO_2} = 10^{1.468} a_{CO_{2(aq)}}, \tag{19}$$

where P_{CO_2} is the partial pressure (atm) calculated using activities in the aqueous phase. In general, the partial pressure of a gas component may be written in terms of aqueous phase activities as

$$P_g = \frac{1}{K_g} \prod_m^{M_{aq}} a_m^{c_{m,g}}, \tag{20}$$

where P_g is the partial pressure of gas component g, calculated using activities in the aqueous phase; K_g is the Henry's law constant for the gas component; and $c_{m,g}$ is the stoichiometric coefficient of aqueous master species, m, in the dissolution equation. The values of $c_{m,g}$ may be positive or negative. For PHREEQC, terms on the left-hand side of a dissolution reaction are assigned negative coefficients and terms on the right-hand side are assigned positive coefficients.

For a fixed-volume gas phase, the total volume of the gas phase is specified to be V_{total}, but the pressure of the gas phase is variable. At equilibrium, the number of moles of a gas component in the gas n_g is calculated as

$$n_g = \frac{V_{total} P_g}{RT} = \frac{V_{total}}{RTK_g} \sum_m^{M_{aq}} a_m^{c_{m,g}}. \tag{21}$$

The total derivative of the moles of a gas component in the gas phase is

$$dn_g = \frac{V_{total}}{RT} \sum_m^{M_{aq}} n_g c_{m,g} d\ln a_m. \tag{22}$$

For a fixed-pressure gas phase, the total pressure is specified as P_{total}, but the volume of the gas phase is variable. At equilibrium, the number of moles of a gas component in the gas phase is equal to the fraction of the total pressure for the gas times the total moles of gas in the gas phase:

$$n_g = N_{gas}\frac{P_g}{P_{total}} = \frac{N_{gas}}{P_{total}K_g}\prod_m^{M_{aq}} c_{m,g}. \quad (23)$$

The total derivative of the moles of a gas component in the gas phase is

$$dn_g = \frac{P_g}{P_{total}}dN_{gas} + \sum_m^{M_{aq}} n_g c_{m,g} d\ln a_m. \quad (24)$$

For data input to PHREEQC, the mass-action equations, Henry's law constant, and temperature dependence of the constant are defined with the **PHASES** data block. The type of gas phase (fixed-volume or fixed-pressure), the components to include in gas-phase calculations, and initial gas-phase composition are defined with the **GAS_PHASE** data block (see "Description of Data Input").

Equations for the Newton-Raphson Method

A series of functions, denoted by f, are used to describe heterogeneous equilibrium. These equations are derived primarily by substituting the equations for the moles of species (derived from mass-action equations in the previous section) into mole- and charge-balance equations. When equilibrium is satisfied, all of the functions relevant to a specific equilibrium calculation are equal to zero. The zeros of the functions are found by the Newton-Raphson method, by which each function is differentiated with respect to each master unknown to form the Jacobian matrix. A set of linear equations is formed from the Jacobian matrix that can be solved to approximate a solution to the nonlinear equations. By iteratively solving successive sets of linear equations, a solution to the nonlinear equations can be found. Each of the f functions that is used in the numerical method is presented in this section along with the total derivative with respect to the master unknowns that is used to form the Jacobian matrix.

Activity of Water

The activity of water is calculated from an approximation that is based on Raoult's law (Garrels and Christ, 1965, p. 65-66):

$$a_{H_2O} = 1 - 0.017\sum_i^{N_{aq}} \frac{n_i}{W_{aq}}. \quad (25)$$

The function f_{H_2O} is defined as

$$f_{H_2O} = W_{aq}(a_{H_2O} - 1) + 0.017\sum_i^{N_{aq}} n_i, \quad (26)$$

and the total derivative of this function is

$$df_{H_2O} = W_{aq}a_{H_2O}d\ln(a_{H_2O}) + (a_{H_2O} - 1)W_{aq}d\ln(W_{aq}) + 0.017\sum_i^{N_{aq}} dn_i. \quad (27)$$

The master unknown is the natural log of the activity of water $\ln a_{H_2O}$.

Ionic Strength

The ionic strength of the aqueous solution is a master unknown and is defined as

$$\mu = \frac{1}{2}\sum_{i}^{N_{aq}} z_i^2 \frac{n_i}{W_{aq}}. \tag{28}$$

The function f_μ is defined as

$$f_\mu = W_{aq}\mu - \frac{1}{2}\sum_{i}^{N_{aq}} z_i^2 n_i, \tag{29}$$

and the total derivative of this function is

$$df_\mu = \mu W_{aq} d\ln(W_{aq}) + W_{aq} d\mu - \frac{1}{2}\sum_{i}^{N_{aq}} z_i^2 dn_i. \tag{30}$$

Equilibrium with a Fixed-Volume Multicomponent Gas Phase

For a fixed-volume gas phase, the moles of each gas component can be calculated from the activities of the aqueous master species, and the numerical model treats the gas phase components in the same way that it treats aqueous species. The terms for the moles of each gas components, n_g, appear in the mole-balance equations for elements and the terms dn_g appear in the Jacobian matrix for the mole-balance equations. No additional equation labeled f is required to calculate equilibrium with the fixed-volume gas phase.

For data input to PHREEQC, the mass-action equations, Henry's law constant, and temperature dependence of the constant are defined with the **PHASES** data block. The type of gas phase (fixed-volume or fixed-pressure), the components to include in gas-phase calculations, and initial gas-phase composition are defined with the **GAS_PHASE** data block (see "Description of Data Input").

Equilibrium with a Fixed-Pressure Multicomponent Gas Phase

For a fixed-volume gas phase, the number of moles of each gas component is calculated from the activities of the aqueous master species and the total moles of gas components in the gas phase, N_g. The terms for the moles of each gas components, n_g, appear in the mole-balance equations for elements and the terms dn_g appear in the Jacobian matrix for the mole-balance equations. Equilibrium between a fixed-pressure multicomponent gas phase and the aqueous phase requires one new equation--the sum of the partial pressures of the component gases is equal to the total pressure, P_{total}. The function $f_{P_{total}}$ is defined as

$$f_{P_{total}} = P_{total} - \sum_{g}^{N_g} P_g, \tag{31}$$

where N_g is the total number of gas components in the gas phase.

The total derivative of $f_{P_{total}}$ with respect to the master unknowns, with the convention that positive dN_{gas} are increases in solution concentration, is

$$df_{P_{total}} = -\sum_{g}^{N_g} \sum_{m}^{M_{aq}} c_{m,g} P_g d\ln a_m . \quad (32)$$

For data input to PHREEQC, the mass-action equations, Henry's law constant, and temperature dependence of the constant are defined with the **PHASES** data block. The type of gas phase (fixed-volume or fixed-pressure), the components to include in gas-phase calculations, and initial gas-phase composition are defined with the **GAS_PHASE** data block (see "Description of Data Input").

Equilibrium with Pure Phases

Equilibrium between the aqueous phase and pure phases, including gases with fixed partial pressures, is included in the model through heterogeneous mass-action equations. PHREEQC allows multiple pure phases, termed a pure-phase assemblage, to exist in equilibrium with the aqueous phase, subject to the limitations of the Gibbs' Phase Rule. The activity of a pure phase is assumed to be identically 1.0. The additional master unknown for each pure phase is the moles of the pure phase that is present in the system, n_p, where p refers to the p^{th} phase. Terms representing the changes in the moles of each pure phase occur in the mole-balance equations for elements. PHREEQC also allows a calculation where equilibrium with a pure phase is produced by adding or removing a specified reactant (*alternative formula* and *alternative phase* in **EQUILIBRIUM_PHASES** data block); the mole transfer of the reactant that is necessary to produce equilibrium with the pure phase is calculated. In this type of calculation, the terms in the mole-balance equations are derived from the stoichiometry of the reactant rather than the stoichiometry of the pure phase, and the unknown is the number of moles of reactant that enter or leave solution.

The new function corresponding to each of the new unknowns is a mass-action expression for each pure phase. PHREEQC uses dissolution reactions, in the sense that the pure phase is on the left-hand side of the chemical equation. For calcite, the dissolution reaction may be written as

$$CaCO_3 = Ca^{2+} + CO_3^{2-}, \quad (33)$$

and, using log K of $10^{-8.48}$ and activity of the pure solid of 1.0, the resulting mass-action expression is

$$K_{calcite} = 10^{-8.48} = a_{Ca^{2+}} a_{CO_3^{2-}} . \quad (34)$$

In general, pure-phase equilibria can be represented with the following equation:

$$K_p = \prod_{m}^{M_{aq}} a_m^{c_{m,p}}, \quad (35)$$

where $c_{m,p}$ is the stoichiometric coefficient of master species m in the dissolution reaction. The values of $c_{m,p}$ may be positive or negative. For PHREEQC, terms on the left-hand side of a dissolution reaction are assigned negative coefficients and terms on the right-hand side are assigned positive coefficients. The saturation index for the mineral, SI_p, is defined to be

$$SI_p = \log \prod_{m}^{M_{aq}} a_m^{c_{m,p}} . \quad (36)$$

The function used for phase equilibrium in the numerical method is

$$f_p = (\ln K_p + [\ln(10)]SI_{p,target}) - \sum_m^{M_{aq}} c_{m,p} \ln(a_m), \quad (37)$$

where $SI_{p,target}$ is the target saturation index for the phase, and $\ln(10)$ converts base-10 log to natural log. The target saturation index is specified by the user; a positive, zero, or negative value specifies supersaturation, equilibrium, or undersaturation for the mineral with respect to the solution. For fixed-partial-pressure gas component, $SI_{p,target}$ is equivalent to the log of the partial pressure of the gas component. The total derivative with respect to the master unknowns is

$$df_p = -\sum_m^{M_{aq}} c_{m,p} d\ln a_m. \quad (38)$$

For data input to PHREEQC, the mass-action equations, equilibrium constant, and temperature dependence of the constant for a pure phase are defined with the **PHASES** data block. Initial composition of a pure-phase assemblage and target saturation indices are defined with the **EQUILIBRIUM_PHASES** data block.

Equilibrium with Solid Solutions

Modeling of ideal, multicomponent or nonideal, binary solid solutions is based on the work of Glynn (Glynn and Reardon, 1990; Glynn and others, 1990; Glynn, 1991; Glynn and Parkhurst, 1992). Equilibrium between the aqueous phase and solid solutions is included in the model through heterogeneous mass-action equations. PHREEQC allows multiple solid solutions, termed a solid-solution assemblage, to exist in equilibrium with the aqueous phase, subject to the limitations of the Gibbs' Phase Rule. Modeling of nonideal solid solutions is limited to two-component (binary) solid solutions; ideal solid solutions may have two or more components. The additional master unknowns for solid solutions are the moles of each component in each solid solution $n_{p_{ss}}$, where *ss* refers to solid solution *ss*. Terms representing the changes in the moles of each component occur in the Jacobian matrix of the mole-balance equations for elements.

Unlike pure phases, the activity of a component in a solid solution is not identically 1.0. The activity of a component is defined to be $a_{p_{ss}} = \lambda_{p_{ss}} x_{p_{ss}}$, where $x_{p_{ss}}$ is the mole fraction of component *p* in the solid solution *ss*, and $\lambda_{p_{ss}}$ is the activity coefficient. The mole fraction of a component in a solid solution is defined as

$x_{p_{ss}} = \dfrac{n_{p_{ss}}}{\sum_{p_{ss}=1}^{N_{ss}} n_{p_{ss}}}$, where N_{ss} is the number of components in solid solution *ss*. For ideal solid solutions, the activity coefficient is 1.0; for nonideal, binary solid solutions, the activity coefficients for the components are defined with the Guggenheim expressions:

$$\lambda_1 = \exp((a_0 - a_1(4x_1 - 1))x_2^2) \text{ and} \quad (39)$$

$$\lambda_2 = \exp((a_0 + a_1(4x_2 - 1))x_1^2), \quad (40)$$

where λ_1 and λ_2 are the activity coefficients of components 1 and 2, and a_0 and a_1 are nondimensional Guggenheim parameters. The nondimensional parameters are calculated from dimensional parameters for the

excess free energy g_0 and g_1 (kJ/mol) by the equations: $a_0 = \frac{g_0}{RT}$ and $a_1 = \frac{g_1}{RT}$. The parameters a_0 and a_1 for the excess free energy may be defined directly or by a variety of means including the mole fractions of component 2 delimiting the miscibility gap, the mole fractions of component 2 delimiting the spinodal gap, the mole fraction of component 2 at the critical point and the critical temperature, Thompson and Waldbaum parameters, Margules parameters, mole fraction of component 2 and the log of the total solubility product of an alyotropic point, solid-phase activity coefficients for trace concentrations of component 1 and component 2, or two distribution coefficients for component 2 (Glynn, 1991).

The new function corresponding to each of the new unknowns is a mass-action expression for each component in each solid solution. PHREEQC uses dissolution reactions, in the sense that the solid-solution component is on the left-hand side of the chemical equation. For aragonite in an aragonite-strontianite solid solution, the dissolution reaction may be written as

$$CaCO_3 = Ca^{2+} + CO_3^{2-}, \tag{41}$$

and, using log K of $10^{-8.34}$ and activity coefficient for the solid, the resulting mass-action expression is

$$K_{Arag} = 10^{-8.34} = \frac{a_{Ca^{2+}} a_{CO_3^{2-}}}{a_{Arag}} = \frac{a_{Ca^{2+}} a_{CO_3^{2-}}}{\lambda_{Arag}\left(\frac{n_{Arag}}{n_{Arag} + n_{Stront}}\right)}. \tag{42}$$

In general, solid-solution phase equilibria can be represented with the following equation for each component:

$$K_{p_{ss}} = \frac{\prod_{m}^{M_{aq}} a_m^{c_{m,p_{ss}}}}{a_{p_{ss}}}, \tag{43}$$

where $K_{p_{ss}}$ is the equilibrium constant of component p in pure form, and $c_{m,p_{ss}}$ is the stoichiometric coefficient of master species m in the dissolution reaction for component p in solid solution ss. The values of $c_{m,p_{ss}}$ may be positive or negative. For PHREEQC, terms on the left-hand side of a phase dissolution reaction are assigned negative coefficients and terms on the right-hand side are assigned positive coefficients. The solubility quotient for a component of the solid solution is defined to be

$$Q_{p_{ss}} = \frac{\sum_{m}^{M_{aq}} a_m^{c_{m,p_{ss}}}}{K_{p_{ss}} a_{p_{ss}}}, \tag{44}$$

where $Q_{p_{ss}}$ is equal to 1 and $\ln Q_{p_{ss}}$ is equal to 0 at equilibrium. The functions used in the numerical method for each component of a nonideal, binary solid solution are

$$f_1 = \sum_{m}^{M_{aq}} c_{m,1} \ln a_m - \ln K_1 - \ln \frac{n_1}{n_1 + n_2} - \ln \lambda_1 \text{ and} \tag{45}$$

$$f_2 = \sum_{m}^{M_{aq}} c_{m,2} \ln a_m - \ln K_2 - \ln \frac{n_2}{n_1 + n_2} - \ln \lambda_2. \tag{46}$$

The total derivative with respect to the master unknowns is

$$df_1 = \sum_m^{M_{aq}} c_{m,1} d\ln a_m + \left(-\frac{x_2}{n_1} + \frac{2a_0x_2^2 - 6a_1x_2^2 + 12a_1x_2^3}{n_1 + n_2} \right) dn_1 + \frac{-2a_0x_2 + 2a_0x_2^2 + 6a_1x_2 - 18a_1x_2^2 + 12a_1x_2^3 + 1}{n_1 + n_2} dn_2 \quad (47)$$

and

$$df_2 = \sum_m^{M_{aq}} c_{m,2} d\ln a_m + \frac{-2a_0x_1 + 2a_0x_1^2 - 6a_1x_1 + 18a_1x_1^2 - 12a_1x_1^3 + 1}{n_1 + n_2} dn_1 + \left(-\frac{x_1}{n_2} + \frac{2a_0x_1^2 + 6a_1x_1^2 - 12a_1x_1^3}{n_1 + n_2} \right) dn_2 \quad . \quad (48)$$

The function used in the numerical method for each component of an ideal solid solution is

$$f_{p_{ss}} = \ln Q_{p_{ss}} = \ln \left(\frac{\sum_m^{M_{aq}} a_m^{c_{m,p_{ss}}}}{K_{p_{ss}}} \right) - \ln \left(\frac{n_{p_{ss}}}{N_{total}} \right), \quad (49)$$

where $N_{total} = \sum_{j_{ss}}^{N_{ss}} n_{j_{ss}}$ and j_{ss} ranges over all the components in solid solution ss. The total derivative with respect to the master unknowns is

$$df_{p_{ss}} = \sum_m^{M_{aq}} c_{m,p_{ss}} d\ln a_m - \frac{1}{n_{p_{ss}}} \left(\frac{N_{total} - n_{p_{ss}}}{N_{total}} \right) dn_{p_{ss}} + \sum_{j_{ss}}^{N_{ss}, j_{ss} \neq p_{ss}} \frac{1}{N_{total}} dn_{j_{ss}} . \quad (50)$$

For data input to PHREEQC, the mass-action equations, equilibrium constant, and temperature dependence of the constant for each pure phase are defined with the **PHASES** data block. Initial composition of a solid-solution assemblage and Guggenheim parameters for nonideal solid solutions are defined with the **SOLID_SOLUTIONS** data block (see "Description of Data Input").

Mole Balance for Surface Sites

Mole balance for a surface site is a special case of the general mole-balance equation. The surface assemblage is a set of one or more surfaces, each of which may have one or more site types. The total number of moles of a surface site type is specified by input to be one of the following: (1) fixed, (2) proportional to the moles of a pure phase, or (3) proportional to the moles of a kinetic reactant. The sum of the moles of surface sites occupied by the surface species of a site type must equal the total moles of that surface site type. The following function is derived from the mole-balance relation for a surface site type s_k of surface s:

$$f_{s_k} = T_{s_k} - \sum_{i_{(s_k)}}^{N_{s_k}} b_{s_k, i_{(s_k)}} n_{i_{(s_k)}}, \tag{51}$$

where the value of the function f_{s_k} is zero when mole balance is achieved, T_{s_k} is the moles of the surface site type, N_{s_k} is the number of surface species for the site type, and $b_{s_k, i_{(s_k)}}$ is the number of surface sites occupied by the surface species $i_{(s_k)}$. The total derivative of f_{s_k} is

$$df_{s_k} = \Delta T_{s_k} - \sum_{i_{(s_k)}}^{N_{s_k}} b_{s_k, i_{(s_k)}} dn_{i_{(s_k)}}. \tag{52}$$

If the total number of sites is proportional to the moles of a pure phase, then $\Delta T_{s_k} = -c_{s_k, p} dn_p$, where $c_{s_k, p}$ is the moles of surface sites per mole of phase p. If the phase dissolves, then dn_p is positive and the number of surface sites decreases. If the total number of sites is proportional to the moles of a kinetic reactant, $\Delta T_{s_k} = 0$ in the total derivative equation. The change in the number of sites is included as part of the reaction that is integrated with the rate equations and no term is included in the Jacobian matrix. As the kinetic reaction increases or decreases the moles of reactant, the number of surface sites is adjusted proportionately. If the number of surface sites is fixed, $\Delta T_{s_k} = 0$.

For data input to PHREEQC, the number of moles of each type of surface site is defined with the **SURFACE** data block and may be a fixed quantity or it may be related to the moles of a pure phase or a kinetic reactant. Surface site types are defined with the **SURFACE_MASTER_SPECIES** data block and surface species are defined with the **SURFACE_SPECIES** data block (see "Description of Data Input").

Mole Balance for Exchange Sites

Mole balance for an exchange site is a special case of the general mole-balance equation. The total number of moles of an exchange site is specified by input to be one of the following: (1) fixed, (2) proportional to the moles of a pure phase, or (3) proportional to the moles of a kinetic reactant. The sum of the moles of sites occupied by exchange species must equal the total moles of the exchange site. The following function is derived from the mole-balance relation for an exchange site:

$$f_e = T_e - \sum_{i_e}^{N_e} b_{e, i_e} n_{i_e}, \tag{53}$$

where the value of the function f_e is zero when mole balance is achieved, T_e is the total moles of exchange sites for exchanger e, and b_{e, i_e} is the number of exchange sites occupied by the exchange species. The total derivative of f_e is

$$df_e = \Delta T_e - \sum_{i_e}^{N_e} b_{e, i_e} dn_{i_e}. \tag{54}$$

If the total number of sites is proportional to the moles of a pure phase, then $\Delta T_e = -c_{e, p} dn_p$, where $c_{e, p}$ is the moles of exchange sites per mole of phase p. If the phase dissolves, then dn_p is positive and the number of exchange sites decreases. If the total number of sites is proportional to the moles of a kinetic reactant, $\Delta T_e = 0$ in

the total derivative equation. The change in the number of sites is included as part of the reaction that is integrated with the rate equations and no term is included in the Jacobian matrix. As the kinetic reaction increases or decreases the moles of the reactant, the number of exchange sites is adjusted proportionately. If the number of exchange sites is fixed, $\Delta T_e = 0$.

For data input to PHREEQC, the moles of exchange sites are defined in the **EXCHANGE** data block and may be a fixed quantity or it may be related to the moles of a pure phase or a kinetic reactant. Exchanger sites are defined with the **EXCHANGE_MASTER_SPECIES** data block and exchange species are defined with the **EXCHANGE_SPECIES** data block (see "Description of Data Input").

Mole Balance for Alkalinity

The mole-balance equation for alkalinity is used only in speciation calculations and in inverse modeling. Mole balance for alkalinity is a special case of the general mole-balance equation where the coefficients are defined by the alkalinity contribution of each aqueous species. Alkalinity is defined as an element in PHREEQC and a master species is associated with this element (see **SOLUTION_MASTER_SPECIES** keyword in "Description of Data Input"). In the default databases for PHREEQC, the master species for alkalinity is CO_3^{2-}. The master unknown for alkalinity is $\ln a_{Alk}$, or for the default databases, $\ln a_{CO_3^{2-}}$.

The total number of equivalents of alkalinity is specified by input to the model. The sum of the alkalinity contribution of each aqueous species must equal the total number of equivalents of alkalinity. The following function is derived from the alkalinity-balance equation:

$$f_{Alk} = T_{Alk} - \sum_{i}^{N_{aq}} b_{Alk,i} n_i, \tag{55}$$

where the value of the function f_{Alk} is zero when mole balance is achieved, T_{Alk} is the number of equivalents of alkalinity in solution, and $b_{Alk,i}$ is the alkalinity contribution of the aqueous species i (eq/mol). The total derivative of f_{Alk} is

$$df_{Alk} = -\sum_{i}^{N_{aq}} b_{Alk,i} dn_i. \tag{56}$$

The value of T_{Alk} must be positive, provided a carbonate species is the master species for alkalinity. Conceptually, a measured alkalinity differs from the alkalinity calculated by PHREEQC. In the default database files for PHREEQC the values of $b_{Alk,i}$ have been chosen such that the reference state ($b_{Alk,i} = 0$) for each element or element valence state is the predominant species at a pH of 4.5. It is assumed that all of the element or element valence state is converted to this predominant species in an alkalinity titration. However, significant concentrations of aqueous species that are not in the reference state (that is species that have nonzero alkalinity contributions) may exist at the endpoint of a titration, and the extent to which this occurs causes the alkalinity calculated by PHREEQC to be a different quantity than the measured alkalinity. Hydroxide complexes of iron and aluminum are the most common examples of species that may not be converted to the defined reference state. Thus, the alkalinity of a solution as calculated by PHREEQC, though it will be numerically equal to the measured alkalinity, is an approximation because of the assumption that a titration totally converts elements and element valence states to

their reference state. In most solutions, where the alkalinity is derived predominantly from carbonate species, the approximation is valid.

For data input to PHREEQC, the alkalinity of each species is calculated from the association reaction for the species, which is defined in the **SOLUTION_SPECIES** data block, and the alkalinity contributions of the master species, which are defined with the **SOLUTION_MASTER_SPECIES** data block. Total alkalinity is part of the solution composition defined with the **SOLUTION** or **SOLUTION_SPREAD** data block (see "Description of Data Input").

Mole Balance for Elements

The total moles of an element in the system are the sum of the moles initially present in the pure-phase and solid-solution assemblages, aqueous phase, exchange assemblage, surface assemblage, gas phase, and diffuse layers of the surfaces. The following function is the general mole-balance equation:

$$f_m = \left(T_m - \sum_p^{N_p} b_{m,p} n_p - \sum_{ss}^{SS} \sum_{p_{ss}}^{N_{ss}} b_{m,p_{ss}} n_{p_{ss}} \right) - \sum_i^{N_{aq}} b_{m,i} n_i - \sum_e^{E} \sum_{i_e}^{N_e} b_{m,i_e} n_{i_e} - \sum_s^{S} \sum_k^{K_s} \sum_{i_{(s_k)}}^{N_{s_k}} b_{m,i_{(s_k)}} n_{i_{(s_k)}} - \sum_g^{N_g} b_{m,g} n_g - \sum_s^{S} \sum_i^{N_{aq}} b_{m,i} n_{i,s} \quad (57)$$

where the value of the function f_m is zero when mole-balance is achieved, T_m is the total moles of the element in the system, N_p is the number of phases in the pure-phase assemblage, SS is the number of solid solutions in the solid-solution assemblage, N_{ss} is the number of components in solid solution ss, N_{aq} is the number of aqueous species, E is the number of exchangers in the exchange assemblage, N_e is the number of exchange species for exchange site e, S is the number of surfaces in the surface assemblage, K_s is the number of surface types for surface s, N_{s_k} is the number of surface species for surface type s_k, and N_g is the number of gas-phase components. The moles of each entity in the system are represented by n_p for phases in the pure-phase assemblage, $n_{p_{ss}}$ for components in a solid solution, n_i for aqueous species, n_{i_e} for the exchange species of exchange site e, $n_{i_{(s_k)}}$ for surface species for surface site type s_k, n_g for the gas components, and $n_{i,s}$ for aqueous species in the diffuse layer of surface s. The moles of element m per mole of each entity are represented by b_m, with an additional subscript to define the relevant entity; b_m is usually, but not always, equal to c_m (the coefficient of the master species for m in the mass-action equation).

To avoid solving for small differences between large numbers, the quantity in parenthesis in equation 57 is not explicitly included in the solution algorithm and the value of T_m is never actually calculated. Instead the quantity

$${}^*T_m = T_m - \sum_p^{N_p} b_{m,p} n_p - \sum_{ss}^{SS} \sum_{p_{ss}}^{N_{ss}} b_{m,p_{ss}} n_{p_{ss}}$$ is used in the function f_m. Initially, ${}^*T_m$ is calculated from the total moles of m in the aqueous phase, the exchange assemblage, the surface assemblage, the gas phase, and the surface diffuse layers:

$$ {}^{*}T_m = \sum_{i}^{N_{aq}} b_{m,i} n_i + \sum_{e}^{E} \sum_{i_e}^{N_e} b_{m,i_e} n_{i_e} + \sum_{s}^{S} \sum_{k}^{K_s} \sum_{i_{(s_k)}}^{N_{s_k}} b_{m,i_{(s_k)}} n_{i_{(s_k)}} + \sum_{g}^{N_g} b_{m,g} n_g + \sum_{s}^{S} \sum_{i}^{N_{aq}} b_{m,i} n_{i,s} . \quad (58) $$

During the iterative solution to the equations, ${}^{*}T_m$ is updated by the mole transfers of the pure phases and components of the solid solutions:

$$ {}^{*}T_{k+1}^{m} = {}^{*}T_{k}^{m} + \sum_{p}^{N_p} b_{m,p} dn_p + \sum_{ss}^{SS} \sum_{p_{ss}}^{N_{ss}} b_{m,p_{ss}} dn_{p_{ss}}, \quad (59) $$

where k refers to the iteration number. It is possible for ${}^{*}T_m$ to be negative in intermediate iterations, but it must be positive when equilibrium is attained.

The total derivative of the function f_m is

$$ \begin{aligned} df_m = & -\sum_{p}^{N_p} b_{m,p} dn_p - \sum_{ss}^{SS} \sum_{p_{ss}}^{N_{ss}} b_{m,p_{ss}} dn_{p_{ss}} - \sum_{i}^{N_{aq}} b_{m,i} dn_i - \sum_{e}^{E} \sum_{i_e}^{N_e} b_{m,i_e} dn_{i_e} - \\ & - \sum_{s}^{S} \sum_{k}^{K_s} \sum_{i_{(s_k)}}^{N_{s_k}} b_{m,i_{(s_k)}} dn_{i_{(s_k)}} - \sum_{g}^{N_g} b_{m,g} dn_g - \sum_{s}^{S} \sum_{i}^{N_{aq}} b_{m,i} dn_{i,s} \end{aligned} \quad (60) $$

For data input to PHREEQC, total moles of elements are initially defined for an aqueous phase with the **SOLUTION** or **SOLUTION_SPREAD** data block, for an exchange assemblage with the **EXCHANGE** data block, for a surface assemblage with the **SURFACE** data block, for the gas phase with a **GAS_PHASE** data block. The moles of each phase in a pure-phase assemblage are defined with the **EQUILIBRIUM_PHASES** data block. The moles of each component in each solid solution in a solid-solution assemblage are defined with the **SOLID_SOLUTIONS** data block. Total moles of elements may also be modified by batch-reaction and transport calculations (see "Description of Data Input").

Aqueous Charge Balance

The charge-balance equation sums the equivalents of aqueous cations and anions and, in some cases, the charge imbalances developed on surfaces and exchangers. When specified, a charge-balance equation is used in initial solution calculations to adjust the pH or the activity of a master species (and consequently the total concentration of an element or element valence state) to produce electroneutrality in the solution. The charge-balance equation is necessary to calculate pH in batch reactions and transport simulations.

In real solutions, the sum of the equivalents of anions and cations must be zero. However, analytical errors and unanalyzed constituents in chemical analyses generally cause electrical imbalances to be calculated for solutions. If a charge imbalance is calculated for an initial solution, the pH is adjusted in subsequent batch reactions or transport simulations to maintain the same charge imbalance. If mixing is performed, the charge imbalance for the batch-reaction step is the sum of the charge imbalances of each solution weighted by its mixing factor. If a surface is used in a simulation and the explicit diffuse-layer calculation is not specified, then the formation of charged surface species will result in a surface charge imbalance. Similarly, if exchange species are not electrically

neutral (all exchange species in the default databases are electrically neutral), the exchanger will accumulate a charge. The charge imbalances of surfaces and exchangers are included in the general charge-balance equation.

The charge imbalance for a solution is calculated in an initial solution calculation, in each batch-reaction step, and for each cell during each time step of transport simulations with the equation:

$$T_{z,q} = \sum_{i}^{N_{aq}} z_i n_i, \tag{61}$$

where q identifies the aqueous phase, $T_{z,q}$ is the charge imbalance for aqueous phase q, and z_i is the charge on aqueous species i. If charged surfaces or exchangers are not present, the charge imbalance for a solution at the end of a batch-reaction or transport simulation will be the same as at the beginning of the simulation.

The charge imbalance on a surface is calculated in the initial surface-composition calculation, in each batch-reaction step, and for each cell during each time step of transport simulations with the equation:

$$T_{z,s} = \sum_{k}^{K_s} \sum_{i_{(s_k)}}^{N_{s_k}} z_{i_{(s_k)}} n_{i_{(s_k)}} + \sum_{i}^{N_{aq}} z_i n_{i,s}, \tag{62}$$

where $T_{z,s}$ is the charge imbalance for the surface, $z_{i_{(s_k)}}$ is the charge on the surface species i of surface type s_k of surface s, and the final term in the equation represents the charge accumulated in the diffuse layer. The final term is used only if the diffuse-layer composition is explicitly included in the calculation (**-diffuse_layer** in the **SURFACE** data block). When the diffuse-layer composition is calculated explicitly, it is required that all solutions be charge balanced, and $T_{z,s}$ will always be equal to zero.

Normally, exchange species have no net charge, but for generality, this is not required. However, the activity of exchange species (the equivalent fraction) is not well defined if the sum of the charged species is not equal to the total number of equivalents of exchange sites (exchange capacity). If charged exchange species exist, then the charge imbalance on an exchanger is calculated in the initial exchange-composition calculation, in each batch-reaction step, and for each cell during each time step of transport simulations with the equation:

$$T_{z,e} = \sum_{i_e}^{N_e} z_{i_e} n_{i_e}, \tag{63}$$

where $T_{z,e}$ is the charge imbalance for the exchanger, and z_{i_e} is the charge on the exchange species i of exchanger e.

The charge imbalance for the system is defined at the beginning of each batch-reaction step and for each cell at the beginning of each time step in transport simulations to be:

$$T_z = \sum_{q}^{Q} \alpha_q T_{z,q} + \sum_{s}^{S} T_{z,s} + \sum_{e}^{E} T_{z,e}, \tag{64}$$

where T_z is the charge imbalance for the system, Q is the number of aqueous phases that are mixed in the batch-reaction step or in the cell for a transport step, α_q is the mixing fraction for aqueous phase q, S is the number of surfaces, and E is the number of exchangers.

The charge-balance function is

$$f_z = T_z - \sum_i^{N_{aq}} z_i n_i - \left(\sum_s \sum_k \sum_{i_{(s_k)}}^{K_s} \sum^{N_{s_k}} z_{i_{(s_k)}} n_{i_{(s_k)}} + \sum_s^{S} \sum_i^{N_{aq}} z_i n_{i,s} \right) - \sum_e^{E} \sum_{i_e}^{N_e} z_{i_e} n_{i_e}, \quad (65)$$

where f_z is zero when charge balance has been achieved. If the diffuse-layer composition is explicitly calculated, a separate charge-balance equation is included for each surface and the sum of the terms in the parentheses will be zero when surface charge balance is achieved. If the diffuse-layer composition is not calculated, the second term inside the parentheses is zero. The total derivative of f_z is

$$df_z = -\sum_i^{N_{aq}} z_i dn_i - \sum_s^{S} \sum_k \sum_{i_{(s_k)}}^{K_s} \sum^{N_{s_k}} z_{i_{(s_k)}} dn_{i_{(s_k)}} - \sum_e^{E} \sum_{i_e}^{N_e} z_{i_e} dn_{i_e}, \quad (66)$$

where the triple summation for surfaces is present only if the diffuse-layer composition is not explicitly calculated.

For data input to PHREEQC, charge imbalance is defined by data input for **SOLUTION** or **SOLUTION_SPREAD**, **EXCHANGE**, and **SURFACE** data blocks combined with speciation, initial exchange-composition, and initial surface-composition calculations. The charge on a species is defined in the balanced chemical reaction that defines the species in **SOLUTION_SPECIES**, **EXCHANGE_SPECIES**, or **SURFACE_SPECIES** data blocks (see "Description of Data Input").

Surface Charge-Potential Equation with No Explicit Calculation of the Diffuse-Layer Composition

By default, PHREEQC uses the approach described by Dzombak and Morel (1990) to relate the charge density on the surface, σ_s, with the potential at the surface, Ψ_s. The surface-charge density is the amount of charge per area of surface material, which can be calculated from the distribution of surface species:

$$\sigma_s = \frac{F}{A_{surf}} \sum_k \sum_{i_{(s_k)}}^{K_s} \sum^{N_{s_k}} z_{i_{(s_k)}} n_{i_{(s_k)}}, \quad (67)$$

where σ_s is the charge density for surface s in coulombs per square meter (C/m^2), F is the Faraday constant in coulombs per mole (96,493.5 C/mol), A_{surf} is the surface area of the material (m^2). The surface area is calculated by one of the following formulas: (1) $A_{surf} = A_s S_s$, where A_s is the specific area of the surface material (m^2/g), and S_s is the mass of surface material (g), or (2) $A_{surf} = A_r n_r$, where A_r is the surface area per mole of a pure phase or kinetic reactant (m^2/mol), and n_r is the moles of the pure phase or reactant. At 25°C, the surface-charge density is related to the electrical potential at the surface by:

$$\sigma_s = (8000 \varepsilon \varepsilon_0 RT)^{\frac{1}{2}} \mu^{\frac{1}{2}} \sinh\left(\frac{\nu F \Psi_s}{2RT}\right), \quad (68)$$

where ε is the dielectric constant of water (78.5, dimensionless), ε_0 is the permittivity of free space (8.854x10^{-12} CV^{-1}m^{-1} or C^2/m-J), ν is the ionic charge of a symmetric electrolyte, R is the gas constant (8.314 J mol^{-1} K^{-1}), T is temperature (K), μ is the ionic strength, and F is the Faraday constant (J V^{-1} eq^{-1} or C/mol), Ψ_s is the potential at the surface in volts. At 25°C, $(8000 \varepsilon \varepsilon_0 RT)^{\frac{1}{2}} = 0.1174$. The charge of the electrolyte ions is assumed to be 1.

The charge-potential function is

$$f_{\Psi_s} = (8000\varepsilon\varepsilon_0 RT)^{\frac{1}{2}}\mu^{\frac{1}{2}}\sinh\left(\frac{F\psi_s}{2RT}\right) - \frac{F}{A_{surf}}\sum_k^{K_s}\sum_{i_{(s_k)}}^{N_{s_k}} z_{i_{(s_k)}} n_{i_{(s_k)}}, \tag{69}$$

and the total derivative of this function is

$$df_{\Psi_s} = \frac{(8000\varepsilon\varepsilon_0 RT)^{\frac{1}{2}}}{2}\mu^{-\frac{1}{2}}\sinh\left(\frac{F\psi_s}{2RT}\right)d\mu + (8000\varepsilon\varepsilon_0 RT)^{\frac{1}{2}}\mu^{\frac{1}{2}}\cosh\left(\frac{F\psi_s}{2RT}\right)d\ln a_{\Psi_s} -$$

$$\frac{F}{A_{surf}}\sum_k^{K_s}\sum_{i_{(s_k)}}^{N_{s_k}} z_{i_{(s_k)}} dn_{i_{(s_k)}} \tag{70}$$

For data input to PHREEQC, calculation without an explicit diffuse layer is the default. Specific surface area (A_s or A_r) and mass of surface (S_s) are defined in the **SURFACE** data block. The moles of surface sites are defined (1) in the **SURFACE** data block if the number of sites is fixed, (2) by a proportionality factor in the **SURFACE** data block and the moles of a phase in **EQUILIBRIUM_PHASES** data block, or (3) by a proportionality factor in the **SURFACE** data block and the moles of a kinetic reactant in **KINETICS** data block. The charge on a surface species is specified in the balanced chemical reaction that defines the species in the **SURFACE_SPECIES** data block (see "Description of Data Input").

Surface Charge-Balance Equation with Explicit Calculation of the Diffuse-Layer Composition

As an alternative to the previous model for the surface charge-potential relation, PHREEQC optionally will use the approach developed by Borkovec and Westall (1983). Their development solves the Poisson-Boltzmann equation to determine surface excesses of ions in the diffuse layer at the oxide-electrolyte interface. Throughout the derivation that follows, it is assumed that a volume of one liter (L) contains 1 kg of water.

The surface excess is:

$$\Gamma_{i,s} = \int_{x_{d,s}}^{\infty} (c_{i,s}(x) - c_i^o)dx, \tag{71}$$

where $\Gamma_{i,s}$ is the surface excess in mol m^{-2} of aqueous species i on surface s, $x_{d,s}$ is the location of the outer Helmholtz plane, $c_{i,s}(x)$ is concentration as a function of distance from the surface in mol m^{-3}, and c_i^o is the concentration in the bulk solution. The surface excess is related to concentration in the reference state of 1.0 kg of water by

$$m_{i,s} = A_{surf}\Gamma_{i,s}, \tag{72}$$

where $m_{i,s}$ is the surface excess of aqueous species i in moles per kilogram water (mol/kgw). This surface-excess concentration can be related to the concentration in the bulk solution by

$$m_{i,s} = g_{i,s} m_i, \tag{73}$$

where $g_{i,s}$ is a function of the potential at the surface and the concentrations and charges of all ions in the bulk solution:

$$g_{i,s} = A_{surf} \text{sign}(X_{d,s} - 1) \alpha \int_1^{X_{d,s}} \frac{(X^{z_i} - 1)}{\left[X^2 \sum_l^{N_{aq}} m_l (X^{z_l} - 1) \right]^{1/2}} dX, \tag{74}$$

where $X = e^{-\frac{F\Psi_s}{RT}}$, $X_{d,s}$ is the value of X at the outer Helmholtz plane, A_{surf} is the surface area (m^2), $\text{sign}(X_{d,s} - 1)$ is +1 or -1 depending on the sign of the term in parentheses, i is the aqueous species for which the surface excess is being calculated, z_i is the charge on aqueous species i, l ranges over all aqueous species, m_l is the molality and z_l is the charge of aqueous species l, and $\alpha = (\varepsilon\varepsilon_0 RT/2)^{1/2}$. The value of α at 25°C is 0.02931 $(L/mol)^{1/2}$ C m^{-2}. The relation between the unknown (X) used by Borkovec and Westall (1983) and the master unknown used by PHREEQC is $a_{\Psi_s} = X^{-2}$.

The development of Borkovec and Westall (1983) calculates only the total excess concentration in the diffuse layer of each aqueous species. A problem arises in batch-reaction and transport modeling when a solution is removed from the surface, for example, in an advection simulation when the water in one cell advects into the next cell. In this case, the total moles that remain with the surface need to be known. In PHREEQC, an arbitrary assumption is made that the diffuse layer is a specified thickness and that all of the surface excess resides in the diffuse layer. The total moles of an aqueous species in the diffuse layer are then the sum of the contributions from the surface excess plus the bulk solution in the diffuse layer:

$$n_{i,s} = n_{i,s,excess} + n_{i,s,aq} = W_{bulk} g_{i,s} \frac{n_i}{W_{aq}} + W_s \frac{n_i}{W_{aq}} \cong g_{i,s} n_i + W_s \frac{n_i}{W_{aq}}, \tag{75}$$

where $n_{i,s,aq}$ refers to the moles of aqueous species i that are present in the diffuse layer due to the contribution from the bulk solution, $n_{i,s,excess}$ refers to the surface excess, W_{aq} is the mass of water in the system excluding the diffuse layer, W_s is the mass of water in the diffuse layer of surface s. It is assumed that the amount of water in the aqueous phase is much greater than in the diffuse layers, such that $W_{bulk} \cong W_{aq}$, (In version 1, $W_{bulk} = W_{aq} + \sum_s^S W_s$). The mass of water in the diffuse layer is calculated from the thickness of the diffuse layer and the surface area, assuming 1 L contains 1 kg water:

$$W_s = t_s A_{surf}, \tag{76}$$

where t_s is the thickness of the diffuse layer in meters. If the moles of surface sites are related to the moles of a pure phase or kinetic reactant, then $A_{surf} = A_r n_r$, otherwise A_{surf} is constant and calculated from the specific area and the mass of the surface that are specified on input. According to electrostatic theory, the thickness of the diffuse layer should be greater at low ionic strength and smaller at high ionic strength. The default value used in PHREEQC for the thickness of the diffuse layer is 1×10^{-8} m, which is approximately the thickness calculated by Debyė theory for an ionic strength of 0.001 molal. For ionic strength 0.00001, the Debye length of the diffuse layer is calculated to be 1×10^{-7} m. The assumption that the amount of water in the diffuse layer is small will be

invalid if the surface area is sufficiently large; for a thickness of 1×10^{-7} m, a surface area of 1000 m^2 results in a diffuse-layer volume of 0.1 L, which is a significant portion of 1 L of bulk solution.

The total derivative of the moles of an aqueous species in the diffuse layer is

$$dn_{i,s} = \left(g_{i,s} + \frac{W_s}{W_{aq}}\right) dn_i + n_i \frac{\partial g_{i,s}}{\partial X} \left(-2e^{\left(\frac{F\Psi_s}{2RT}\right)^{-2}}\right) d\ln a_{\Psi_s} - n_i \frac{W_s}{W_{aq}} d\ln W_{aq} + n_i \frac{t_s A_r}{W_{aq}} dn_r, \tag{77}$$

where the second term is the partial derivative with respect to the master unknown for the potential at the surface, $\ln a_{\Psi_s}$, and the last term is present only if the number of surface sites is related to the moles of a pure phase or kinetic reactant. The partial derivative, $\frac{\partial g_{i,s}}{\partial X}$, is equal to the integrand from equation 74 evaluated at $X_{d,s}$:

$$\left.\frac{\partial g_{i,s}}{\partial X}\right|_{X_{d,s}} = A_{surf} \text{sign}(X_{d,s} - 1) \alpha \frac{(X_{d,s}^{z_i} - 1)}{\left[X_{d,s}^2 \sum_{l}^{N_{aq}} m_l (X_{d,s}^{z_l} - 1)\right]^{1/2}}, \tag{78}$$

and the partial derivative of the function $g_{i,s}$ with respect to the master unknown is

$$\frac{\partial g_{i,s}}{\partial \ln a_{\Psi_s}} = \frac{\partial g_{i,s}}{\partial X} \left(-2e^{\left(\frac{F\Psi_s}{2RT}\right)^{-2}}\right) = -A_{surf} \text{sign}(X_{d,s} - 1) \alpha \left(2e^{\left(\frac{F\Psi_s}{2RT}\right)^{-2}}\right) \frac{(X_{d,s}^{z_i} - 1)}{\left[X_{d,s}^2 \sum_{l}^{N_{aq}} m_l (X_{d,s}^{z_l} - 1)\right]^{1/2}}. \tag{79}$$

In the numerical method, it is computationally expensive to calculate the functions $g_{i,s}$, so the same approach as Borkovec and Westall (1983) is used in PHREEQC to reduce the number of function evaluations. A new level of iterations is added when the diffuse layer is explicitly included in the calculations. The functions and their partial derivatives are explicitly evaluated once at the beginning of each of these diffuse-layer iterations. During the model iterations, which occur within the diffuse-layer iterations, the values of the functions are updated using the following equation:

$$g_{i,s}^{k+1} = g_{i,s}^{k} + \frac{\partial g_{i,s}}{\partial \ln a_{\Psi_s}} d\ln a_{\Psi_s}, \tag{80}$$

where k is the model iteration number and $g_{i,s}^{0}$ is the value that is evaluated explicitly at the beginning of the diffuse-layer iteration. The model iterations end when the Newton-Raphson method has converged on a solution; however, convergence is based on the values of the functions $g_{i,s}$ that are estimates. Thus, diffuse-layer iterations continue until the values of the functions are the same on successive diffuse-layer iterations within a specified tolerance.

When explicitly calculating the composition of the diffuse layer, the function involving the *sinh* of the potential unknown (equation 69) is replaced with a charge-balance function that includes the surface charge and the diffuse-layer charge:

$$f_{z,s} = \sum_{k}^{K_s} \sum_{i_{(s_k)}}^{N_{s_k}} z_{i_{(s_k)}} n_{i_{(s_k)}} + \sum_{i}^{N_{aq}} z_i n_{i,s}, \tag{81}$$

where the function $f_{z,s}$ is zero when charge balance is achieved. The total derivative of $f_{z,s}$ is

$$df_{z,s} = \sum_{k}^{K_s} \sum_{i_{(s_k)}}^{N_{s_k}} z_{i_{(s_k)}} dn_{i_{(s_k)}} + \sum_{i}^{N_{aq}} z_i dn_{i,s}. \quad (82)$$

For data input to PHREEQC, explicit calculation of the diffuse layer is invoked using the **-diffuse_layer** identifier in the **SURFACE** data block. Specific surface area (A_s or A_r) and mass of surface (S_s) are defined in the **SURFACE** data block. The moles of surface sites are defined (1) in **SURFACE** if the number of sites is fixed, or (2) by a proportionality factor in the **SURFACE** data block and the moles of a phase in **EQUILIBRIUM_PHASES** data block, or (3) by a proportionality factor in the **SURFACE** data block and the moles of a kinetic reactant in **KINETICS** data block. The charge on a surface species is specified in the balanced chemical reaction that defines the species in the **SURFACE_SPECIES** data block (see "Description of Data Input").

Non-Electrostatic Surface Complexation

Davis and Kent (1990) describe a non-electrostatic surface-complexation model. In this model, the electrostatic term is ignored in the mass-action expressions for surface complexes. In addition, no surface charge-balance or surface charge-potential relation is used; only the mole-balance equation is included for each surface site type.

For data input to PHREEQC, the non-electrostatic model for a surface is invoked by using the **-no_edl** identifier in the **SURFACE** data block (see "Description of Data Input").

NUMERICAL METHOD FOR SPECIATION AND FORWARD MODELING

The formulation of any chemical equilibrium problem solved by PHREEQC is derived from the set of functions denoted f in the previous sections. These include f_{Alk}, f_e, f_g, f_H, f_{H_2O}, $f_{m'}$, f_O, $f_{P_{total}}$, f_p, $f_{p_{ss}}$, f_{s_k}, f_z, $f_{z,s}$, f_μ, and f_{Ψ_s}, where f_H and f_O are the simply the mole-balance functions for hydrogen and oxygen and m' refers to all aqueous master species except H^+, e^-, H_2O and the alkalinity master species. The corresponding set of master unknowns is $\ln a_{Alk}$, $\ln a_e$, n_g, $\ln a_{e^-}$, $\ln a_{H_2O}$, $\ln a_{m'}$, $\ln W_{aq}$, N_{gas}, n_p (or possibly $\ln a_{m'}$ in speciation calculations), n_{ss}, $\ln a_{s_k}$, $\ln a_{H^+}$ (or possibly $\ln a_{m'}$ in speciation calculations), $\ln a_{\Psi_s}$ (explicit diffuse-layer calculation), μ, and $\ln a_{\Psi_s}$ (implicit diffuse-layer calculation). When the residuals of all the functions that are included for a given calculation are equal to zero, a solution to the set of nonlinear equations has been found, and the equilibrium values for the chemical system have been determined. (Note that some equations that are initially included in a given calculation may be dropped if a pure phase or gas phase does not exist at equilibrium.) The solution technique assigns initial values to the master unknowns and then uses a modification of the Newton-Raphson method iteratively to revise the values of the master unknowns until a solution to the equations has been found within specified tolerances.

For a set of equations, $f_i = 0$, in unknowns x_j the Newton-Raphson method involves iteratively revising an initial set of values for the unknowns. Let $r_i = f_i$ be the residuals of the equations for the current values of the unknowns. The following set of equations is formulated:

$$r_i = -\sum_j^J \frac{\partial f_i}{\partial x_j} dx_j, \tag{83}$$

where J is the total number of master unknowns for the calculation. The set of equations is linear and can be solved simultaneously for the unknowns, dx_j. New values of the unknowns are calculated, ${}^{k+1}x_j = {}^{k}x_j + dx_j$, where k refers to the iteration number, after which, new values of the residuals are calculated. The process is repeated until the values of the residuals are less than a specified tolerance.

Two problems arise when using the Newton-Raphson method for chemical equilibria. The first is that the initial values of the unknowns must be sufficiently close to the equilibrium values, or the method does not converge, and the second is that a singular matrix may arise if the chemical reactions for a set of phases are not linearly independent. PHREEQC uses an optimization technique developed by Barrodale and Roberts (1980) to avoid the occurrence of singular matrices. The optimization technique also allows inequality constraints to be added to the problem, which are useful for constraining the total amounts of phases and solid solutions that can react.

The selection of initial estimates for the master unknowns is described for each type of modeling in the following sections. Regardless of the strategy for assigning the initial estimates, the estimates for the activities of the master species for elements or element valence states are revised, if necessary, before the Newton-Raphson iterations to produce approximate mole balance. The procedure for aqueous master species is as follows. After the initial estimates have been made, the distribution of species is calculated for each element (except hydrogen and oxygen) and, in initial solution calculations only, for the individual valence states which were defined. Subsequently, the ratio of the calculated moles to the input moles is calculated. If the ratio for a master species m' is greater than 1.5 or less than 10^{-5}, the following equation is used to revise the value of the master unknown:

$$\ln a_{m'}^{k+1} = \ln a_{m'}^{k} + w \ln \left(\frac{\sum_i^{N_{aq}} b_{m',i} n_i}{T_{m'}} \right), \tag{84}$$

where w is 1.0 if the ratio is greater than 1.5 and 0.3 if the ratio is less than 10^{-5}, and $T_{m'}$ is the total concentration of an element or element valence state. Analogous equations are used for exchange and surface master species. After revisions to the initial estimates, the distribution of species is calculated. The iterations continue until the ratios are within the specified ranges, at which point the modified Newton-Raphson technique is used. If the successive revisions fail to find activities such that the ratios are within the specified bounds, then a second set of iterations tries to reduce the ratios below 1.5 with no lower limit to these ratios. Whether or not the second set of iterations succeeds, the Newton-Raphson technique is then used.

The optimization technique of Barrodale and Roberts (1980) is a modification of the simplex linear programming algorithm that minimizes the sum of absolute values of residuals (L1 optimization) on a set of linear equations subject to equality and inequality constraints. The general problem can be posed with the following matrix equations:

$$\begin{aligned} AX &= B \\ CX &= D \\ EX &\leq F \ . \end{aligned} \tag{85}$$

The first matrix equation is minimized in the sense that $\sum_{i}^{I_B} \left| b_i - \sum_{j}^{J} a_{i,j} x_j \right|$ is a minimum, where I_B is the number of equations to be optimized, subject to the equality constraints of the second matrix equation and the inequality constraints of the third matrix equation.

The approach of PHREEQC is to include some of the Newton-Raphson equations in the optimization equations ($AX = B$), rather than include all of the Newton-Raphson equations as equalities ($CX = D$). Equations that are included in the A matrix may not be solved for exact equality at a given iteration, but will be optimized in the sense given above. Thus, at a given iteration, an approximate mathematical solution to the set of Newton-Raphson equations can be found even if no exact equality solution exists, for example when forcing equality for all equations would result in an unsolvable singular matrix. The equations for alkalinity, total moles of gas in the gas phase, pure phases, and solid-solution components are included in the A matrix. All mole-balance, charge-balance, and surface-potential equations are included in the B matrix. Inequalities that limit the dissolution of pure phases, solid-solution components, and gas components to the amounts present in the system are included in the C matrix.

In an attempt to avoid some numerical problems related to small numbers in the B matrix, a row of the matrix that represents a mole-balance equation is scaled if all coefficients (a column of A and B) of the corresponding unknown (change in the log activity of the element master species) are less than 1e-10. In this case, the equation is scaled by 1e-10 divided by the absolute value of the largest coefficient. Alternatively, when specified, (**-diagonal_scale** in **KNOBS**), a mole-balance equation is scaled by 1e-10 divided by the coefficient of the corresponding unknown if the coefficient of the unknown in the mole-balance equation is less than 1e-10.

The scaled matrix is solved by the optimizing solver, and the solution that is returned is a vector of changes to the values of the master unknowns. The values of the changes are checked to ensure that the changes to the unknowns are less than criteria that limit the maximum allowable size of changes. These criteria are specified by default in the program or by input in the **KNOBS** data block. If any of the changes are too large, then all the changes to the unknowns, except the mole transfers of pure phases and solid-solution components, are decreased proportionately to satisfy all of the criteria. Pure-phase and solid-solution mole transfers are not altered except to produce nonnegative values for the total moles of the pure phases and solid-solution components. After suitable changes to the unknowns have been calculated, the master unknowns are updated; new molalities and activities of all the aqueous, exchange, and surface species are calculated, and residuals for all of the functions are calculated. The residuals are tested for convergence (convergence criteria are defined internally in the program, but can be switched to an alternate set with the **-convergence_tolerance** in **KNOBS** or **-high_precision** option in **SELECTED_OUTPUT** data blocks), and a new iteration is begun if convergence has not been attained.

Aqueous Speciation Calculations

Aqueous speciation calculations use a chemical composition for a solution as input and calculate the distribution of aqueous species and saturation indices for phases. Aqueous speciation calculations include the equations $f_{m'}$, f_{H_2O}, and f_{μ}, which are equations for mole balance for elements or element valence states, activity of water, and ionic strength. Mole-balance equations for hydrogen and oxygen are not included, because the total masses of hydrogen and oxygen generally are not known. Instead, the mass of water is assumed to be 1.0 kg or is specified (**-water** in the **SOLUTION** or **SOLUTION_SPREAD** data block) and the total masses of

hydrogen and oxygen are calculated in the speciation calculation from the mass of water and the concentrations of all hydrogen and oxygen containing aqueous species.

If pH, pe, or the master unknown for an element or element valence state is specified to be adjusted to obtain charge balance for the solution, f_z is included to calculate the value of the master unknown ($\ln a_{H^+}$, $\ln a_{e^-}$, or $\ln a_{m'}$) that produces charge balance. In this case, the calculated pH, pe, or total concentration of m' will differ from the input value. If f_z is included for the master unknown $\ln a_{m'}$, the equation $f_{m'}$ is excluded.

If pH, pe, or the master unknown for an element or element valence state is specified to be adjusted to obtain a specified saturation index for a pure phase, f_p is included to calculate the value of the master unknown ($\ln a_{H^+}$, $\ln a_{e^-}$, or $\ln a_{m'}$) that produces the target saturation index. In this case, the calculated pH, pe, or total concentration of m' will differ from the input value. If f_p is included for the master unknown $\ln a_{m'}$, the equation $f_{m'}$ is excluded.

If total alkalinity is specified in the input, the mole-balance equation for alkalinity, f_{Alk}, is included to calculate $\ln a_{Alk}$ and the total molality of the element associated with alkalinity (carbon in the default database). If the problem definition contains a mole-balance equation for both carbon [or carbon(+4)] and alkalinity, then the two master unknowns associated with these equations are $\ln a_{Alk} = \ln a_{CO_3^{-2}}$ (for the default database files) and $\ln a_{H^+}$. In this case, the pH will be calculated in the speciation calculation and will not be equal to the input pH.

For speciation calculations, if the alkalinity mole-balance equation is included in the problem formulation, it is included as the only optimization equation for the solver. All other equations are included as equality constraints. No inequality constraints are included for speciation calculations.

Partial redox disequilibrium is allowed in initial solution calculations, and redox options in the **SOLUTION** or **SOLUTION_SPREAD** data block affect the aqueous speciation and saturation index calculations. By default, whenever a value of the activity of the electron is needed to calculate the molality or activity of an aqueous species, the input pe is used. If a default redox couple is given (**-redox**) or a redox couple is specified for an element (or combination of element valence states) (see **SOLUTION** keyword in "Description of Data Input"), then the mass-action expression for each aqueous species of the redox element is rewritten to remove the activity of the electron from the expression and replace it with the activities of the redox couple. For example, if iron (Fe) is to be distributed using the sulfate-sulfide redox couple [S(+6)/S(-2)], then the original chemical reaction for Fe^{+3}:

$$Fe^{+2} = Fe^{+3} + e^- \tag{86}$$

would be rewritten using the association reaction for sulfide,

$$SO_4^{-2} + 9H^+ + 8e^- = HS^- + 4H_2O, \tag{87}$$

to produce the following chemical reaction that does not include electrons:

$$Fe^{+2} + \frac{1}{8}SO_4^{-2} + \frac{9}{8}H^+ = Fe^{+3} + \frac{1}{8}HS^- + \frac{1}{2}H_2O. \tag{88}$$

The mass-action expression for this final reaction would be used as the mass-action expression for the species Fe^{+3}, and the differential for the change in the moles of Fe^{+3}, $dn_{Fe^{+3}}$, would also be based on this mass-action expression. However, the original mass-action expression (based on equation 86) is used to determine the mole-balance equations in which the term $dn_{Fe^{+3}}$ appears, that is, the species Fe^{+3} would appear in the mole-balance equation for iron, but not in the mole-balance equations for S(+6) or S(-2). The effect of these manipulations is that ferrous iron, ferric iron, sulfate, and sulfide are in redox equilibrium. Another set of redox elements (for

example oxygen and nitrogen) may also be defined to be in equilibrium among themselves, but not necessarily in redox equilibrium with iron and sulfur.

By default, if a saturation-index calculation requires a value for pe (or activity of the electron), then the input pe is used. If a default redox couple has been defined (**-redox**), then the dissolution reaction for the phase is rewritten as above to eliminate the activity of the electron and replace it with the activities of the redox couple.

The set of master unknowns may change for redox elements during a calculation. The process, which is termed "basis switching", occurs if the activity of the master species which is the master unknown for a mole-balance equation becomes ten orders of magnitude smaller than the activity of another master species included in the same mole-balance equation. In this case, all of the mass-action expressions involving the current master unknown (including aqueous, exchange, gas, and surface species, and pure phases) are rewritten in terms of the new master species that has the larger activity. An example of this process is, if nitrogen is present in a system that becomes reducing, the master unknown for nitrogen would switch from nitrate, which would be present in negligible amounts under reducing conditions, to ammonium, which would be the dominant species. Basis switching does not affect the ultimate equilibrium distribution of species, but it does speed calculations and avoid numerical problems in dealing with small concentrations.

Initial values for the master unknowns are estimated and then revised according to the strategy described in the previous section. For initial solution calculations, the input values for pH and pe are used as initial estimates. The mass of water is 1.0 kg unless otherwise specified, and the activity of water is estimated to be 1.0. Ionic strength is estimated assuming the master species are the only species present and their concentrations are equal to the input concentrations (converted to units of molality). The activity of the master species of elements (except hydrogen and oxygen) and element valence states are set equal to the input concentration (converted to molality). If the charge-balance equation or a phase-equilibrium equation is used in place of the mole-balance equation for an element or element valence state, then the initial activity of the master species is set equal to one thousandth of the input concentration (converted to molality).

For data input to PHREEQC all options for a speciation calculation--use of an alkalinity equation, charge-balance equation, phase-equilibrium equation, and redox couples--are defined in a **SOLUTION** or **SOLUTION_SPREAD** data block (see "Description of Data Input").

Calculation of the Initial Composition of an Exchanger

An initial exchange-composition calculation is needed if the composition of an exchanger is not defined explicitly, but rather, is indicated to be in equilibrium with a specified solution composition. In this case, the composition of the exchanger is not known, only that it is in equilibrium with a solution. The equations for an initial exchange-composition calculation are f_e, $f_{m'}$, f_{H_2O}, and f_μ, which are equations for mole balance for each exchanger, mole balance for each element or element valence state, activity of water, and ionic strength.

For initial exchange-composition calculations, the values of $T_{m'}$ include only the aqueous concentrations and the mole-balance equations $f_{m'}$ do not contain terms for the contribution of the exchangers to the total element concentrations. All quantities related to the aqueous phase are the same as for the solution without the exchanger present. Essentially, only the values of the master unknowns of the exchange assemblage, $\ln a_e$, are adjusted to achieve mole balance for the exchanger. Once mole balance is achieved, the composition of each exchanger is known.

All equations for initial exchange-composition calculations are included as equality constraints in the solver. No equations are optimized and no inequality constraints are included.

An initial exchange-composition calculation is performed only if the exchanger is defined to be in equilibrium with a specified solution. The distribution of species for this solution has already been calculated, either by an initial solution calculation or by a batch-reaction or transport calculation. Thus, the values of all master unknowns related to the aqueous phase are known and are used as initial estimates for the exchange calculation. The initial estimate of the master unknown for each exchanger is set equal to the moles of exchange sites for that exchanger.

For data input to PHREEQC, definition of the initial exchange-composition calculation is made with the **EXCHANGE** data block (see "Description of Data Input").

Calculation of the Initial Composition of a Surface

An initial surface-composition calculation is needed if the composition of a surface is not defined explicitly, but is indicated to be in equilibrium with a specified solution composition. In this case, the composition of the surface is not known, only that it is in equilibrium with a solution. The equations for the initial surface-composition calculation are f_{s_k}, f_{Ψ_s} or $f_{z,s}$, $f_{m'}$, f_{H_2O}, and f_{μ}, which are equations for mole-balance for each type of surface site in the surface assemblage, the charge-potential relation or charge-balance for each surface (both of these equations are excluded in the non-electrostatic model), mole balance for each element or element valence state, activity of water, and ionic strength.

For initial surface-composition calculations, the values of $T_{m'}$ include only the aqueous concentrations and the corresponding mole-balance equations $f_{m'}$ do not contain terms for the contribution of the surfaces to the total element concentrations. All quantities related to the aqueous phase are the same as for the solution without the surface assemblage present.

For the explicit calculation of the diffuse layer, a charge-balance equation is used for each surface, $f_{z,s}$; the values of the master unknowns for each surface type of the surface assemblage, $\ln a_{s_k}$ and the potential unknowns $\ln a_{\Psi_s}$, are adjusted to achieve mole balance and charge balance for each surface. If the diffuse-layer composition is not explicitly included in the calculation, then the charge-potential equation f_{Ψ_s} is used in place of the surface charge-balance equation. If the non-electrostatic model is used for the surface assemblage, then neither the surface charge-balance nor the charge-potential equation is included in the set of equations to be solved.

All equations for initial surface-composition calculations are included as equality constraints in the solver. No equations are optimized and no inequality constraints are included.

An initial surface-composition calculation is performed only if the initial surface is defined to be in equilibrium with a specified solution. The distribution of species for this solution has already been calculated, either by an initial solution calculation or by a batch-reaction or transport calculation. Thus, the values of all master unknowns related to the aqueous phase are known and are used as starting estimates for the surface calculation. The initial estimate of the activity of the master species for each surface is set equal to one tenth of the moles of surface sites for that surface. For explicit and implicit diffuse-layer calculations, the initial estimate of the potential unknown $\ln a_{\Psi_s}$ is zero for each surface, which implies that the surface charge is zero.

For data input to PHREEQC, definition of the initial surface-composition calculation is made with the **SURFACE** data block (see "Description of Data Input").

Calculation of the Initial Composition of Fixed-Volume Gas Phase

An initial gas-phase-composition calculation is needed if the composition of a gas phase is not defined explicitly, but rather, the composition of a fixed-volume gas phase is defined to be that which is in equilibrium with a specified solution composition. The equations for the initial gas-phase-composition calculation are the same as an initial solution calculation and are $f_{m'}$, f_{H_2O}, and f_{μ}, which are equations for mole balance for each element or element valence state, activity of water, and ionic strength.

For initial gas-phase-composition calculations, the values of $T_{m'}$ include only the aqueous concentrations and the corresponding mole-balance equations $f_{m'}$ do not contain terms for the contribution of the gas components to the total element concentrations. The values calculated for all quantities related to the aqueous phase are the same as for the solution without the gas phase present. Once the distribution of species in the aqueous phase is determined, the partial pressures of all components in the gas phase can be calculated. The partial pressures and the specified fixed volume are used with the ideal gas law to calculate the moles of each component in the gas phase.

All equations for initial gas-phase-composition calculations are included as equality constraints in the solver. No equations are optimized and no inequality constraints are included.

An initial gas-phase-composition calculation is performed only if the gas phase is defined to have a constant volume and is defined to be initially in equilibrium with a specified solution. The distribution of species for this solution has already been calculated, either by an initial solution calculation or by a batch-reaction or transport calculation. Thus, the values of all master unknowns related to the aqueous phase are known and are used as initial estimates for the initial gas-phase-composition calculation.

For data input to PHREEQC, definition of the initial gas-phase-composition calculation is made with the **GAS_PHASE** data block (see "Description of Data Input").

Batch-Reaction and Transport Calculations

Batch-reaction and transport calculations require calculating equilibrium between the aqueous phase and any equilibrium-phase assemblage, surface assemblage, exchanger assemblage, solid-solution assemblage, and gas phase that is defined to be present in a chemical system. Irreversible reactions that occur prior to equilibration include mixing, specified stoichiometric reactions, kinetic reactions, and temperature change. The complete set of Newton-Raphson equations that can be included in batch-reaction and transport calculations contains f_e, f_H, f_{H_2O}, $f_{m'}$, f_O, $f_{P_{total}}$, f_p, $f_{p_{ss}}$, f_{s_k}, f_z, $f_{z,s}$, f_{μ}, and f_{Ψ_s}.

Equations for mole balance on hydrogen f_H, activity of water f_{H_2O}, mole balance on oxygen f_O, charge balance f_z, and ionic strength f_{μ} are always included and are associated with the master unknowns $\ln a_{e^-}$, $\ln a_{H_2O}$, W_{aq} (mass of water), $\ln a_{H^+}$, and μ, which are always included as master unknowns.

Mole-balance equations $f_{m'}$ are included for total concentrations of elements, not individual valence states or combinations of individual valence states. A mole-balance equation for alkalinity can not be included; it is used only in initial solution calculations.

The equation $f_{P_{total}}$ is included if a fixed-pressure gas phase is specified and is present at equilibrium. The equations f_e are included if an exchange assemblage is specified. The equations f_{s_k} are included if a surface assemblage is specified. In addition, f_{Ψ_s} is included for each surface for which an implicit diffuse-layer

calculation is specified or $f_{z,s}$ is included for each surface for which an explicit diffuse-layer calculation is specified. An equation f_p is included for each pure phase that is present at equilibrium. An equation $f_{p_{ss}}$ is included for each component of each solid solution that is present at equilibrium.

It is not known at the beginning of the calculation whether a pure phase, solid solution, or fixed-pressure gas phase will be present at equilibrium. Thus, at each iteration, the following logic is used to determine which of the equations should be included in the equilibrium calculations. The equation for a phase is included if it has a positive moles, $n_p > 0$, or if the saturation index is calculated to be greater than the target saturation index. If the equation is not included in the matrix, then all coefficients for the unknown dn_p in the matrix are set to zero.

For an ideal solid solution, the equations $f_{p_{ss}}$ are included if the moles of any of the components are greater than a small number ($1x10^{-13}$) or if the sum, $\sum_{p_{ss}} \frac{IAP_{p_{ss}}}{K_{p_{ss}}}$, is greater than 1.0. For an ideal solid solution, $\frac{IAP_{p_{ss}}}{K_{p_{ss}}} = x_{p_{ss}}$, so the summation determines if the sum of the mole fractions is greater than 1.0. If the equations for a solid solution are not included in the matrix, then all coefficients for the unknowns $dn_{p_{ss}}$ in the matrix are set to zero.

For nonideal, binary solid solutions the following procedure to determine whether to include solid-solution equations is developed from the equations of Glynn and Reardon (1990, equations 37 through 48). If the moles of any of the solid-solution components are greater than a small number ($1x10^{-13}$) then all the equations for the solid solution are included. Otherwise, the aqueous activity fractions of the components are calculated from

$$x_{1,aq} = \frac{IAP_1}{IAP_1 + IAP_2} \text{ and } x_{2,aq} = \frac{IAP_2}{IAP_1 + IAP_2}, \tag{89}$$

where *IAP* is the ion activity product for the pure component. Next the mole fractions of the solids that would be in equilibrium with those aqueous activity fractions are determined by solving the following equation for x_1 and x_2 (=1-x_1):

$$x_1\lambda_1 K_1 + x_2\lambda_2 K_2 = \frac{1}{\frac{x_{1,aq}}{\lambda_1 K_1} + \frac{x_{2,aq}}{\lambda_2 K_2}}, \tag{90}$$

where x_1 and x_2 are the mole fractions in the solid phase, K_1 and K_2 are the equilibrium constants for the pure components, λ_1 and λ_2 are the activity coefficients of the components as calculated from the Guggenheim parameters for the excess free energy. This equation is highly nonlinear and is solved by first testing subintervals between 0 and 1 to find one that contains the mole fraction of component 1 that satisfies the equation and then interval halving to refine the estimate of the mole fraction. Once the mole fractions of the solid have been determined, two values of the "total activity product" ($\sum\Pi$) are calculated as follows:

$$\sum\Pi_{aq} = IAP_1 + IAP_2 \tag{91}$$

$$\text{and } \sum\Pi_{solid} = x_1\lambda_1 K_1 + x_2\lambda_2 K_2. \tag{92}$$

If $\sum\Pi_{solid} < \sum\Pi_{aq}$, then the equations for the solid solution are included, otherwise, the equations are not included. If the equations for a solid solution are not included in the matrix, all coefficients for the unknowns $dn_{p_{ss}}$ in the matrix are set to zero.

At each iteration, the equation for the sum of partial pressures of gas components in the gas phase is included for a fixed-pressure gas phase if the moles in the gas phase are greater than a small number ($1x10^{-14}$), or if the sum of the partial pressures of the gas-phase components, as calculated from the activities of aqueous species, is greater than the total pressure. If the equation for the sum of the partial pressures of gas components in the gas phase is not included in the matrix, then all coefficients of the unknown dN_g are set to zero.

Equations $f_{P_{total}}$, f_p and $f_{p_{ss}}$ are included as optimization equations in the solver. All other equations are included as equality constraints in the solver. In addition, several inequality constraints are included in the solver: (1) the value of the residual of an optimization equation f_p, which is equal to $b_p - \sum_j a_{p,j} x_j$, is constrained to be nonnegative, which maintains an estimate of saturation or undersaturation for the mineral; (2) the value of the residual of an optimization equation $f_{p_{ss}}$, which is equal to $b_{p_{ss}} - \sum_j a_{p_{ss},j} x_j$, is constrained to be nonnegative, which maintains an estimate of saturation or undersaturation for the component of the solid solution; (3) the residual of the optimization equation for $f_{P_{total}}$ is constrained to be nonnegative, which maintains a nonnegative estimate of the total gas pressure; (4) the decrease in the mass of a pure phase, dn_p, is constrained to be less than or equal to the total moles of the phase present, n_p; (5) the decrease in the mass of a component of a solid solution, $dn_{p_{ss}}$, is constrained to be less than or equal to the total moles of the component present, n_p; and (6) the decrease in the moles in the gas phase, dN_{gas}, is constrained to be less than the moles in the gas phase, N_{gas}.

Initial values for the master unknowns for the aqueous phase are taken from the previous distribution of species for the solution. If mixing of two or more solutions is involved, the initial values are the sums of the values in the solutions, weighted by their mixing factor. If exchangers or surfaces have previously been equilibrated with a solution, initial values are taken from the previous equilibration. If they have not been equilibrated with a solution, the estimates of the master unknowns are the same as those used for initial exchange-composition and initial surface-composition calculations. Initial values for the moles of each phase in the pure-phase assemblage, each component in the solid solutions in the solid-solution assemblage, and each gas component in the gas phase are set equal to the input values or the values from the last simulation in which they were saved.

For data input to PHREEQC, definition of batch-reaction and transport calculations rely on many of the data blocks. Initial conditions are defined with **SOLUTION** or **SOLUTION_SPREAD**, **EXCHANGE**, **SURFACE**, **GAS_PHASE**, **EQUILIBRIUM_PHASES**, **SOLID_SOLUTIONS**, and **USE** data blocks. Batch reactions are defined by initial conditions and with **MIX**, **KINETICS**, **REACTION**, **REACTION_TEMPERATURE**, and **USE** data blocks. Transport calculations are specified with the **ADVECTION** or the **TRANSPORT** data block (see "Description of Data Input").

NUMERICAL METHOD AND RATE EXPRESSIONS FOR CHEMICAL KINETICS

A major deficiency with geochemical equilibrium models is that minerals, organic substances, and other reactants often do not react to equilibrium in the time frame of an experiment or a model period. A kinetically controlled reaction of a solid or a nonequilibrium solute generates concentration changes of aqueous species according to the rate equation:

$$\frac{dm_i}{dt} = c_{i,k} R_k, \tag{93}$$

where $c_{i,k}$ is the stoichiometric coefficient of species i in the kinetic reaction, and R_k is the overall reaction rate for substance k (mol/kgw/s). In general, reaction rates vary with reaction progress, which leads to a set of ordinary differential equations that must be solved.

Kinetic rates have been published for numerous reactions, and for various conditions of temperature, pressure, and solution composition. However, different researchers applied different rate expressions to fit observed rates, and it is difficult to select rate expressions (which commonly have been hard coded into programs) that have sufficient generality. The problem is circumvented in PHREEQC with an embedded BASIC interpreter that allows definition of rate expressions for kinetic reactions in the input file in a general way, obviating the need for hard-coded rate expressions in the program.

Numerical Method

The rate must be integrated over a time interval, which involves calculating the changes in solution concentrations while accounting for effects on the reaction rate. Many geochemical kinetic reactions result in "stiff" sets of equations in which some rates (the time derivatives of concentration change) are changing rapidly while others are changing slowly as the reactions unfold in time. PHREEQC solves such systems by a Runge-Kutta (RK) algorithm, which integrates the rates over time. An RK scheme by Fehlberg (1969) is used, with up to 6 intermediate evaluations of the derivatives. The scheme includes an RK method of lower order to derive an error estimate. The error estimate is compared with a user-defined error tolerance to automatically decrease or increase the integration time interval to maintain the errors within the given tolerance. Furthermore, if the rates in the first three RK evaluations differ by less than the tolerance, the final rate is calculated directly and checked once more against the required tolerance. The user can specify the number of intermediate RK subintervals which are evaluated before final integration of the interval is attempted (see "Description of Data Input"). The coefficients in the scheme are from Cash and Karp (1990).

Rate Expressions

The overall rate for a kinetic reaction of minerals and other solids is:

$$R_k = r_k \frac{A_0}{V} \left(\frac{m_k}{m_{0k}} \right)^n, \tag{94}$$

where r_k is the specific rate (mol/m^2/s), A_0 is the initial surface area of the solid (m^2), V is the amount of solution (kgw), m_{0k} is the initial moles of solid, m_k is the moles of solid at a given time, and $(m_k/m_{0k})^n$ is a factor to account for changes in A_0/V during dissolution and also for selective dissolution and aging of the solid. For uniformly dissolving spheres and cubes $n = 2/3$. All calculations in PHREEQC are in moles, and the factor A_0/V must be provided by the user to obtain the appropriate scaling.

The specific rate expressions, r_k, for a selection of substances have been included in the database under keyword **RATES**. These specific rates have various forms, largely depending on the completeness of the experimental information. When information is lacking, a simple rate that is often applied is

$$r_k = k_k\left(1 - \left(\frac{IAP}{K_k}\right)^{\sigma}\right), \tag{95}$$

where k_k is an empirical constant and IAP/K_k is the saturation ratio (SR). This rate equation can be derived from transition-state theory, where the coefficient σ is related to the stoichiometry of the reaction when an activated complex is formed (Aagaard and Helgeson, 1982; Delany and others, 1986). Often, $\sigma = 1$. An advantage of this expression is that the rate equation applies for both supersaturation and undersaturation, and the rate is zero at equilibrium. The rate is constant over a large domain whenever the geochemical reaction is far from equilibrium ($IAP/K < 0.1$), and the rate approaches zero when IAP/K approaches 1.0 (equilibrium).

The rate expression may also be based on the saturation index (SI) in the following form:

$$r_k = k_k \sigma \log\left(\frac{IAP}{K_k}\right). \tag{96}$$

This rate expression has been applied with some success to dissolution of dolomite (Appelo and others, 1984).

Rate expressions often contain concentration-dependent terms. One example is the Monod equation:

$$r_k = r_{max}\left(\frac{C}{K_m + C}\right), \tag{97}$$

where r_{max} is the maximal rate, and K_m is equal to the concentration where the rate is half of the maximal rate. The Monod rate equation is commonly used for simulating the sequential steps in the oxidation of organic matter (Van Cappellen and Wang, 1996). A series of rate expressions can be developed in line with the energy yield of the oxidant; first O_2 is consumed, then NO_3^-, and successively other, more slowly operating oxidants such as Fe(III) oxides and SO_4^{-2}. The coefficients in the Monod equation can be derived from first-order rate equations for the individual processes. For degradation of organic matter (C) in soils the first-order rate equation is

$$\frac{ds_C}{dt} = -k_1 s_C, \tag{98}$$

where s_C is organic carbon content (mol/kg soil), and k_1 is the first-order decay constant (s^{-1}). The value of k_1 is approximately equal to 0.025 yr^{-1} in a temperate climate with aerobic soils (Russell, 1973), whereas in sandy aquifers in The Netherlands, where NO_3^- is the oxidant, $k_1 \approx 5e-4$ yr^{-1}. Concentrations of up to 3 μM O_2 are found in ground water even outside the redox-domain of organic degradation by O_2, and 3 μM O_2 may be taken as the concentration where the (concentration-dependent) rate for aerobic degradation equals the reaction rate for denitrification. First-order decay ($k_1 = 0.025$ yr^{-1} for 0.3 mM O_2 and $k_1 = 5e\text{-}4$ yr^{-1} for 3 μM O_2) is obtained with the coefficients $r_{max} = 1.57e\text{-}9$ s^{-1} and $K_m = 294$ μM in the Monod equation, and oxygen as the limiting solute. A similar estimate for denitrification is based on $k_1 = 5e\text{-}4$ yr^{-1} for $NO_3^- = 3$ mM and $k_1 = 1e\text{-}5$ yr^{-1} for $NO_3^- = 3$ μM, which yields $r_{max} = 1.67e\text{-}11$ s^{-1} and $K_m = 155$ μM. The combined overall Monod expression for degradation of organic carbon in a fresh-water aquifer is then:

$$R_C = 6\, s_C \left(\frac{s_C}{s_{C_0}}\right) \left\{ \frac{1.57 \times 10^{-9} m_{O_2}}{2.94 \times 10^{-4} + m_{O_2}} + \frac{1.67 \times 10^{-11} m_{NO_3^-}}{1.55 \times 10^{-4} + m_{NO_3^-}} \right\} \tag{99}$$

where the factor 6 derives from recalculating the concentration of s_C from mol/kg soil to mol/kg pore water.

A further aspect of organic matter decomposition is that a part appears to be refractory and particularly

resistant to degradation. Some models have been proposed to account for the tendency of part of the sedimentary organic carbon to survive; tentatively, a factor $\left(\frac{s_C}{s_{C_0}}\right)$ may be assumed, which makes the overall rate second order. This factor implies that a decrease to 1/10 of the original concentration results in a decrease of 1/100 in the rate of further breakdown. It must be noted that this simple factor is used to approximate a very complicated process and a more thorough treatment of the process is needed, but is also possible given the flexibility of defining rates in PHREEQC.

Still other rate expressions are based on detailed measurements in solutions with varying concentrations of the aqueous species that influence the rate. For example, Williamson and Rimstidt (1994) give a rate expression for oxidation of pyrite:

$$r_{pyrite} = 10^{-10.19} m_{O_2}^{0.5} m_{H^+}^{-0.11}, \tag{100}$$

which shows a square root dependence on the molality of oxygen, and a small increase of the rate with increase in pH. This rate is applicable for the dissolution reaction only, and only when the solution contains oxygen. It is probably inadequate when the solution approaches equilibrium or when oxygen is depleted.

An example of a more complete rate expression which applies for both dissolution and precipitation is the rate equation for calcite. Plummer and others (1978) have found that the rate can be described by the equation:

$$r_{calcite} = k_1[H^+] + k_2[CO_2] + k_3[H_2O] - k_4[Ca^{2+}][HCO_3^-], \tag{101}$$

where bracketed chemical symbols indicate activity, and the coefficients k_1, k_2 and k_3 have been determined as a function of temperature by Plummer and others (1978) from calcite dissolution experiments in CO_2-charged solutions. The rate contains a forward part r_f (first three terms of equation 101), and a backward part r_b (last term of equation 101), and thus is applicable for both dissolution and precipitation. The backward rate contains a coefficient k_4, the value of which depends on the solution composition. In a pure water-calcite system, bicarbonate concentration is approximately equal to twice the calcium concentration and the backward rate can be approximated as

$$r_b = k_4[Ca^{2+}][HCO_3^-] \approx k_4 2[Ca^{2+}]^2. \tag{102}$$

At equilibrium, $[Ca^{2+}]$ is the activity at saturation $[Ca^{2+}]_s$. Also $r_{calcite} = 0$, and therefore,

$$2k_4 = \frac{r_f}{[Ca^{2+}]_s^2}. \tag{103}$$

Combining equations 101, 102, and 103 gives:

$$r_{calcite} = r_f\left[1 - \left(\frac{[Ca^{2+}]}{[Ca^{2+}]_s^2}\right)^2\right]. \tag{104}$$

In a pure Ca-CO_2 system at constant CO_2 pressure, the ion activity product (IAP) is:

$$IAP_{calcite} = \frac{[Ca^{2+}][HCO_3^-]^2}{P_{CO_2}} \approx 4\frac{[Ca^{2+}]^3}{P_{CO_2}} \text{ and } K_{Calcite} = 4\frac{[Ca^{2+}]_s^3}{P_{CO_2}}. \tag{105}$$

Thus, for a calcite-water system, the rate for calcite can be approximated as:

$$r_{calcite} \approx r_f \left[1 - \left(\frac{IAP}{K_{calcite}} \right)^{\frac{2}{3}} \right] \tag{106}$$

where r_f contains the first three terms given in equation 101.

EQUATIONS AND NUMERICAL METHOD FOR TRANSPORT MODELING

PHREEQC has the capability to model several one-dimensional transport processes including: (1) diffusion, (2) advection, (3) advection and dispersion, and (4) advection and dispersion with diffusion into stagnant zones, which is referred to as dual porosity. All of these processes can be combined with equilibrium and kinetic chemical reactions.

The Advection-Reaction-Dispersion Equation

Conservation of mass for a chemical that is transported (fig. 1) yields the advection-reaction-dispersion (ARD) equation:

$$\frac{\partial C}{\partial t} = -v\frac{\partial C}{\partial x} + D_L\frac{\partial^2 C}{\partial x^2} - \frac{\partial q}{\partial t}, \tag{107}$$

where C is concentration in water (mol/kgw), t is time (s), v is pore water flow velocity (m/s), x is distance (m), D_L is the hydrodynamic dispersion coefficient [m^2/s, $D_L = D_e + \alpha_L v$, with D_e the effective diffusion coefficient, and α_L the dispersivity (m)], and q is concentration in the solid phase (expressed as mol/kgw in the pores). The term $-v\frac{\partial C}{\partial x}$ represents advective transport, $D_L\frac{\partial^2 C}{\partial x^2}$ represents dispersive transport, and $\frac{\partial q}{\partial t}$ is the change in concentration in the solid phase due to reactions (q in the same units as C). The usual assumption is that v and D_L are equal for all solute species, so that C can be the total dissolved concentration for an element, including all redox species.

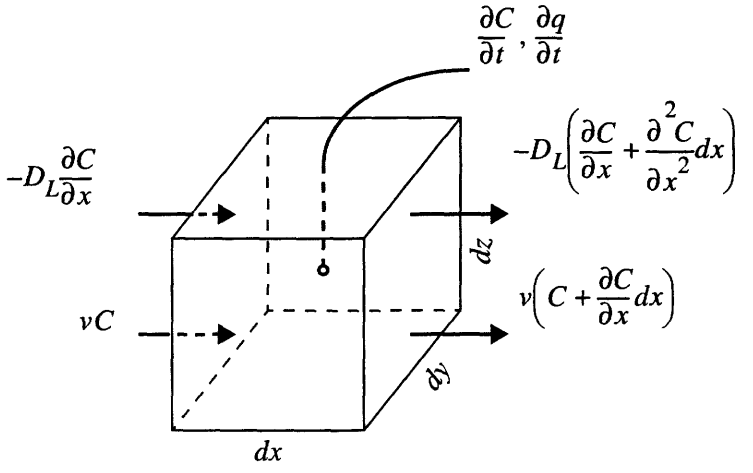

Figure 1.--Terms in the advection-reaction-dispersion equation.

The transport part of equation 107 is solved with an explicit finite difference scheme that is forward in time, central in space for dispersion, and upwind for advective transport. The chemical interaction term $\partial q/\partial t$ for each element is calculated separately from the transport part for each time step and is the sum of all equilibrium and non-equilibrium reaction rates. The numerical approach follows the basic components of the ARD equation in a split-operator scheme (Press and others, 1992; Yanenko, 1971). With each time step, first advective transport is calculated, then all equilibrium and kinetically controlled chemical reactions, thereafter dispersive transport, which is followed again by calculation of all equilibrium and kinetically controlled chemical reactions. The scheme differs from the majority of other hydrogeochemical transport models (Yeh and Tripathi, 1989) in that kinetic and equilibrium chemical reactions are calculated both after the advection step and after the dispersion step. This reduces numerical dispersion and the need to iterate between chemistry and transport.

A major advantage of the split-operator scheme is that numerical accuracy and stability can be obtained by adjusting time step to grid size for the individual parts of the equation. Numerical dispersion is minimized by always having the following relationship between time and distance discretization:

$$(\Delta t)_A = \frac{\Delta x}{v}, \tag{108}$$

where $(\Delta t)_A$ is the time step for advective transport, and Δx is the cell length. Numerical instabilities (oscillations) in the calculation of diffusion/dispersion are eliminated with the constraint:

$$(\Delta t)_D \leq \frac{(\Delta x)^2}{3D_L}, \tag{109}$$

where $(\Delta t)_D$ is the time step (s) for dispersive/diffusive transport calculations. The two conditions of equation 108 and 109 are the Courant condition for advective transport and the Von Neumann criterion for dispersive transport calculations, respectively (for example, Press and others, 1992). Numerical dispersion is in many cases negligible when $\Delta x \leq \alpha_L$, because physical dispersive transport is then equally or more important than advective transport. When a fine grid is used to reduce numerical dispersion, the time step for dispersive transport calculations (equation 109) may become smaller than the time step for advective calculations (equation 108), because the first has quadratic dependence on grid size. The conflict is solved by multiple dispersion time steps such that $\sum(\Delta t)_D = (\Delta t)_A$, and calculating chemical reactions after each of the dispersion time steps. For input to PHREEQC, a time step must be defined which equals the advection time step $(\Delta t)_A$, or, if diffusion is modeled, equals the diffusion period. Furthermore, the number of *shifts* must be defined, which is the number of advection time steps (or diffusion periods) to be calculated.

Dispersive transport in a central difference scheme is essentially mixing of cells. A mixing factor *mixf* is defined as

$$mixf = \frac{D_L(\Delta t)_A}{n(\Delta x)^2}, \tag{110}$$

where n is a positive integer. The restriction is that never more is mixed out of a cell than stays behind, that is, *mixf* must be less than 1/3 as follows from equation 109. When, according to equation 110 with $n = 1$, *mixf* is greater than 1/3, the value of n is increased such that *mixf* is less than or equal to 1/3. The dispersion time step is then $(\Delta t)_D = \frac{(\Delta t)_A}{n}$ and n mixes are performed.

The numerical scheme has been checked by comparison with analytical solutions for simple cases with linear exchange. Linear exchange results when the exchange coefficient for the exchange half-reaction is equal for two homovalent cations. It gives a linear retardation $R = 1 + CEC / C$, where CEC is the cation exchange capacity, expressed in mol/kgw. In the following example, a 130 m flow tube contains water with an initial concentration $C(x,0) = C_i = 0$. The displacing solution has concentration $C = C_0 = 1$ mmol/kgw, and the pore-water flow velocity is $v = 15$ m/year. The dispersivity is $\alpha_L = 5$ m, and the effective diffusion coefficient is $D_e = 0$ m^2/s. The profile is given after 4 years for two chemicals, one with $R = 1$ (Cl^-) and the other with $R = 2.5$ (Na^+).

Two boundary conditions can be considered for this problem. One entails a flux or third type boundary condition at $x = 0$:

$$C(0, t) = C_0 + \frac{D_L}{v}\frac{\partial C(x_{end}, t)}{\partial x}. \qquad (111)$$

This boundary condition is appropriate for laboratory columns with inlet tubing much smaller than the column cross section. The solution for the ARD equation is then (Lindstrom and others, 1967):

$$C(x, t) = C_i + \frac{1}{2}(C_0 - C_i)\, A, \qquad (112)$$

where, with $D_L = \alpha_L v$:

$$A = \operatorname{erfc}\left(\frac{x - vt/R}{\sqrt{4\alpha_L vt/R}}\right) + \sqrt{\frac{x}{\pi\alpha_L}}\exp\left[-\frac{(x - vt/R)^2}{4\alpha_L vt/R}\right] - \frac{1}{2}\left(1 + \frac{x}{\alpha_L} + \frac{vt/R}{\alpha_L}\right)\exp\left(\frac{x}{\alpha_L}\right)\operatorname{erfc}\left(\frac{x + vt/R}{\sqrt{4\alpha_L vt/R}}\right) \qquad (113)$$

Figure 2 shows the comparison for three simulations with different grid spacings, $\Delta x = 15$, 5, and 1.67 m, which correspond with $(\Delta t)_A = 1$, 1/3, and 1/9 years, respectively. For Cl^-, which has $R = 1$, the fronts of the three simulations are indistinguishable and in excellent agreement with the analytical solution. For the retarded ion Na^+, which has $R = 2.5$, the average location of the breakthrough curve for all grid spacings is correct and is in agreement with the analytical solution. However, the simulations with coarser grids show a more spread-out breakthrough that is due to numerical dispersion. The finest grid gives the closest agreement with the analytical solution, but requires the most computer time.

Computer time is primarily dependent on the number of calls to the geochemical subroutines of PHREEQC, and in the absence of kinetic reactions, the number of calls is proportional to (number of cells) x (number of advection steps) x (1 + number of dispersion steps). In this example, $D_L = D_e + \alpha_L v = 0 + 5 \times 15$ m^2/yr. Thus, by equation 110, *mixf* = 1/3, 1, and 3, respectively for the progressively smaller cell sizes. For the 15-meter cell-size (*mixf* = 1/3), one dispersion step is taken for each advection step; for the 5-meter cell size (*mixf* = 1), three dispersion steps are taken for each advection step; and for the 1.67-meter cell size (*mixf* = 3), nine dispersion steps are taken for each advection step. Figure 2 shows profiles the advective front of Cl ($C/C_0 = 0.5$) after 4 years of travel, when it has arrived at 60 m; for the 15-meter cell size, this requires 4 advection steps. The flowtube consists of 9 cells for which geochemical calculations are done for each step; therefore, the number of the reaction calculations is 9 x 4 x (1 + 1) = 72. Larger numbers of cells and advection steps apply for the smaller grids. The number of calls to the reaction calculations for the other two cases is 27 x 12 x (1 + 3) = 1,296; and 81 x 36 x (1 + 9) = 29,160.

The examples given here have linear retardation to enable comparison with analytical solutions. However, linear retardation is subject to large numerical dispersion, and the examples are, in a sense, worst cases with respect

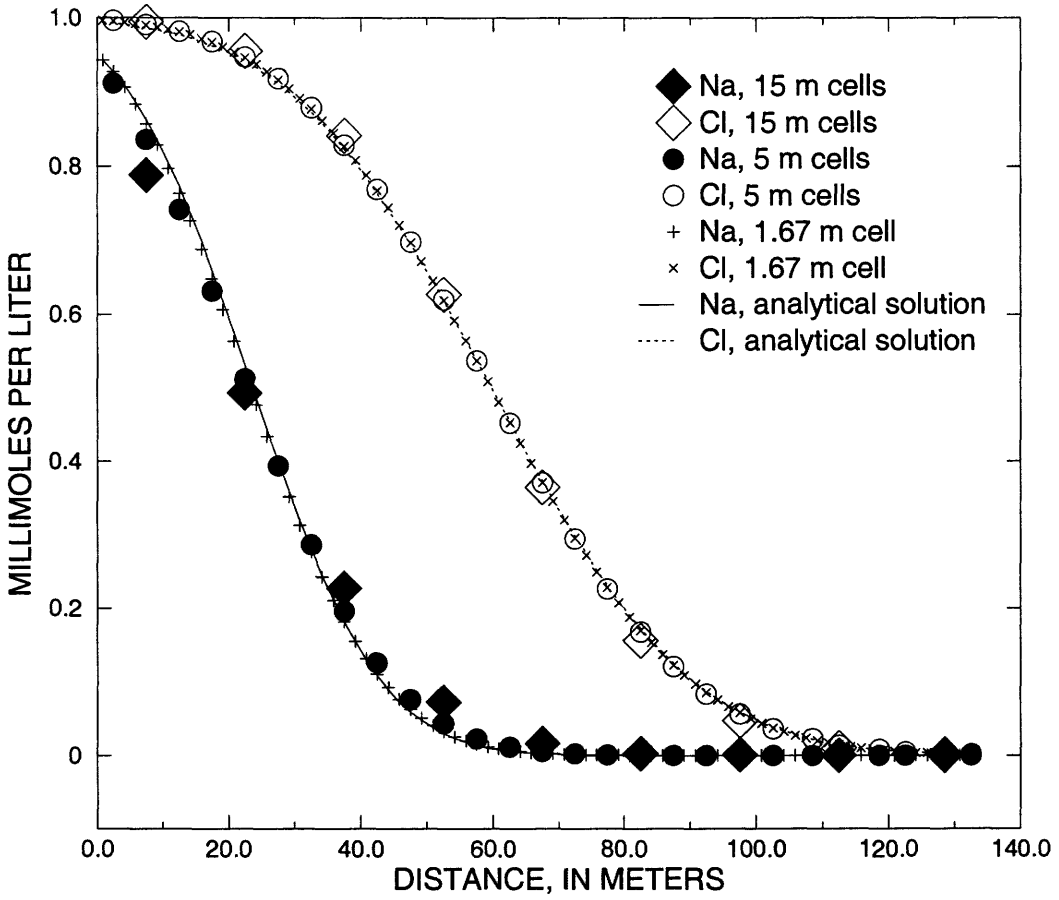

Figure 2.--Analytical solution for 1D transport with ion-exchange reactions and flux boundary condition compared with PHREEQC calculations at various grid spacings.

to numerical dispersion. In many cases of geochemical interest, the chemical reactions help to counteract numerical dispersion because the reactions tend to sharpen fronts, for example with precipitation/dissolution reactions and displacement chromatography. In other cases, exchange with a less favored ion may give a real, chemical dispersion that exceeds the effects of numerical dispersion.

Another boundary condition is the Dirichlet, or first-type, boundary condition, which involves a constant concentration $C(0,t)$ at $x = 0$:

$$C(0, t) = C_0. \tag{114}$$

This boundary condition is valid for water infiltrating from a large reservoir in full contact with the underlying soil, for example infiltration from a pond, or diffusion of seawater into underlying sediment. The solution for the ARD equation is in this case (Lapidus and Amundson, 1952):

$$C(x, t) = C_i + \frac{1}{2}(C_0 - C_i)\, B, \tag{115}$$

where,

$$B = \operatorname{erfc}\left(\frac{x - vt/R}{\sqrt{4\alpha_L vt/R}}\right) + \exp\left(\frac{x}{\alpha_L}\right)\operatorname{erfc}\left(\frac{x + vt/R}{\sqrt{4\alpha_L vt/R}}\right). \tag{116}$$

Figure 3 shows the results of three simulations with the same discretizations as the previous transport example. Again, the conservative solute (Cl^- with $R = 1$) is modeled accurately for all three grid sizes. The retarded chemical (Na^+, $R = 2.5$) shows numerical dispersion for the coarser grids, but again, the average front locations agree. With the constant concentration-boundary condition, the number of dispersion time steps is twice the number for the flux case because of the specified condition at $x = 0$. Also the effect of the first-type boundary condition is to increase diffusion over the contact surface of the column with the outer solution. The flux of chemical over the boundary is correspondingly larger and the fronts have progressed a few meters further than in figure 2. More comparisons of analytical solutions are given in the discussion of example 11 (breakthrough at the outlet of a column) and example 12 (diffusion from a constant source).

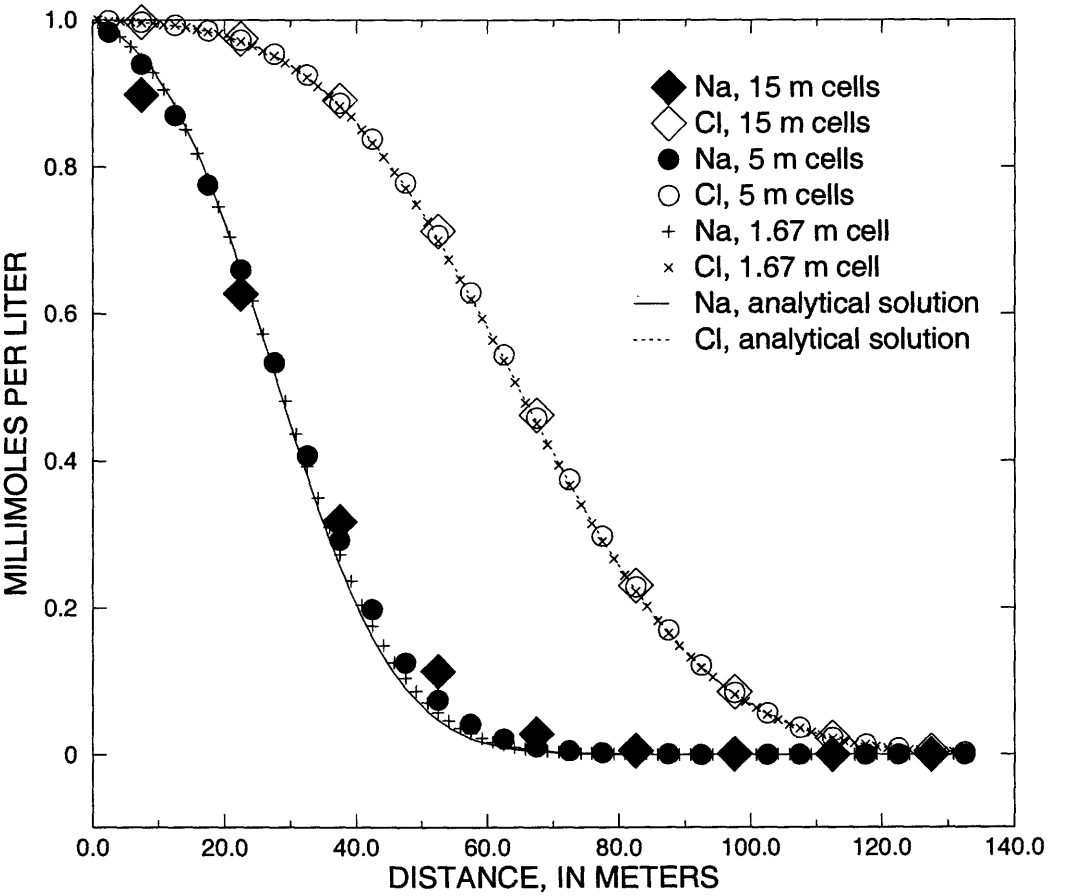

Figure 3.--Analytical solution for 1D transport with ion-exchange reactions and constant boundary condition compared with PHREEQC calculations at various grid spacings.

Transport of Heat

Conservation of heat yields the transport equation for heat, or rather, for the change of temperature. The equation is identical to the advection-reaction-dispersion equation for a chemical substance:

$$(\theta \rho_w k_w)\frac{\partial T}{\partial t} + (1-\theta)\rho_s k_s \frac{\partial T}{\partial t} = -(\theta \rho_w k_w) v \frac{\partial T}{\partial x} + \kappa \frac{\partial^2 T}{\partial x^2}, \quad (117)$$

where T is the temperature (°C), θ is the porosity (a fraction of total volume, unitless), ρ is the density (kg/m^3), k is the specific heat (kJ°C^{-1}kg^{-1}), κ is a term which entails both the dispersion by advective flow and the heat conductivity of the aquifer (kJ°C^{-1}m^{-1}s^{-1}), and subscripts w and s indicate water and solid, respectively. The temperature T is assumed to be uniform over the volume of water and solid.

Dividing equation 117 by $\theta \rho_w k_w$ gives:

$$R_T \frac{\partial T}{\partial t} = -v \frac{\partial T}{\partial x} + \kappa_L \frac{\partial^2 T}{\partial x^2}, \tag{118}$$

where $R_T = 1 + \frac{(1-\theta)\,\rho_s k_s}{\theta\,\rho_w k_w}$ is the temperature retardation factor (unitless), and $\kappa_L = \frac{\kappa}{\theta\,\rho_w k_w}$ is the thermal dispersion coefficient. The thermal dispersion coefficient contains a component for pure diffusion, and a component for dispersion due to advection: $\kappa_L = \kappa_e + \beta_L v$, similar to the hydrodynamic dispersion coefficient. The analogy permits the use of the same numerical scheme for both mass and heat transport.

De Marsily (1986) suggests that the thermal dispersivity β_L and the hydrodynamic dispersivity α_L may be equal, whereas the thermal diffusion coefficient κ_e is orders of magnitude larger than D_e. Thus, dispersion due to advection can be calculated in the same algorithm for both mass and heat, while thermal diffusion may require an additional calculation when it exceeds hydrodynamic diffusion. When temperatures are different in the column, and when the thermal diffusion coefficient is larger than the hydrodynamic diffusion coefficient, PHREEQC first calculates, for one time step, the temperature distribution and the chemical reactions due to thermal diffusion in excess of the hydrodynamic diffusion. Subsequently PHREEQC calculates transport for the combination of heat and mass due to hydrodynamic diffusion for the time step. The temperature retardation factor and the thermal diffusion coefficient must be defined in the input file (identifier **-thermal_diffusion** in keyword **TRANSPORT**). Both parameters may vary in time, but are uniform (and temperature independent) over the flow domain.

The similarity between thermal and hydrodynamic transport is an approximation which mainly falls short because diffusion of mass is by orders of magnitude larger in water than in minerals, whereas diffusion of heat is comparable in the two media although often anisotropic in minerals. The (small) difference in thermal diffusivity leads to complicated heat transfer at phase boundaries which is not accounted for by PHREEQC. Also, PHREEQC does not consider the convection that may develop in response to temperature gradients.

Transport in Dual Porosity Media

Water in structured soils and in solid rock has often a dual character with regard to flow: part of the water is mobile and flows along the conduits (continuous joints, fractures, connected porosity), while another part remains immobile or stagnant within the structural units. Exchange of water and solutes between the two parts may occur through diffusion. Dual porosity media can be simulated in PHREEQC either with the first-order exchange approximation or with finite differences for diffusion in the stagnant zone.

First-Order Exchange Approximation

Diffusive exchange between mobile and immobile water can be formulated in terms of a mixing process between mobile and stagnant cells. In the following derivation, one stagnant cell is associated with one mobile cell. The first-order rate expression for diffusive exchange is

$$\frac{dM_{im}}{dt} = \theta_{im} R_{im} \frac{dC_{im}}{dt} = \alpha(C_m - C_{im}), \tag{119}$$

where subscript m indicates mobile and im indicates immobile, M_{im} are moles of chemical in the immobile zone, θ_{im} is porosity of the stagnant (immobile) zone (a fraction of total volume, unitless), R_{im} is retardation in the stagnant zone (unitless), C_{im} is the concentration in stagnant water (mol/kgw), t is time (s), C_m is the concentration in mobile water (mol/kgw), and α is the exchange factor (s^{-1}). The retardation is equal to $R = 1 + dq/dC$, which is calculated implicitly by PHREEQC through the geochemical reactions. The retardation contains the change dq in concentration of the chemical in the solid due to *all* chemical processes including exchange, surface complexation, kinetic and mineral reactions; it may be non-linear with solute concentration and it may vary over time for the same concentration.

The equation can be integrated with the following initial conditions:

$C_{im} = C_{im_0}$ and $C_m = C_{m_0}$, at $t = 0$, and by using the mole-balance condition:

$$C_m = C_{m_0} - (C_{im} - C_{im_0}) \frac{R_{im}\theta_{im}}{R_m \theta_m}.$$

The integrated form of equation 119 is then:

$$C_{im} = \beta f C_{m_0} + (1 - \beta f) C_{im_0}, \tag{120}$$

where $\beta = \dfrac{R_m \theta_m}{R_m \theta_m + R_{im} \theta_{im}}$, $f = 1 - \exp\left(-\dfrac{\alpha t}{\beta \theta_{im} R_{im}}\right)$, θ_m is the water filled porosity of the mobile part (a fraction of total volume, unitless), and R_m is the retardation in the mobile area.

A mixing factor, $mixf_{im}$, can be defined that is a constant for a given time t:

$$mixf_{im} = \beta f. \tag{121}$$

When $mixf_{im}$ is entered in equation 120, the first-order exchange is shown to be a simple mixing process in which fractions of two solutions mix:

$$C_{im} = mixf_{im} C_{m_0} + (1 - mixf_{im}) C_{im_0}. \tag{122}$$

Similarly, an equivalent mixing factor, $mixf_m$, for the mobile zone concentrations is obtained with the mole-balance equation:

$$mixf_m = mixf_{im} \frac{R_{im}\theta_{im}}{R_m \theta_m} \tag{123}$$

and the concentration of C_m at time t is

$$C_m = (1 - mixf_m) C_{m_0} + mixf_m C_{im_0}. \tag{124}$$

The exchange factor α can be related to specific geometries of the stagnant zone (Van Genuchten, 1985). For example, when the geometry is spherical, the relation is

$$\alpha = \frac{D_e \theta_{im}}{(a\, f_{s \to l})^2}, \tag{125}$$

where D_e is the diffusion coefficient in the sphere (m^2/s), a is the radius of the sphere (m), and $f_{s\rightarrow 1}$ is a shape factor for sphere-to-first-order-model conversion (unitless). Other geometries can likewise be transformed to a value for α using other shape factors (Van Genuchten, 1985). These shape factors are given in table 1.

An analytical solution is known for a pulse input in a medium with first-order mass transfer between mobile and stagnant water (Van Genuchten, 1985; Toride and others, 1993); example 13 defines a simulation that can be compared with the analytical solution. A 2 m column is discretized in 20 cells of 0.1 m. The resident solution is 1 mM KNO_3 in both the mobile and the stagnant zone. An exchange complex of 1 mM is defined, and exchange coefficients are adapted to give linear retardation $R = 2$ for Na^+. A pulse that lasts for 5 shifts of 1 mM NaCl is followed by 10 shifts of 1 mM KNO_3. The Cl ($R = 1$) and Na ($R = 2$) profiles are calculated as a function of depth.

The transport variables are $\theta_m = 0.3$; $\theta_{im} = 0.1$; $v_m = 0.1 / 3600 = 2.778\text{e-}5$ m/s; and $\alpha_L = 0.015$ m. The stagnant zone consists of spheres with radius a = 0.01 m, diffusion coefficient $D_e = 3.\text{e-}10$ m^2/s, and a shape factor $f_{s\rightarrow 1} = 0.21$. This gives an exchange factor $\alpha_L = 6.8\text{e-}6$ s^{-1}. In the PHREEQC input file α, θ_m, and θ_{im} must be given; R_m and R_{im} are calculated implicitly by PHREEQC through the geochemical reactions.

Figure 4 shows the comparison of PHREEQC with the analytical solution (obtained with CXTFIT, version 2, Toride and others, 1995). The agreement is excellent for Cl^- ($R = 1$), but the simulation shows numerical dispersion for Na^+ ($R = 2$). When the grid is made finer so that Δx is equal to or smaller than α_L (0.015 m), numerical dispersion is much reduced. In the figure, the effect of a stagnant zone is to make the shape of the pulse asymmetrical. The leading edge is steeper than the trailing edge, where a slow release of chemical from the stagnant zone maintains higher concentrations for a longer period of time.

Finite Differences for the Stagnant Zone

As an alternative to first-order exchange of stagnant and mobile zones, a finite difference grid can be laid over the stagnant region. Fick's diffusion equations, $F = -D_e \nabla C$ and $\frac{\partial C}{\partial t} = -\nabla \bullet F$, transform to finite differences for an arbitrarily shaped cell j:

$$C_j^{t2} = C_j^{t1} + D_e \Delta t \sum_{i \neq j}^{n} \frac{A_{ij}}{h_{ij} V_j} (C_i^{t1} - C_j^{t1}) f_{bc}, \tag{126}$$

where C_j^{t1} is the concentration in cell j at the current time, C_j^{t2} is the concentration in cell j after the time step, Δt is the time step [s, equal to $(\Delta t)_D$ in PHREEQC], i is an adjacent cell, A_{ij} is shared surface area of cell i and j (m^2), h_{ij} is the distance between midpoints of cells i and j (m), V_j is the volume of cell j (m^3), and f_{bc} is a factor for boundary cells (-). The summation is for all cells (up to n) adjacent to j. When A_{ij} and h_{ij} are equal for all cells, a central difference algorithm is obtained that has second-order accuracy $[O(h)^2]$. It is therefore advantageous to make the grid regular.

The correction factor f_{bc} depends on the ratio of the volume of the mobile zone, V_m, to the volume of the boundary cell which contacts the mobile zone, V_{bc}. When the two volumes are equal, $f_{bc} = 1$. It can be shown that $f_{bc} \rightarrow 2$ when $V_m \rightarrow \infty$ (or if the concentration is constant in the mobile region, Appelo and Postma, 1993, p. 376). Likewise, $f_{bc} = 0$ when $V_m = 0$. To a good approximation therefore,

$$f_{bc} = 2 \frac{V_m}{V_m + V_{bc}}. \tag{127}$$

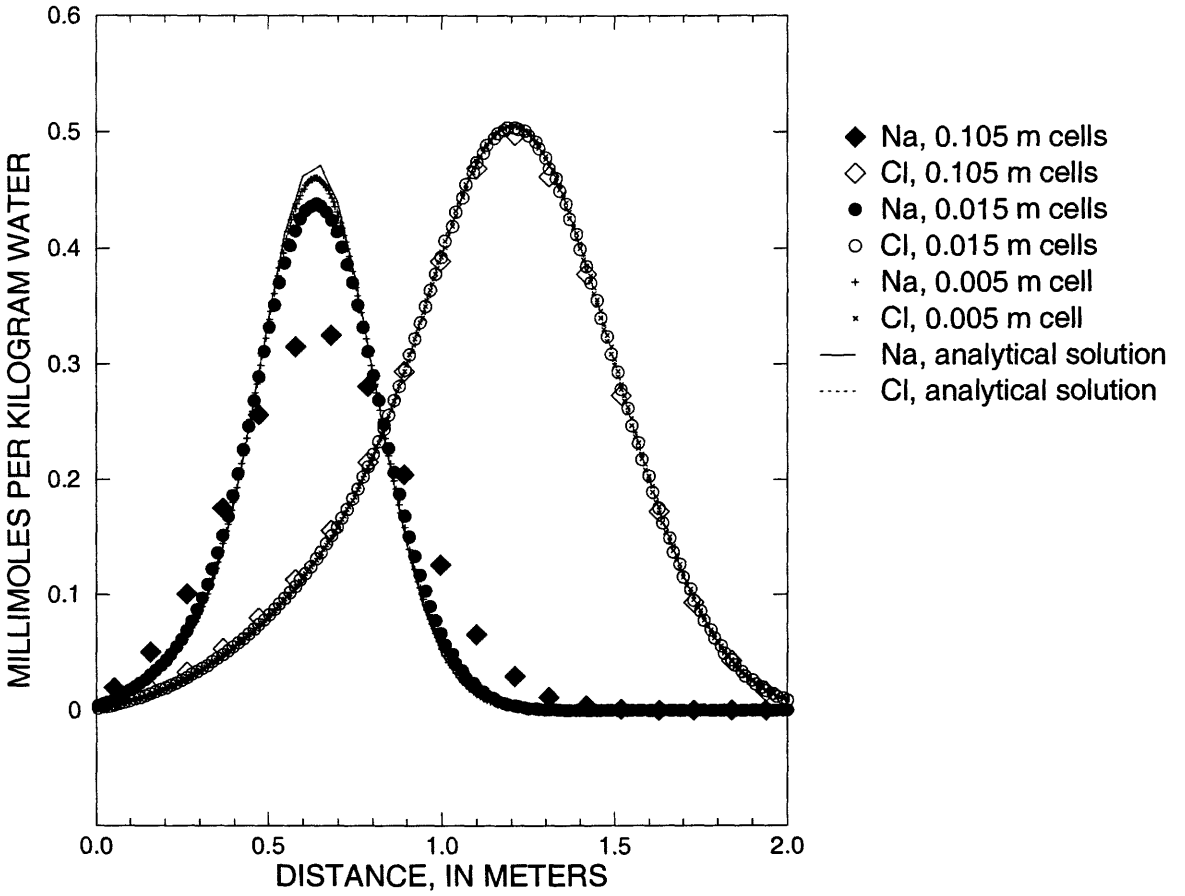

Figure 4.--Analytical solution for transport with stagnant zones, a pulse input, and ion-exchange reactions compared with PHREEQC calculations at various grid spacings.

Equation 126 can be restated in terms of mixing factors for combinations of adjacent cells. For an adjacent cell, the mixing factor contains the terms which multiply the concentration difference $(C_i - C_j)$,

$$mixf_{ij} = \frac{D_e \Delta t A_{ij} f_{bc}}{h_{ij} V_j} \tag{128}$$

and for the central cell, the mixing factor is

$$mixf_{jj} = 1 - D_e \Delta t \sum_{i \neq j}^{n} \frac{A_{ij} f_{bc}}{h_{ij} V_j}, \tag{129}$$

which give in equation 126:

$$C_j^{t2} = mixf_{jj}\, C_j^{t1} + \sum_{i \neq j}^{n} mixf_{ij}\, C_i^{t1}. \tag{130}$$

It is necessary that $0 < mixf < 1$ to prevent numerical oscillations. If any $mixf$ is outside the range, the grid of mobile and stagnant cells must be adapted. Generally, this requires a reduction of Δt, which can be achieved by increasing the number of mobile cells. An example calculation is given in example 13, where the stagnant zone consists of spheres.

Table 1.--Shape factors for diffusive first-order exchange between cells with mobile and immobile water

Shape of stagnant region	Dimensions (x, y, z) or ($2r$, z)	First-order equivalent $f_{s\to 1}$	Comments
Sphere	$2a$	0.210	$2a$ = diameter
Plane sheet	$2a$, ∞, ∞	0.533	$2a$ = thickness
Rectangular prism	$2a$, $2a$, ∞	0.312	rectangle
	$2a$, $2a$, $16a$	0.298	
	$2a$, $2a$, $8a$	0.285	
	$2a$, $2a$, $6a$	0.277	
	$2a$, $2a$, $4a$	0.261	
	$2a$, $2a$, $3a$	0.246	
	$2a$, $2a$, $2a$	0.220	cube $2a$X$2a$X$2a$
	$2a$, $2a$, $4a/3$	0.187	
	$2a$, $2a$, a	0.162	
	$2a$, $2a$, $2a/3$	0.126	
	$2a$, $2a$, $2a/4$	0.103	
	$2a$, $2a$, $2a/6$	0.0748	
	$2a$, $2a$, $2a/8$	0.0586	
Solid cylinder	$2a$, ∞	0.302	$2a$ = diameter
	$2a$, $16a$	0.289	
	$2a$, $8a$	0.277	
	$2a$, $6a$	0.270	
	$2a$, $4a$	0.255	
	$2a$, $3a$	0.241	
	$2a$, $2a$	0.216	
	$2a$, $4a/3$	0.185	
	$2a$, a	0.161	
	$2a$, $2a/3$	0.126	
	$2a$, $2a/4$	0.103	
	$2a$, $2a/6$	0.0747	
	$2a$, $2a/8$	0.0585	
Pipe wall	$2r_i$, $2r_o$, ∞		$2r_i$ = pore diameter
(surrounds the	$2a$, $4a$	0.657	$2r_o$ = outer diameter
mobile pore)	$2a$, $10a$	0.838	of pipe (Enter wall
	$2a$, $20a$	0.976	thickness $r_o - r_i = a$ in
	$2a$, $40a$	1.11	Equation 125).
	$2a$, $100a$	1.28	
	$2a$, $200a$	1.40	
	$2a$, $400a$	1.51	

For data input to PHREEQC, 1D transport including only advection is accomplished with the **ADVECTION** data block. All other 1D transport calculations, including diffusion, advection and dispersion, and advection and dispersion in a dual porosity medium, require the **TRANSPORT** data block. Initial conditions of the transport column are defined with **SOLUTION** (or **SOLUTION_SPREAD**), **EQUILIBRIUM_PHASES**, **EXCHANGE**, **GAS_PHASE**, **SOLID_SOLUTIONS**, and **SURFACE** data blocks. Kinetic reactions are defined with **KINETICS** data blocks. Infilling solutions are defined with **SOLUTION** (or **SOLUTION_SPREAD**) data blocks (see "Description of Data Input").

EQUATIONS AND NUMERICAL METHOD FOR INVERSE MODELING

PHREEQC has capabilities for geochemical inverse modeling, which attempts to account for the chemical changes that occur as a water evolves along a flow path (Plummer and Back, 1980; Parkhurst and others, 1982; Plummer and others, 1991, Plummer and others, 1994). In inverse modeling, one aqueous solution is assumed to mix with other aqueous solutions and to react with minerals and gases to produce the observed composition of a second aqueous solution. Inverse modeling calculates mixing fractions for the aqueous solutions and mole transfers of the gases and minerals that produce the composition of the second aqueous solution. The basic approach in inverse modeling is to solve a set of linear equalities that account for the changes in the moles of each element by the dissolution or precipitation of minerals (Garrels and Mackenzie, 1967, Parkhurst and others, 1982). Previous approaches have also included equations to account for mixing, conservation of electrons, which forces oxidative reactions to balance reductive reactions, and isotope mole balance (Plummer and Back, 1980; Parkhurst and others, 1982; Plummer and others, 1983; Plummer, 1984; Plummer and others, 1990; Plummer and others, 1991; and Plummer and others, 1994).

Equations and Inequality Constraints

PHREEQC expands on previous approaches by the inclusion of a more complete set of mole-balance equations and the addition of inequality constraints that allow for uncertainties in the analytical data. Mole-balance equations are included for (1) each element or, for a redox-active element, each valence state of the element, (2) alkalinity, (3) electrons, which allows redox processes to be modeled, (4) water, which allows for evaporation and dilution and accounts for water gained or lost from minerals, and (5) each isotope (Parkhurst, 1997). Also included are (6) a charge-balance equation for each aqueous solution, and (7) an equation that relates uncertainty terms for pH, alkalinity, and total dissolved inorganic carbon for each solution. Furthermore, inequalities are used (8) to constrain the size of the uncertainty terms within specified limits, and (9) to constrain the sign of the mole transfer of reactants.

The unknowns for this set of equations and inequalities are (1) the mixing fraction of each aqueous solution α_q, (2) the mole transfers of minerals and gases into or out of the aqueous solution α_p, (3) the aqueous mole transfers between valence states of each redox element α_r (the number of redox reactions for each redox element is the number of valence states minus one), and (4) a set of uncertainty terms that account for uncertainties in the analytical data $\delta_{m,q}$. Unlike previous approaches to inverse modeling, uncertainties are assumed to be present in the analytical data, as evidenced by the charge imbalances found in all water analyses. Thus, the uncertainty terms $\delta_{m,q}$ represent uncertainties due to analytical error and spatial or temporal variability in concentration of each

element, element valence state, or alkalinity, m, in each aqueous solution q. The uncertainty terms can be constrained to be less than specified uncertainty limits, $u_{m,q}$, which allows user-supplied estimates of uncertainty for each element or element valence state to limit the deviation from the analytical data ($T_{m,q}$) of revised element concentrations ($T_{m,q} + \delta_{m,q}$) that are calculated in mole-balance models.

Mole-Balance Equations

The mole-balance equations, including the uncertainty terms and redox reactions, for elements and valence states are defined as

$$\sum_{q}^{Q} c_q \alpha_q (T_{m,q} + \delta_{m,q}) + \sum_{p}^{P} c_{m,p} \alpha_p + \sum_{r}^{R} c_{m,r} \alpha_r = 0, \tag{131}$$

where Q indicates the number of aqueous solutions that are included in the calculation, $T_{m,q}$ is the total moles of element or element valence state m in aqueous solution q, $\delta_{m,q}$ can be positive or negative, $c_{m,p}$ is the coefficient of master species m in the dissolution reaction for phase p (by convention, all chemical reactions for phases are written as dissolution reactions; precipitation in mole-balance models is indicated by negative mole transfers, $\alpha_p < 0$), P is the total number of reactive phases, $c_{m,r}$ is the stoichiometric coefficient of secondary master species m in redox reaction r, and R is the total number of aqueous redox reactions. The last aqueous solution, number Q, is assumed to be formed from mixing the first Q-1 aqueous solutions, or, $c_q = 1$ for $q < Q$ and $c_Q = -1$.

For PHREEQC, redox reactions are taken from the reactions for secondary master species defined in **SOLUTION_SPECIES** input data blocks. Dissolution reactions for the phases are derived from chemical reactions defined in **PHASES** and **EXCHANGE_SPECIES** input data blocks (see "Description of Data Input").

Alkalinity-Balance Equation

The form of the mole-balance equation for alkalinity is identical to the form for other mole-balance equations:

$$\sum_{q}^{Q} c_q \alpha_q (T_{Alk,q} + \delta_{Alk,q}) + \sum_{p}^{P} c_{Alk,p} \alpha_p + \sum_{r}^{R} c_{Alk,r} \alpha_r = 0, \tag{132}$$

where Alk refers to alkalinity. The difference between alkalinity and other mole-balance equations is contained in the meaning of $c_{Alk,r}$ and $c_{Alk,p}$. What is the contribution to the alkalinity of an aqueous solution due to aqueous redox reactions or the dissolution or precipitation of phases? The alkalinity contribution of a reaction is defined by the sum of the alkalinities of the aqueous species in a redox or phase-dissolution reaction. PHREEQC defines $c_{Alk,r}$ and $c_{Alk,p}$ as follows:

$$c_{Alk,r} = \sum_{i}^{N_{aq}} b_{Alk,i} c_{i,r}, \tag{133}$$

and

$$c_{Alk,p} = \sum_{i}^{N_{aq}} b_{Alk,i} c_{i,p}, \tag{134}$$

where $b_{Alk,\,i}$ is the number of equivalents of alkalinity per mole of species i, $c_{i,\,r}$ is the stoichiometric coefficient of the species i in the aqueous redox reaction r, and $c_{i,\,p}$ is the stoichiometric coefficient of the species i in the dissolution reaction for phase p.

Electron-Balance Equation

The mole-balance equation for electrons assumes that no free electrons are present in any of the aqueous solutions. Electrons may enter or leave the system through the aqueous redox reactions or through the phase dissolution reactions. However, the electron-balance equation requires that any electrons entering the system through one reaction be removed from the system by another reaction:

$$\sum_{r}^{R} c_{e^-,\,r}\alpha_r + \sum_{p}^{P} c_{e^-,\,p}\alpha_p = 0, \tag{135}$$

where $c_{e^-,\,r}$ is the number of electrons released or consumed in aqueous redox reaction r, and $c_{e^-,\,p}$ is the number of electrons released or consumed in the dissolution reaction for phase p.

Water-Balance Equation

The mole-balance equation for water is

$$\sum_{q}^{Q} \frac{W_{aq,\,q}}{GFW_{H_2O}} c_q\alpha_q + \sum_{r}^{R} c_{H_2O,\,r}\alpha_r + \sum_{p}^{P} c_{H_2O,\,p}\alpha_p = 0, \tag{136}$$

where GFW_{H_2O} is the gram formula weight for water (approximately 0.018 kg/mol), $W_{aq,\,q}$ is the mass of water in aqueous solution q, $c_{H_2O,\,r}$ is the stoichiometric coefficient of water in aqueous redox reaction r, and $c_{H_2O,\,p}$ is the stoichiometric coefficient of water in the dissolution reaction for phase p.

Charge-Balance Equation

The charge-balance equations for the aqueous solutions constrain the unknown δ's to be such that, when the δ's are added to the original data, charge balance is produced in each aqueous solution. The charge-balance equation for an aqueous solution is

$$\sum_{m}^{M} \tilde{z}_m \delta_{m,\,q} = -T_{z,\,q}, \tag{137}$$

where $T_{z,\,q}$ is the charge imbalance in aqueous solution q calculated by a speciation calculation and $\tilde{z}_m$ is defined to be the charge on the master species plus the alkalinity assigned to the master species, $\tilde{z}_m = z_m + b_{Alk,\,m}$. For alkalinity, $\tilde{z}_{Alk}$ is defined to be -1.0. The summation ranges over all elements or element valence states and includes a term for alkalinity, just as charge balance is commonly calculated by summing over cationic and anionic elements plus a contribution from alkalinity. In the definition of $\tilde{z}_m$, the alkalinity of the master species is added to the charge for that master species to remove the equivalents for the element or element redox state that are already accounted for in the alkalinity. For example, the contribution of carbonate species in equation 137 is zero with this definition of $\tilde{z}_m$ ($z_{CO_3^{2-}} = -2$, $b_{Alk,\,CO_3^{2-}} = 2$, $\tilde{z} = 0$); all of the charge contribution of carbonate species is included in the alkalinity term of the summation.

Isotope-Balance Equations

Geochemical mole-balance models must account for the isotopic composition as well as the chemical composition of the final aqueous solution. In general, isotopic evolution requires solving a differential equation that accounts for fractionation processes for precipitating solids and exsolving gases. In the development presented here, only the simpler case of isotopic mole balance, without fractionation, is considered. This approach is correct if aqueous mixing occurs and (or) all isotope-bearing phases dissolve, but is approximate when isotope-bearing phases precipitate or exsolve. The approach does not calculate isotopic compositions of individual redox states within the aqueous phase, only net changes in isotopic composition of the aqueous phase are considered.

Mole balance for an isotope can be written as

$$0 = \sum_{q}^{Q}\left(c_q \alpha_q \sum_{m}^{M_e}\left(R^i_{m,q} + \delta_{R^i_{m,q}} \right)(T_m + \delta_{m,q}) \right) + \sum_{p}^{P} c_{e,p}\left(R^i_{e,p} + \delta_{R^i_{e,p}} \right)\alpha_p , \tag{138}$$

where M_e is the number of valence states of element e, $R^i_{m,q}$ is the isotopic ratio [which may be delta notation (for example $\delta^{13}C$ or $\delta^{34}S$), ^{14}C activity in percent modern carbon, or any units that allow linear mixing] for isotope i for valence state m in aqueous solution q, $\delta_{R^i_{m,q}}$ is an uncertainty term for the isotopic ratio for a valence state in the aqueous solution, $R^i_{e,p}$ is the isotopic ratio of element e in phase p, and $\delta_{R^i_{e,p}}$ is an uncertainty term for the isotopic ratio of the element in the phase.

Expanding equation 138 and neglecting the products of δ's gives the following approximation:

$$0 \approx \sum_{q}^{Q}\sum_{m}^{M_e}\left(c_q R^i_{m,q} T_m \alpha_q + c_q R^i_{m,q} \alpha_q \delta_{m,q} + c_q T_m \alpha_q \delta_{R^i_{m,q}} \right) + \sum_{p}^{P}\left(c_{e,p} R^i_{e,p} \alpha_p + c_{e,p} \alpha_p \delta_{R^i_{e,p}} \right). \tag{139}$$

Commonly, $\delta_{m,q}$ will be small relative to the concentration of the valence state or $\delta_{R^i_{m,q}}$ for the isotopic ratio will be small relative to the isotopic ratio itself. In either case, the products of δ's that are neglected will be small relative to the other terms and equation 139 will be a good approximation. The approximation in equation 139 will be poor only if the concentration of the valence state and the isotopic ratio have large calculated δ's. In this case, the overall effect is that the true values of the uncertainty terms will be larger than specified uncertainty limits. The neglected terms can be made smaller by decreasing the uncertainty limits on either the valence-state concentrations or the isotopic ratios for each aqueous solution.

Relation Among pH, Alkalinity, and Total Dissolved Inorganic Carbon Uncertainty Terms

One additional equation is added for each aqueous solution to relate the uncertainty terms in pH, alkalinity, and total dissolved inorganic carbon. Unlike all other mole-balance quantities, which are assumed to vary independently, alkalinity, pH, and inorganic carbon are not independent. The following equation is used to relate the uncertainty terms for each of these quantities:

$$\delta_{Alk,q} = \frac{\partial Alk_q}{\partial C_q}\delta_{C_q} + \frac{\partial Alk_q}{\partial pH_q}\delta_{pH,q}, \tag{140}$$

where Alk_q is the alkalinity of solution q, and C_q is the total inorganic carbon of solution q. The partial derivatives are evaluated numerically for each aqueous solution.

Inequality Constraints

This formulation of the inverse problem makes sense only if the values of the δ's are small, meaning that the revised aqueous solution compositions (original plus δ's) do not deviate unreasonably from the original data. A set of inequalities places limits on the magnitudes of the δ's. The absolute value of each δ is constrained to be less than or equal to a specified uncertainty limit, $u_{m,q}$:

$$|\delta_{m,q}| \le u_{m,q}. \tag{141}$$

Inequality constraints (equation 141) are also included for carbon(+4), alkalinity, and pH for each aqueous solution. In addition, the mixing fractions for the initial aqueous solutions ($q < Q$) are constrained to be nonnegative,

$$\alpha_q \ge 0, \tag{142}$$

and the final aqueous-solution mixing fraction is fixed to -1.0 ($\alpha_Q = -1.0$). If phases are known only to dissolve, or only to precipitate, the mole transfer of the phases may be constrained to be nonnegative or nonpositive:

$$\alpha_p \ge 0, \tag{143}$$

or

$$\alpha_p \le 0. \tag{144}$$

Change of Variables

The system of equations for inverse modeling, formulated in the previous section, is nonlinear because it includes the product of unknowns of the form $\alpha_q(T_{m,q} + \delta_{m,q})$, where α and δ are unknowns. However, the equations can be linearized with the substitution

$$\varepsilon_{m,q} = \alpha_q \delta_{m,q}. \tag{145}$$

The mole-balance equations now become

$$\sum_q^Q c_q T_{m,q} \alpha_q + \sum_q^Q c_q \varepsilon_{m,q} + \sum_p^P c_{m,p} \alpha_p + \sum_r^R c_{m,r} \alpha_r = 0. \tag{146}$$

The alkalinity balance equation can be written as

$$\sum_q^Q c_q T_{Alk,q} \alpha_q + \sum_q^Q c_q \varepsilon_{Alk,q} + \sum_p^P c_{Alk,p} \alpha_p + \sum_r^R c_{Alk,r} \alpha_r = 0. \tag{147}$$

The electron-balance equation is unchanged. The charge-balance equation can be rewritten into

$$\sum_m^M \tilde{z}_m \varepsilon_{m,q} + \alpha_q T_{z,q} = 0. \tag{148}$$

The water-balance equation is unchanged. The isotope-balance equation 139 is

$$0 \approx \sum_q^Q \sum_m^{M_e} \left(c_q R^i_{m,q} T_m \alpha_q + c_q R^i_{m,q} \varepsilon_{m,q} + c_q T_m \varepsilon_{R^i_{m,q}} \right) + \sum_p^P \left(c_{e,p} R^i_{e,p} \alpha_p + c_{e,p} \varepsilon_{R^i_{e,p}} \right) \tag{149}$$

The relation among carbon(+4), pH, and alkalinity is

$$\varepsilon_{Alk,q} = \frac{\partial Alk_q}{\partial C_q}\varepsilon_{C,q} + \frac{\partial Alk_q}{\partial pH_q}\varepsilon_{pH,q}; \tag{150}$$

and lastly, the inequality constraints become

$$|\varepsilon_{m,q}| \le \alpha_q u_{m,q}. \tag{151}$$

All of these equality and inequality equations are linear in the unknowns α and ε, and once the values of all of the α and ε are known, the values of the uncertainty terms δ can be determined.

This formulation of the inverse-modeling problem produces a series of linear equality and inequality constraints, which are solved with the algorithm developed by Barrodale and Roberts (1980). Their algorithm performs an L1 optimization (minimize sum of absolute values) on a set of linear equations subject to equality and inequality constraints. The problem can be posed with the following matrix equations:

$$\begin{aligned} AX &= B \\ CX &= D \\ EX &\le F \ . \end{aligned} \tag{152}$$

The first matrix equation is minimized in the sense that $\sum_i \left| b_i - \sum_j a_{i,j} x_j \right|$ is a minimum, where i is the index of rows and j is the index for columns, subject to the equality constraints of the second matrix equation and the inequality constraints of the third matrix equation. The method will find a solution that minimizes the objective functions ($AX = B$) or it will determine that no feasible model for the problem exists.

Initially, $AX = B$ is set to minimize $\sum_q \sum_m \frac{S|\varepsilon_{m,q}|}{u_{m,q}}$, where $S = 0.001$ is a scaling factor that limits the size of the coefficients in the A matrix; A is a diagonal matrix with elements $\frac{S}{u}$, and $B = 0$. The equality constraints ($CX = D$) include all mole-balance, alkalinity-balance, charge-balance, electron-balance, and water-balance equations and all inorganic carbon-alkalinity-pH relations. The inequality constraints ($EX \le F$) include two inequalities for each of the ε's, one for positive and one for negative (to account for the absolute values used in the formulation), an inequality relation for each mixing fraction for the aqueous solutions, which forces each mixing fraction to be nonnegative, and an inequality relation for each phase that is specified to dissolve only or precipitate only. Application of the optimization technique will determine whether any inverse models exist that are consistent with the constraints.

Thus, one set of mixing fractions and phase mole transfers (plus associated ε's) that satisfy the constraints may be found. Ignoring the values of the ε's and redox mole transfers (α_r), let the set of nonzero α_q and α_p (mixing fractions and phase mole transfers) uniquely identify an inverse model. The magnitude of the α's is not important in the identity of an inverse model, only the fact that the α's are nonzero in a certain set is considered. (At this point, little significance should be placed on the exact mole transfers that are found, only that it is possible to account for the observations using the aqueous solutions and phases of the inverse model.) But could other sets of aqueous solutions and phases also produce feasible inverse models? An additional algorithm is used to find all of the unique inverse models.

Assuming P phases and Q aqueous solutions, we proceed as follows: If no model is found when all Q aqueous solutions and P phases are included in the equations, we are done and no feasible models exist. If a model is found, then each of the phases in the model is sequentially removed and the remaining set of phases and aqueous solutions

is tested to see if other feasible models exist. If no model is found when a particular phase is removed, the phase is retained in the model; otherwise, the phase is discarded. After each phase has been tested and possibly discarded, the phases that remain constitute a "minimal" model, that is, to obtain a feasible model none of the phases can be removed. Three lists are kept during this process: each feasible model is kept in one list, each infeasible model is kept in another list, and each minimal model is kept in a third list.

Next, each combination of *P-1* phases is tested for feasible models in the following way. If a trial model with Q aqueous solutions and *P-1* phases is a subset of a model in the infeasible- or minimal-model list, the trial model is skipped because it must be either infeasible or a previously identified minimal model. If only minimal models are to be found (**-minimal** in **INVERSE_MODELING** data block), the trial model is skipped if it is a superset of a model in the minimal-model list. Otherwise, the inverse problem is formulated for the trial model and solved using the set of aqueous solutions and the *P-1* phases in the same way as described above, maintaining the three lists during the process. Once all sets of *P-1* phases have been tested, the process continues with sets of *P-2* phases, and so on until the set containing no phases is tested or until, for the given number of phases, every trial model is either a subset of a model in the infeasible- or minimal-model list.

At this point, the entire process is repeated using each possible combination of one or more of the Q aqueous solutions. Although the process at first appears extremely computer intensive, most sets of phases are rapidly eliminated by subset and superset comparisons with models in the three lists. The number of models that are formulated and solved by the optimization methods are relatively few. Also the process has the useful feature that if no feasible models exist, this is determined immediately when the optimization procedure is invoked the first time. For PHREEQC, during all of the testing, whenever a feasible model is found, it is printed to the output device or optionally, only the minimal models are printed to the output device.

An alternative formulation of the objective functions can be used to determine the range of mole transfer for each aqueous solution and each phase that is consistent with the specified uncertainty limits. For the "range" calculation (**-range** in **INVERSE_MODELING** data block), the equations for a given model are solved twice for each aqueous solution and phase in the model, once to determine the maximum value of the mixing fraction or mole transfer and once to determine the minimum value of the mixing fraction or mole transfer. In these calculations, the $\frac{\varepsilon}{u}$'s are not minimized, but instead, the single objective function for maximization is

$$\alpha = M, \tag{153}$$

and in the minimization case,

$$\alpha = -M, \tag{154}$$

where α refers to either α_q or α_p, and M is a large number. By default, the value of M is 1000. The optimization method will try to minimize the difference between α and 1000 for maximization and between α and -1000 for minimization. It is possible that the mixing fraction for a solution (α_q) could exceed 1000 in an evaporation problem. In this case, the method would fail to find the true maximum for α_q, and instead find a value closest to 1000. This error can be remedied by choosing a larger value for M. The value of M may be changed with the **-range** identifier in the **INVERSE_MODELING** data block.

For data input to PHREEQC, identifiers in the **INVERSE_MODELING** data block are used for the selection of aqueous solutions (**-solutions**), uncertainty limits (**-uncertainties** and **-balances**), reactants (**-phases**), mole-balance equations (**-balances**), range calculations (**-range**) and minimal models (**-minimal**). Aqueous

solution compositions are defined with the **SOLUTION**, **SOLUTION_SPREAD**, or **SAVE** data block and reactants must be defined with **PHASES** or **EXCHANGE_SPECIES** data blocks (see "Description of Data Input").

ORGANIZATION OF THE COMPUTER CODE

The computer code for PHREEQC is divided into 22 files of C code, roughly corresponding to processing tasks. Definitions of global variables and global structures are defined in the header file *global.h*, which is included in all of the source code files (those ending in ".c") except *cl1.c*. Definitions of variables and structures for the Basic interpreter are contained in *p2c.h*. Definitions for memory allocation routines are contained in *phqalloc.h*, which is included in all source code files except *phqalloc.c*.

The main program is in the file *main.c*. It contains the logic for the sequence of calculations, which occur in the following order: (1) At the beginning of the run, the database file is read. The database file usually defines the elements and mass-action expressions for all of the aqueous species and phases. Definition of species for exchangers and surfaces and rate expressions may also be included in this file. (2) A simulation is read from the input data file. (3) Any initial solution calculations are performed. (4) Any initial exchange-composition calculations are performed. (5) Any initial surface-composition calculations are performed. (6) Any initial gas-phase-composition calculations are performed. (7) Any batch reactions (mixing, irreversible reaction, mineral equilibration, and others) are performed. (8) Any inverse modeling calculations are performed. (9) Any advective-reactive transport calculations (**ADVECTION** keyword) are performed. And, (10) any advective-dispersive-reactive transport calculations (**TRANSPORT** keyword) are performed. The sequence from (2) through (10) is repeated until the end of the input file is encountered. The subroutines that perform tasks (3) through (7) are found in the file *mainsubs.c*. The subroutines to perform inverse modeling (8) are found in *inverse.c*, to perform advective-reactive transport modeling (9) are found in *advection.c*, and to perform dispersive/diffusive-reactive transport modeling (10) are found in *transport.c*.

The file *read.c* is used to read both the database file and the input file. It contains a main subroutine that calls subroutines that read each data block. All subroutines to read data blocks are found in *read.c*, except the subroutines for **TRANSPORT**, which are found in *readtr.c*, and **SOLUTION_SPREAD**, which are found in *spread.c*. In the process of reading, memory is allocated to store the information for each keyword. Thus, the memory used by the program grows depending on the number and type of keywords that are included in the input file. The only restriction on the size of the program is the available memory and swap space that is physically present in the computer that is used. Chemical equations that are read from the input files are interpreted and checked for charge and mole balance by the subroutines in *parse.c*.

Subroutines in the file *tidy.c* check and organize the data read through *read.c*. These subroutines sort the lists of species, solutions, phases, pure-phase assemblages, and others, so that the order of these entities is known. They ensure that any elements used in mass-action equations are defined to the program and that all necessary primary and secondary master species exist. In addition, they rewrite all mass-action equations so that they contain only primary and secondary master species. Other consistency checks and data organization for exchangers, gas phases, pure-phase assemblages, surfaces, solid solutions, and inverse modeling are performed by the subroutines in this file. Also, the selected-output file is opened and headings are written.

Subroutines in the file *prep.c* set up the equations for a calculation. The equations and unknowns that are needed for the calculation are determined and work space to solve a matrix with this number of equations and

unknowns is allocated. All mass-action expressions are rewritten according to the master-species and redox information for the calculation. Several lists of pointers are prepared that allow the residuals of equations, the Newton-Raphson array, and the change in moles of elements due to mineral mole transfers to be calculated quickly. These lists are C structures that in general contain a pointer to a "source" datum in memory, a coefficient, and another pointer to a "target" memory location. The source datum is retrieved, multiplied by the coefficient, and added to the target memory location. As an example, consider the species $CaSO_4$, which should appear in the mole-balance equations for calcium, sulfur, and oxygen. One of the lists is used to calculate the residuals of the mole-balance equations. There would be three entries in this list for the species $CaSO_4$. In all three entries, the source datum would be a pointer to the moles of $CaSO_4$. The target memory locations would be the variable locations where the residuals for calcium, sulfur, and oxygen mole balances are stored, and the coefficients would be 1.0, 1.0, and 4.0, respectively. Once the entire list is generated, at each iteration, it is only necessary to perform the multiplications and additions as described by the list to calculate the residuals of the mole-balance equations, no extraneous calculations (multiplication by zero, for example), additional loops, or conditional statements are necessary. The actual implementation uses several lists for each task to skip multiplication if the coefficient is 1.0, and to include constants that are not iteration dependent (that is, do not require the pointer to a source datum). An additional list is generated that is used for printing. For each aqueous species, this list includes an entry for each master species in the mass-action equation. This list is sorted by master species and concentration after the equilibrium calculation is completed and provides all the information for aqueous, exchange, and surface species for printing results to the output file.

In batch-reaction and transport calculations, if the set of elements, exchanger components, gas-phase components, pure phases, solid solutions, and surface components does not change from one calculation to the next, then the lists prepared in *prep.c* do not need to be regenerated. In this case, the lists used during the previous calculation are used for the current calculation. Thus, most of the time spent in the subroutines of the file *prep.c* can be saved.

The subroutines in *model.c* solve the equations that have been set up in *prep.c*. Initial estimates for the master unknowns are calculated and the residuals for mole-balance equations are reduced below tolerances to provide suitable estimates for the Newton-Raphson technique. Once suitable estimates of the master unknowns have been found, the following iterative process occurs. (1) The residuals of the equations are tested for convergence; if convergence is found, the calculation is complete. Otherwise, (2) the Newton-Raphson matrix is formulated and solved (by subroutine cl1, in file *cl1.c*), (3) the master unknowns are updated, (4) activity coefficients are calculated, (5) the distribution of species is calculated, (6) if a master species of a redox element becomes small, basis switching may be performed. In the basis-switching process, new mass-action equations are written and the lists for calculating residuals and the Newton-Raphson matrix are remade, and (7) the residuals of the equations are calculated. Steps (1) through (7) are repeated until a solution to the equations is found or a prescribed number of iterations is exceeded. If the numerical method fails to find a solution, six additional sets of convergence parameters are used in trying to obtain convergence (see description of **KNOBS** data block in "Description of Data Input" for details on alternative convergence parameters). If all sets of parameters fail, the program terminates.

Following a calculation, the subroutines in *print.c* write data to the output file and to the selected-output file. Concentration data for species are sorted so that species are printed in descending order by concentration. The blocks of output that are written are selected with the keywords **PRINT**, **SELECTED_OUTPUT**,

USER_PRINT, and **USER_PUNCH**. If no data are to be printed to the output file, the species sort is not needed and is not performed. If the aqueous solution, exchange assemblage, gas phase, pure-phase assemblage, solid-solution assemblage, or surface assemblage is saved following a calculation, the routines that perform these tasks are found in *mainsubs.c*.

The subroutines in *step.c* are used to accumulate the moles of each element before batch-reaction and transport calculations. Total concentrations of elements are calculated from the amounts in solution, on exchangers, in the gas phase, and on surfaces. A check is made to ensure that all of the elements in the pure phases and solid solutions are included in the list of elements with positive concentrations. If an element is in a pure phase or solid solution, but not in the aqueous solution, a small amount of the pure phase is added to the aqueous solution. If the moles of the pure phase or solid solution are zero and one of its constituent elements is not present, that pure phase or solid solution is ignored in the calculations.

If kinetic calculations are defined for batch-reaction or transport calculations, the reactions are integrated by routines in *kinetics.c* and the extent of the kinetic reactions at each time increment are calculated by the Basic interpreter, which is found in *basic.c* and *p2clib.c*. If explicit diffuse-layer calculations are made, the integration of the Poisson-Boltzmann equation is performed by the subroutines in *integrate.c*.

A few functions that are used throughout the code are found in *utilities.c*. Finally, many of the manipulations of structures, including allocating space, initializing, copying, and freeing space are performed by subroutines in the file *structures.c*. The subroutine "clean_up" (in *structures.c*) frees all allocated memory at the termination of the program. The file *phqalloc.c* contains subroutines to allocate, reallocate, and free memory. These subroutines have capabilities for debugging memory leaks, that is memory that is allocated, but not released when it is no longer needed.

For efficiency, a hash table of character strings is kept by the program. Each character string, including element names, species names, phase names, and others, is stored only once. All references to the same string then point to the same memory location. Thus, for example, a comparison of element names need only check to see if the memory address is the same, avoiding the necessity of comparing the strings character by character. Finding the memory location of a specified string is performed by a hash table lookup. Hash tables are also used to speed up lookups for species, elements, and phases.

DESCRIPTION OF DATA INPUT

The input for PHREEQC is arranged by keyword data blocks. Each data block begins with a line that contains the keyword (and possibly additional data) followed by additional lines containing data related to the keyword. The keywords that define the input data for running the program are listed in table 2. Keywords and their associated data are read from a database file at the beginning of a run to define the elements, exchange reactions, surface complexation reactions, mineral phases, gas components, and rate expressions. Any data items read from the database file can be redefined by keyword data blocks in the input file. After the database file is read, data are read from the input file until the first **END** keyword is encountered, after which the specified calculations are performed. The process of reading data from the input file until an **END** is encountered followed by performing calculations is repeated until the last **END** keyword or the end of the input file is encountered. The set of calculations, defined by keyword data blocks terminated by an **END**, is termed a "simulation". A "run" is a series of one or more simulations that are contained in the same input data file and calculated during the same invocation of the program PHREEQC.

Table 2.--List of keyword data blocks and their function

Keyword data block	Function
ADVECTION	Specify parameters for advective-reactive transport, no dispersion
END	Demarcate end of a simulation
EQUILIBRIUM_PHASES	Phase assemblage to react with an aqueous solution
EXCHANGE	Define exchange assemblage composition
EXCHANGE_MASTER_SPECIES	Identify exchange sites and corresponding exchange master species
EXCHANGE_SPECIES	Define association half-reaction and thermodynamic data for exchange species
GAS_PHASE	Define a gas-phase composition
INCREMENTAL_REACTIONS	Define whether reaction increments are defined incrementally or cumulatively
INVERSE_MODELING	Specify solutions, reactants, and parameters for mole-balance modeling
KINETICS	Specify kinetic reactions and define parameters
KNOBS	Define parameters for numerical method and printing debugging information
MIX	Define mixing fractions of aqueous solutions
PHASES	Define dissociation reactions and thermodynamic data for minerals and gases
PRINT	Select data blocks to be printed to the output file
RATES	Define rate equations with Basic language statements
REACTION	Add specified irreversible reactions
REACTION_TEMPERATURE	Specify temperature for batch reactions
SAVE	Save results of batch reactions for use in subsequent simulations
SELECTED_OUTPUT	Print specified quantities to a user-defined file
SOLID_SOLUTIONS	Define the composition of a solid-solution assemblage
SOLUTION	Define the composition of an aqueous solution
SOLUTION_MASTER_SPECIES	Identify elements and corresponding aqueous master species
SOLUTION_SPECIES	Define association reaction and thermodynamic data for aqueous species
SOLUTION_SPREAD	Define one or more aqueous solution compositions using a tab-delimited format. Alternative input format for **SOLUTION**.
SURFACE	Define the composition of an assemblage of surfaces
SURFACE_MASTER_SPECIES	Identify surface sites and corresponding surface master species
SURFACE_SPECIES	Define association reaction and thermodynamic data for surface species
TITLE	Specify a text string to be printed in the output file
TRANSPORT	Specify parameters for an advective-dispersive-reactive transport, optionally with dual porosity
USE	Select aqueous solution or other reactants that define batch reactions
USER_PRINT	Print user-defined quantities to the output file
USER_PUNCH	Print user-defined quantities to the selected-output file

Each simulation may contain one or more of seven types of speciation, batch-reaction, and transport calculations: (1) initial solution speciation, (2) determination of the composition of an exchange assemblage in

equilibrium with a fixed solution composition, (3) determination of the composition of a surface assemblage in equilibrium with a fixed solution composition, (4) determination of the composition of a fixed-volume gas phase in equilibrium with a fixed solution composition, (5) calculation of chemical composition as a result of batch-reactions, which include mixing; kinetically controlled reactions; net addition or removal of elements from solution, termed "net stoichiometric reaction"; variation in temperature; equilibration with assemblages of pure phases, exchangers, surfaces, and (or) solid solutions; and equilibration with a gas phase at a fixed total pressure or fixed volume, (6) advective-reactive transport, or (7) advective-dispersive-reactive transport through a series of cells in combination with any of the available chemical processes. The combination of capabilities allows the modeling of very complex geochemical reactions and transport processes during one or more simulations.

In addition to speciation, batch-reaction, and transport calculations, the code may be used for inverse modeling, by which net chemical reactions are deduced that account for differences between an initial water composition or a mixture of initial water compositions and a final water composition.

Conventions for Data Input

PHREEQC was designed to eliminate some of the input errors due to complicated data formatting. Data for the program are free format; spaces or tabs may be used to delimit input fields (except **SOLUTION_SPREAD**, which is delimited only with tabs); blank lines are ignored. Keyword data blocks within a simulation may be entered in any order. However, data elements entered on a single line are order specific. As much as possible, the program is case insensitive. However, chemical formulas are case sensitive.

The following conventions are used for data input to PHREEQC:

Keywords--Input data blocks are identified with an initial keyword. This word must be spelled exactly, although case is not important. Several of the keywords have synonyms. For example **PURE_PHASES** is a synonym for **EQUILIBRIUM_PHASES**.

Identifiers--Identifiers are options that may be used within a keyword data block. Identifiers may have two forms: (1) they may be spelled completely and exactly (case insensitive) or (2) they may be preceded by a hyphen and then only enough characters to uniquely define the identifier are needed. The form with the hyphen is always acceptable and is recommended. Usually, the form without the hyphen is acceptable, but in some cases the hyphen is needed to indicate the word is an identifier rather than an identically spelled keyword; these cases are noted in the definition of the identifiers in the following sections. In this report, the form with the hyphen is used except for identifiers of the **SOLUTION** keyword and the identifiers **log_k** and **delta_h**. The hyphen in the identifier never implies that the negative of a quantity is entered.

Chemical equations--For aqueous, exchange, and surface species, chemical reactions must be *association* reactions, with the defined species occurring in the first position past the equal sign. For phases, chemical reactions must be *dissolution* reactions with the formula for the defined phase occurring in the first position on the left-hand side of the equation. Additional terms on the left-hand side are allowed. All chemical equations must contain an equal sign, "=". In addition, left- and right-hand sides of all chemical equations must balance in numbers of atoms of each element and total charge. All equations are checked for these criteria at runtime, unless they are specifically excepted. Nested parentheses in chemical formulas are acceptable. Spaces and tabs within chemical equations are ignored. Waters of hydration and other chemical formulas that are normally represented by a " · "--as in the formula

for gypsum, $CaSO_4 \cdot 2H_2O$--are designated with a colon (":") in PHREEQC ($CaSO_4$:$2H_2O$), but only one colon per formula is permitted.

Element names--An element formula, wherever it is used, must begin with a capital letter and may be followed by one or more lowercase letters or underscores, "_". In general, element names are simply the chemical symbols for elements, which have a capital letter and zero or one lower case letter. It is sometimes useful to define other entities as elements, which allows mole balance and mass-action equations to be applied. Thus, "Fulvate" is an acceptable element name, and it would be possible to define metal binding constants in terms of metal-Fulvate complexes.

Charge on a chemical species--The charge on a species may be defined by the proper number of pluses or minuses following the chemical formula or by a single plus or minus followed by a integer number designating the charge. Either of the following is acceptable, Al+3 or Al+++. However, Al3+ would be interpreted as a molecule with three aluminum atoms and a charge of plus one.

Valence states--Redox elements that exist in more than one valence state in solution are identified for definition of solution composition by the element name followed by the formal valence in parentheses. Thus, sulfur that exists as sulfate is defined as S(6) and total sulfide (H_2S, HS^-, and others) is identified by S(-2).

log *K* and temperature dependence--The identifier **log_k** is used to define the log *K* at 25°C for a reaction. The temperature dependence for log *K* may be defined by the van't Hoff expression or by an analytical expression. The identifier **delta_h** is used to give the standard enthalpy of reaction at 25°C for a chemical reaction, which is used in the van't Hoff equation. By default the units of the standard enthalpy are kilojoules per mole (kJ/mol). Optionally, for each reaction the units may be defined to be kilocalories per mole (kcal/mol). An analytical expression for the temperature dependence of log *K* for a reaction may be defined with the **-analytical_expression** identifier. Up to five numbers may be given, which are the coefficients for the equation:

$$\log_{10} K = A_1 + A_2 T + \frac{A_3}{T} + A_4 \log_{10} T + \frac{A_5}{T^2},$$ where T is in Kelvin. A log *K* must always be defined either with **log_k** or **-analytical_expression**; the enthalpy is optional. If both are present, an analytical expression is used in preference to the enthalpy value for calculation of temperature dependence of a log *K*.

Comments--The "#" character delimits the beginning of a comment in the input file. All characters in the line that follow this character are ignored. If the entire line is a comment, the line is not echoed to the output file. If the comment follows input data on a line, the entire line, including the comment, is echoed to the output file. The "#" is useful for adding comments explaining the source of various data or describing the problem set up. In addition, it is useful for temporarily removing lines from an input file.

Logical line separator--A semicolon (";") is interpreted as a logical end of line character. This allows multiple logical lines to be entered on the same physical line. For example, solution data could be entered as:

```
pH 7.0; pe 4.0; temp 25.0
```

on one line. The semicolon should not be used in character fields, such as the title or other comment or description fields.

Logical line continuation--A backslash ("\") at the end of a line may be used to merge two physical lines into one logical line. For example, a long chemical equation could be entered as:

```
"Ca0.165Al2.33Si3.67O10(OH)2 + 12 H2O = \"
"0.165Ca+2 + 2.33 Al(OH)4- + 3.67 H4SiO4 + 2 H+"
```

on two lines. The program would interpret this sequence as a balanced equation entered on a single logical line. Note the backslash must be the last character in the line; if white space or other characters follow the backslash, the next line is not considered to be a continuation line.

Repeat count--An asterisk ("*") can be used with data for the identifiers **-length** and **-dispersivity** in the **TRANSPORT** data block to indicate a repeat count for the following data item. The format is an integer followed directly by the asterisk, which is followed directly by a numeric value. For example "4*1.0" is the same as entering four values of 1.0 ("1.0 1.0 1.0 1.0").

Range of integers--A hyphen ("-") can be used to indicate a range of integers for the keywords with an identification number. It is also possible to define a range of cell numbers for the identifiers **-print_cells** and **-punch_cells** in the **ADVECTION** and **TRANSPORT** data blocks. A range of integers is given in the form *m-n*, where *m* and *n* are positive integers, *m* is less than *n*, and the two numbers are separated by a hyphen without intervening spaces.

Table 3.--Summary of special characters for input data files

Special character	Use
-	When preceding a character string, a hyphen indicates an identifier for a keyword (or subkeyword).
-	Indicates a range of cell numbers for keyword data blocks (for example, **SOLUTION** 2-5) or for identifiers **-print_cells** and **-punch_cells** in the **ADVECTION** and **TRANSPORT** data blocks.
:	In a chemical equation, ":" replaces "·" in a formula like $CaSO_4 \cdot 2H_2O$.
( )	The redox state of an element is defined by a valence enclosed by parentheses following an element name.
#	Comment character, all characters following # are ignored.
;	Logical line separator.
\	Line continuation if "\" is the last non-white-space character of a line.
*	Can be used to indicate a repeat count for **-length** and **-dispersivity** values in the **TRANSPORT** data block.

Reducing Chemical Equations to a Standard Form

The numerical algorithm of PHREEQC requires that chemical equations be written in a particular form. Internally, every equation must be written in terms of a minimum set of chemical species, essentially, one species for each element or valence state of an element. For the program PHREEQE, these species were called "master species" and the reactions for all aqueous complexes had to be written using only these species. PHREEQC also needs reactions in terms of master species; however, the program contains the logic to rewrite the input equations into this form. Thus, it is possible to enter an association reaction and log K for an aqueous species in terms of any aqueous species in the database (not just master species) and PHREEQC will rewrite the equation to the proper internal form. PHREEQC will also rewrite reactions for phases, exchange complexes, and surface complexes. Reactions are still required to be dissolution reactions for phases and association reactions for aqueous, exchange, or surface complexes.

There is one major restriction on the rewriting capabilities for aqueous species. PHREEQC allows mole balances on individual valence states or combinations of valence states of an element for initial solution calculations. It is

necessary for PHREEQC to be able to determine the valence state of an element in a species from the chemical equation that defines the species. To do this, the program requires that at most one aqueous species of an element valence state is defined by an electron half-reaction that relates it to another valence state. The aqueous species defined by this half-reaction is termed a "secondary master species"; there must be a one-to-one correspondence between valence states and secondary master species. In addition, there must be one "primary master species" for each element, such that reactions for all aqueous species for an element can be written in terms of the primary master species. The equation for the primary master species is simply an identity reaction. If the element is a redox element, the primary master species must also be a secondary master species. For example, to be able to calculate mole balances on total iron, total ferric iron, or total ferrous iron, a primary master species must be defined for Fe and secondary master species must be defined for Fe(+3) (ferric iron) and Fe(+2) (ferrous iron). In the default databases, the primary master species for Fe is Fe^{+2}, the secondary master species for Fe(+2) is Fe^{+2}, and the secondary master species for Fe(+3) is Fe^{+3}. The correspondence between master species and elements and element valence states is defined by the **SOLUTION_MASTER_SPECIES** data block. The chemical equations for the master species and all other aqueous species are defined by the **SOLUTION_SPECIES** data block.

Conventions for Documentation

The descriptions of keywords and their associated input are now described in alphabetical order. Several formatting conventions are used to help the user interpret the input requirements. Keywords are always capitalized and bold. Words in bold must be included literally when creating input data sets (although upper and lower case are interchangeable and optional spellings may be permitted). "Identifiers" are additional keywords that apply only within a given keyword data block; they can be described as sub-keywords. "**Temperature**" is an identifier for **SOLUTION** input. Each identifier may have two forms: (1) the identifier word spelled exactly (for example, "**temperature**"), or (2) a hyphen followed by a sufficient number of characters to define the identifier uniquely (for example, **-t** for temperature). The form with the hyphen is recommended. Words in *italics* are input values that are variable and depend on user selection of appropriate values. Items in brackets ([ ]) are optional input fields. Mutually exclusive input fields are enclosed in parentheses and separated by the word "or". In general, the optional fields in a line must be entered in the specified order and it is sometimes possible to omit intervening fields. For clarity, commas are used to delimit input fields in the explanations of data input; however, commas are not allowed in the input data file; only white space (spaces and tabs) may be used to delimit fields in input data sets. Where applicable, default values for input fields are stated.

Getting Started

When the program PHREEQC is invoked, two files are used to define the thermodynamic model and the types of calculations that will be done, the database file and the input file. The database file is read once (to the end of the file or until an **END** keyword is encountered) at the beginning of the program. The input file is then read and processed simulation by simulation (as defined by **END** keywords) until the end of the file. The formats for the keyword data blocks are the same for either the input file or the database file.

The database file is used to define static data for the thermodynamic model. Although any keyword data block can occur in the database file, normally, it contains the keyword data blocks:
EXCHANGE_MASTER_SPECIES, EXCHANGE_SPECIES, PHASES, RATES,

SOLUTION_MASTER_SPECIES, SOLUTION_SPECIES, SURFACE_MASTER_SPECIES, and **SURFACE_SPECIES**. These keyword data blocks define rate expressions; master species; and the stoichiometric and thermodynamic properties of all of the aqueous phase species, exchange species, surface species, and pure phases. Three database files are provided with the program: (1) a database file derived from PHREEQE (Parkhurst and others, 1980), (2) a database file derived from WATEQ4F (Ball and Nordstrom, 1991), and (3) a database file derived from MINTEQA2 (Allison and others, 1990). These files are described in more detail in Attachment B and the PHREEQE-derived database file (*phreeqc.dat*) is listed. The elements and element valence states that are included in *phreeqc.dat* are listed in table 4 along with the PHREEQC notation and the default formula used to convert mass concentration units to mole concentration units.

The input data file is used (1) to define the types of calculations that are to be done, and (2) if necessary, to modify the data read from the database file. If new elements and aqueous species, exchange species, surface species, or phases need to be included in addition to those defined in the database file, or if the stoichiometry, log K, or activity coefficient information from the database file needs to be modified for a given run, then the keywords mentioned in the previous paragraph can be included in the input file. The data read for these keyword data blocks in the input file will augment or supersede the data read from the database file. In many cases, the thermodynamic model defined in the database will not be modified, and the above keywords will not be used in the input data file.

Speciation Calculations

Speciation modeling uses a chemical analysis of a water to calculate the distribution of aqueous species using an ion-association aqueous model. The most important results of speciation calculations are saturation indices for minerals, which indicate the saturation state of each mineral relative to the water. Normally, speciation modeling requires only a **SOLUTION** (or **SOLUTION_SPREAD**) data block for each water analysis for which saturation indices are to be calculated. Example 1 demonstrates speciation calculations.

Batch-Reaction Calculations

Batch-reaction calculations simulate reactions occurring in a beaker and can involve equilibrium and irreversible reactions. Equilibrium reactions are defined by specifying a solution or mixture of solutions to be put in the beaker along with a pure-phase assemblage, an exchange assemblage, a multicomponent gas phase, a solid-solution assemblage, or a surface assemblage. The solution or mixture is brought to equilibrium with the reactants. Furthermore, irreversible reactions can be specified, including addition or removal of specified reactants, temperature changes, and (or) kinetic reactions--for which the reaction rate depends on solution composition. Conceptually, the irreversible reactions are added and equilibrium is calculated for the system (see examples 2, 3, 4, 5, 6, 7, 8, and 10 in "Examples"). Kinetic reactions are integrated for a specified time step by calculating equilibrium following each of a series of irreversible reactions that depend on the evolving composition of the solution (see examples 6 and 9 in "Examples").

Initial conditions for batch reactions are defined with **SOLUTION, SOLUTION_SPREAD, EQUILIBRIUM_PHASES, EXCHANGE, GAS_PHASE, SOLID_SOLUTIONS,** and **SURFACE** data blocks. Irreversible reactions are defined with the data blocks **MIX**, for mixing of solutions; **REACTION**, for adding or removing fixed amounts of specified reactants; **KINETICS** and **RATES**, for defining kinetic reactions; and **REACTION_TEMPERATURE**, for changing the temperature at which the batch reaction occurs.

Table 4.--Elements and element valence states included in default database *phreeqc.dat*, including PHREEQC notation and default formula for gram formula weight

[For alkalinity, formula for gram equivalent weight is given]

Element or element valence state	PHREEQC notation	Formula used for default gram formula weight
Alkalinity	Alkalinity	$Ca_{0.5}(CO_3)_{0.5}$
Aluminum	Al	Al
Barium	Ba	Ba
Boron	B	B
Bromide	Br	Br
Cadmium	Cd	Cd
Calcium	Ca	Ca
Carbon	C	HCO_3
Carbon(IV)	C(4)	HCO_3
Carbon(-IV), methane	C(-4)	CH_4
Chloride	Cl	Cl
Copper	Cu	Cu
Copper(II)	Cu(2)	Cu
Copper(I)	Cu(1)	Cu
Fluoride	F	F
Hydrogen(0), dissolved hydrogen	H(0)	H
Iron	Fe	Fe
Iron(II)	Fe(2)	Fe
Iron(III)	Fe(3)	Fe
Lead	Pb	Pb
Lithium	Li	Li
Magnesium	Mg	Mg
Manganese	Mn	Mn
Manganese(II)	Mn(2)	Mn
Manganese(III)	Mn(3)	Mn
Nitrogen	N	N
Nitrogen(V), nitrate	N(5)	N
Nitrogen(III), nitrite	N(3)	N
Nitrogen(0), dissolved nitrogen	N(0)	N
Nitrogen(-III), ammonia	N(-3)	N
Oxygen(0), dissolved oxygen	O(0)	O
Phosphorous	P	P
Potassium	K	K
Silica	Si	SiO_2
Sodium	Na	Na
Strontium	Sr	Sr
Sulfur	S	SO_4
Sulfur(VI), sulfate	S(6)	SO_4
Sulfur(-II), sulfide	S(-2)	S
Zinc	Zn	Zn

Different sets of keyword data blocks can be defined within one simulation, each set being identified by the number or range of numbers which follow the keyword. In the subsequent batch reaction, a set may be included either implicitly or explicitly. For an implicit calculation, a solution or mixture (**SOLUTION** or **MIX** keywords) must be defined within the simulation, and the first of each keyword set (defined before the END) will be included in the calculation. That is, the first solution (or mixture) will be used along with the first of each of the data blocks **EQUILIBRIUM_PHASES**, **EXCHANGE**, **GAS_PHASE**, **SOLID_SOLUTIONS**, **SURFACE**, **KINETICS**, **REACTION**, and **REACTION_TEMPERATURE**. For an explicit calculation, "**USE** *keyword number*" defines a set that is to be used regardless of position within the input lines (see examples 3, 6, 7, 8, and 9 in "Examples"). "**USE** *keyword* **none**" eliminates a set that was implicitly defined (see example 8 in "Examples"). If the composition of the solution, pure-phase assemblage, exchange assemblage, gas phase, solid-solution assemblage, or surface assemblage has changed after the batch-reaction calculation, it can be saved with the **SAVE** keyword.

Advective-Transport Calculations

Advective-transport calculations are used to simulate advection and chemical reactions as water moves through a one dimensional column (**ADVECTION** data block). The column is divided into a number of cells, *n*, which is defined by the user. The cells are numbered 1 through *n*, and these cells initially contain solutions with identifying numbers 1 through *n*. A solution composition for each of these integers must have been defined by **SOLUTION** data blocks or the **SAVE** keyword. The cells may also contain other reversible or irreversible reactants. For a given cell number, *i*, if a phase assemblage, exchange assemblage, solid-solution assemblage, surface assemblage, gas phase, mixture, reaction, or reaction-temperature data block with identifying number *i* has been defined, then it is present in cell *i* during the advective-transport calculation. Thus, the initial conditions and the set of reactants can be defined individually for each cell, which provides flexibility to simulate a variety of chemical conditions throughout the column (see examples 11 and 14 in "Examples").

The infilling solution for the column is always solution number 0. Advection is modeled by "shifting" solution 0 to cell 1, the solution in cell 1 to cell 2, and so on. At each shift, kinetic reactions are integrated in each cell, while maintaining equilibrium with any gas phase or solid-phase assemblages that are present in each cell. To facilitate definition of the initial conditions the keywords **EQUILIBRIUM_PHASES**, **EXCHANGE**, **GAS_PHASE**, **KINETICS**, **MIX**, **REACTION**, **REACTION_TEMPERATURE**, **SOLID_SOLUTIONS**, **SOLUTION**, and **SURFACE** allow simultaneous definition of a range of cell numbers. The **SAVE** keyword also permits a range of solution, gas phase, or assemblage numbers to be saved simultaneously.

Advective-Dispersive-Transport Calculations

Analogous to purely advective transport, advective-dispersive or diffusive 1D transport can be modeled with the **TRANSPORT** data block. A single dispersion or diffusion coefficient is used for all chemical species, and, therefore, dispersion and diffusion have the same mathematical formulation for all species. For this document, the terms "dispersion" and "dispersive transport" are used to describe both dispersion and diffusion restricted to a single dispersion/diffusion coefficient. Like for purely advective transport, a column of *n* cells is defined, but also dispersion/diffusion parameters, boundary conditions, direction of flow, cell lengths, and advective time step can be provided. It is also possible to model double porosity by including the relevant information. The infilling solution depends on the direction of flow and may be solution number 0 or *n*+*1*. For each shift (advection step), a number of

dispersion/diffusion mixing steps are performed. For each shift and dispersion step, kinetic reactions are integrated for each cell while maintaining equilibrium with any gas phase or assemblages that are present in the cell (see examples 11, 12, 13, and 15, in "Examples").

Inverse Modeling

Inverse modeling is used to deduce the geochemical reactions that account for the change in chemical composition of water along a flow path. At least two chemical analyses of water at different points along the flow path are needed, as well as a set of phases that are potentially reactive along the flow path. From the analyses and phases, mole-balance models are calculated. A mole-balance model is a set of mole transfers of phases and reactants that accounts for the change in composition along the flow path. Normally, only **SOLUTION** or **SOLUTION_SPREAD** data blocks and an **INVERSE_MODELING** keyword data block are needed for inverse modeling calculations. However, additional reactant phases may need to be defined with **PHASES** or **EXCHANGE_SPECIES** data blocks (see examples 16, 17, and 18, in "Examples").

Units

The concentrations of elements in solution and the mass of water in the solution are defined through the **SOLUTION** or **SOLUTION_SPREAD** data blocks. Internally, all concentrations are converted to molality and the number of moles of each element in solution (including hydrogen and oxygen) is calculated from the molalities and the mass of water. Thus, internally, a solution is simply a list of elements and the number of moles of each element.

PHREEQC allows each reactant to be defined independently. In particular, reactants (**EQUILIBRIUM_PHASES**, **EXCHANGE**, **GAS_PHASE**, **KINETICS**, **REACTION**, **SOLID_SOLUTIONS**, and **SURFACE**) are defined in terms of moles, without reference to a volume or mass of water. Systems are defined by combining a solution with a set of reactants that react either reversibly (**EQUILIBRIUM_PHASES**, **EXCHANGE**, **GAS_PHASE**, **SOLID_SOLUTIONS**, and **SURFACE**) or irreversibly (**KINETICS** or **REACTION**). Essentially, all of the moles of elements in the solution and the reversible reactants are combined, the moles of irreversible reactants are added (or removed), and a new system equilibrium is calculated. Only after system equilibrium is calculated is the mass of water in the system known, and only then the molalities of all entities can be calculated.

For transport calculations, each cell is a system that is defined by the solution and all the reactants contained in keywords that bear the same number as the cell number. The system for the cell is initially defined by the moles of elements that are present in the solution and the moles of each reactant. The compositions of all these entities evolve as the transport calculations proceed.

Keywords

The following sections describe the data input requirements for the program. Each type of data is input through a specific keyword data block. The keywords are listed in alphabetical order. Each keyword data block may have a number of identifiers, many of which are optional. Identifiers may be entered in any order; the line numbers

given in examples for the keyword data blocks are for identification purposes only. Default values for identifier obtained if the identifier is omitted.

ADVECTION

This keyword data block is used to specify the number of cells and the number of "shifts" for an advection simulation. Advection simulations are used to model one dimensional advective or "plug" flow with reactions. No dispersion or diffusion is simulated and no cells with immobile water are allowed. However, all chemical processes modeled by PHREEQC may be included in an advection simulation. The **TRANSPORT** data block may be used to model additional physical processes, such as dispersion, diffusion, and connected cells with immobile water.

Example data block

```
Line 0: ADVECTION
Line 1:      -cells 5
Line 2:      -shifts 25
Line 3:      -time_step 3.15e7 # seconds = 1 yr.
Line 4:      -initial_time 1000
Line 5:      -print_cells 1-3 5
Line 6:      -print_frequency 5
Line 7:      -punch_cells 2-5
Line 8:      -punch_frequency 5
Line 9:      -warnings false
```

Explanation

Line 0: **ADVECTION**

ADVECTION is the keyword for the data block. No other data are input on the keyword line.

Line 1: **-cells** *cells*

-cells--Identifier for number of cells in the advection simulation. Optionally, **cells**, or **-c[ells]**.

cells--Number of cells in the one dimensional column to be used in the advection simulation. Default is 0.

Line 2: **-shifts** *shifts*

-shifts--Identifier for the number of shifts or time steps in the advection simulation. Optionally, **shifts**, or **-sh[ifts]**.

shifts--Number of times the solution in each cell will be shifted to the next higher numbered cell. Default is 0.

Line 3: **-time_step** *time_step*

-time_step--Identifier for time step associated with each advective shift. The identifier is required if kinetic reactions (**KINETICS** data blocks) are part of the advection simulation and optional for other advection simulations. If **-time_step** is defined, then the value for time printed to the selected-output file will be *initial_time* + *advection_shift_number* x *time_step*, if **-time_step** is not defined, the value of time printed to the selected-output file will be the advection shift number. Once **-time_step** is defined, the time step will be used for all subsequent advection simulations until it is redefined. Optionally, **timest**, **-t[imest]**, **time_step**, or **-t[ime_step]**.

time_step--The time in seconds associated with each advective shift. Kinetic reactions will be integrated for this period of time for each advective shift. Default is 0 s.

Line 4: **-initial_time** *initial_time*

-initial_time--Identifier to set the time at the beginning of an advection simulation. The identifier **-initial_time** has effect only if **-time_step** has been set in this or a previous **ADVECTION** data block. The identifier sets the initial value of the variable controlled by **-time** in **SELECTED_OUTPUT** data block. Optionally, **initial_time** or **-i[nitial_time]**.

initial_time--Time (seconds) at the beginning of the advection simulation. Default is the cumulative time including all preceding **ADVECTION** simulations for which **-time_step** has been defined and all preceding **TRANSPORT** simulations.

Line 5: **-print_cells** *list of cell numbers*

-print_cells--Identifier to select cells for which results will be written to the output file. If **-print_cells** is not included, results for all cells will be written to the output file. Once **-print_cells** is defined, the list of cells will be used for all subsequent advection simulations until the list is redefined. Optionally, **print_cells** or **-pr[int_cells]**. Note the hyphen is required in **-print** to avoid a conflict with the keyword **PRINT**.

list of cell numbers--Printing to the output file will occur only for these cell numbers. The list of cell numbers must be delimited by spaces or tabs and may be continued on the succeeding line(s). A range of cell numbers may be included in the list in the form *m-n*, where *m* and *n* are positive integers, *m* is less than *n*, and the two numbers are separated by a hyphen without intervening spaces. Default 1-*cells*.

Line 6: **-print_frequency** *print_modulus*

-print_frequency--Identifier to select shifts for which results will be written to the output file. Once defined, the print frequency will be used for all subsequent advection simulations until it is redefined. Optionally, **print_frequency**, **-print_f[requency]**, **output_frequency**, **-o[utput_frequency]**.

print_modulus--Printing to the output file will occur after every *print_modulus* advection shifts. Default is 1.

Line 7: **-punch_cells** *list of cell numbers*

-punch_cells--Identifier to select cells for which results will be written to the selected-output file. If **-punch_cells** is not included, results for all cells will be written to the selected-output file. Once defined, the list of cells will be used for all subsequent advection simulations until the list is redefined. Optionally, **punch**, **punch_cells**, **-pu[nch_cells]**, **selected_cells**, or **-selected_c[ells]**.

list of cell numbers--Printing to the selected-output file will occur only for these cell numbers. The list of cell numbers must be delimited by spaces or tabs and may be continued on the succeeding line(s). A range of cell numbers may be included in the list in the form *m-n*, where *m* and *n* are positive integers, *m* is less than *n*, and the two numbers are separated by a hyphen without intervening spaces. Default 1-*cells*.

Line 8: **-punch_frequency** *punch_modulus*

-punch_frequency--Identifier to select shifts for which results will be written to the selected-output file. Once defined, the punch frequency will be used for all subsequent advection simulations until it is redefined. Optionally, **punch_frequency**, **-punch_f[requency]**, **selected_output_frequency**, **-selected_o[utput_frequency]**.

punch_modulus--Printing to the selected-output file will occur after every *punch_modulus* advection shifts. Default is 1.

Line 9: **-warnings** [(*True* or *False*)]

-warnings--Identifier enables or disables printing of warning messages for advection calculations. In some cases, advection calculations could produce many warnings that are not errors. Once it is determined that the warnings are not due to erroneous input, disabling the warning messages can avoid generating large output files. Optionally, **warnings**, **warning**, or **-w[arnings]**.

[(*True* or *False*)]--If value is true, warning messages are printed to the screen and the output file; if value is false, warning messages are not printed to the screen or the output file. The value set with **-warnings** is retained in all subsequent advection simulations until changed. Default is true, value at beginning of run is true.

Notes

The capabilities available through the **ADVECTION** data block are a simplified version of a more complete formulation of 1D advective-dispersive-reactive transport that is presented by Appelo and Postma (1993) and implemented in the **TRANSPORT** data block. Calculations using the **ADVECTION** keyword are sufficient for initial investigations, and in comparison to other problems that include dispersion, the calculations are fast. For many systems with limited data, the kinds of calculations available with **ADVECTION** are adequate and appropriate. The **TRANSPORT** data block allows modeling of the additional processes of diffusion, dispersion, and diffusion into stagnant zones. The transport capabilities of the **ADVECTION** keyword in PHREEQC version 2 are equivalent to the capabilities of the **TRANSPORT** keyword in PHREEQC version 1.

In the example data block given in this section, a column of five cells (*cells*) is modeled and 5 pore volumes of filling solution are moved through the column (*shifts*/*cells* is 5). Unless kinetic reactions are modeled, no explicit definition of time is required, only the number of shifts. Also, no distance is explicitly specified for advection calculations, only the number of cells.

The **-time_step** identifier is required if kinetic reactions (**KINETICS** data block) are defined for at least one cell in the column. If kinetic reactions are defined, then an integration is performed for each cell that has kinetic reactions for each advective shift. Kinetic reactions significantly increase the run time of a simulation because the integration of the rates of reaction imposes 1 to 6 (or possibly more) additional batch-reaction calculations for each cell that has kinetic reactions for each advective shift. The total time modeled in the example data block simulation is 25,000 seconds ($time\_step \times shifts$).

By default, the composition of the solution, pure-phase assemblage, exchange assemblage, gas phase, solid-solution assemblage, surface assemblage, and kinetic reactants are printed for each cell for each shift. Use of **-print_cells** and **-print_frequency**, will limit the data written to the output file. The **-print_cells** identifier restricts printing in the output file to the specified cells; in the example data block, results for cells 1, 2, 3, and 5 are printed to the output file. The identifier **-print_frequency** restricts printing in the output file to those advection shifts that are evenly divisible by *print_modulus*. In the example data block, results are printed to the output file after each integer pore volume (5 shifts). Data written to the output file can be further limited with the keyword **PRINT** (see **-reset false**). The **USER_PRINT** data block can be used to calculate quantities to be printed to the output file.

If the **SELECTED_OUTPUT** data block has been defined (recommended), then data specified in the **SELECTED_OUTPUT** and **USER_PUNCH** data blocks are written to the selected-output file. Use of **-punch_cells** and **-punch_frequency** in the **ADVECTION** data block will limit what is written to the selected-output file. The **-punch_cells** identifier restricts printing to the selected-output file to the specified cells; in the example data block, results for cells 2, 3, 4, and 5 are printed to the selected-output file. The identifier **-punch_frequency** restricts printing to the selected-output file to those advection shifts that are evenly divisible by *punch_modulus*. In the example data block, results are printed to the selected-output file after each integer pore volume (5 shifts). All printing to the selected-output file can be switched on or off through the **-selected_output** identifier of the keyword **PRINT**.

Most of the information for advection calculations must be entered with other keywords. This advection calculation assumes that solutions with numbers 0 through 5 have been defined using the **SOLUTION** or **SAVE** data blocks. Solution 0 is the infilling solution and solutions 1 through 5 are the initial solutions in the cells of the column. Other reactants may be defined for each of the cells. Pure-phase assemblages may be defined with **EQUILIBRIUM_PHASES** or **SAVE**, with the number of the assemblage corresponding to the cell number. Likewise, an exchange assemblage, gas phase, solid-solution assemblage, or surface assemblage can be defined for each cell through **EXCHANGE**, **GAS_PHASE**, **SOLID_SOLUTIONS**, **SURFACE**, or **SAVE** data blocks, with the identifying number corresponding to the cell number. Note that ranges of numbers can be used (for example **SOLUTION** 1-5) to define multiple solutions, pure-phase assemblages, exchange assemblages, gas phases, solid-solution assemblages, or surface assemblages and that **SAVE** allows a range of numbers to be used.

The **REACTION** data block can be used to define a stoichiometric reaction that applies to a cell at each shift, with the reaction number corresponding to the cell number. This capability is not very useful because it represents only zero-order kinetics, the reaction rate is constant throughout the advection simulation. The **KINETICS** data block provides a better definition of time-varying reactions for individual cells.

The **MIX** keyword can be used with **ADVECTION** modeling to define simplistic dispersion or lateral inflow to the column. At each shift, solution 0 is moved to cell 1, any stoichiometric reaction or mixing for cell 1 is added, kinetic reactions are integrated while maintaining equilibrium with the contents of cell 1; solution 1 (before mixing and reaction) is moved to cell 2, reaction or mixing for cell 2 is added, kinetic reactions are integrated while maintaining equilibrium with the contents of cell 2; and so on until solution *cells-1* is moved to cell *cells*. The moles of pure phases and kinetic reactants, and the compositions of the exchange assemblage, surface assemblage, and gas phase in each cell are updated with each shift, but only after mixing for the next cell has been accomplished.

Example problems

The keyword **ADVECTION** is used in example problems 11 and 14.

Related keywords

EQUILIBRIUM_PHASES, **EXCHANGE**, **GAS_PHASE**, **KINETICS**, **MIX**, **PRINT**, **REACTION**, **REACTION_TEMPERATURE**, **SAVE**, **SELECTED_OUTPUT**, **SOLID_SOLUTIONS**, **SOLUTION**, **SURFACE**, **TRANSPORT**, **USER_PRINT**, and **USER_PUNCH**.

END

This keyword has no associated data. It ends the data input for a simulation. After this keyword is read by the program, the calculations described by the input for the simulation are performed and the results printed. Additional simulations may follow in the input data set, each in turn will be terminated with an **END** keyword or the end of the file.

Example problems

The keyword **END** is used in all example problems, 1 through 18.

EQUILIBRIUM_PHASES

This keyword data block is used to define the amounts of an assemblage of pure phases that can react reversibly with the aqueous phase. When the phases included in this keyword data block are brought into contact with an aqueous solution, each phase will dissolve or precipitate to achieve equilibrium or will dissolve completely. Pure phases include minerals with fixed composition and gases with fixed partial pressures. Two types of input are available: in one type, the phase itself reacts to equilibrium (or a specified saturation index or gas partial pressure); in the other type, an alternative reaction occurs to the extent necessary to reach equilibrium (or a specified saturation index or gas partial pressure) with the specified pure phase.

Example data block

```
Line 0:  EQUILIBRIUM_PHASES 1 Define amounts of phases in assemblage.
Line 1a:      Chalcedony  0.0     0.0
Line 1b:      CO2(g)      -3.5    1.0
Line 1c:      Gibbsite(c) 0.0     KAlSi3O8  1.0
Line 1d:      Calcite     1.0     Gypsum    1.0
Line 1e:      pH_Fix      -5.0    HCl       10.0
```

Explanation

Line 0: **EQUILIBRIUM_PHASES** [*number*] [*description*]

EQUILIBRIUM_PHASES is the keyword for the data block. Optionally, **EQUILIBRIUM, EQUILIBRIA, PURE_PHASES, PURE.**

number--Positive number to designate the following phase assemblage and its composition. A range of numbers may also be given in the form *m-n*, where *m* and *n* are positive integers, *m* is less than *n*, and the two numbers are separated by a hyphen without intervening spaces. Default is 1.

description--Optional comment that describes the phase assemblage.

Line 1: *phase name* [*saturation index* [(*alternative formula* or *alternative phase*) [*amount*]]]

phase name--Name of a phase. The phase must be defined with **PHASES** input, either in the database file or in the current or previous simulations of the run. The name must be spelled identically to the name used in **PHASES** input (except for case).

saturation index--Target saturation index for the pure phase in the aqueous phase (line 1a); for gases, this number is the log of the partial pressure (line 1b). The target saturation index (partial pressure) may not be attained if the amount of the phase in the assemblage is insufficient. Default is 0.0.

alternative formula--Chemical formula that is added (or removed) to attain the target saturation index (or log partial pressure). By default, the mineral defined by *phase name* dissolves or precipitates to attain the target saturation index. If *alternative formula* is entered, *phase name* does not react; the stoichiometry of *alternative formula* is added or removed from the aqueous phase to attain the target saturation index. *Alternative formula* must be a legitimate chemical formula composed of elements defined to the program. Line 1c indicates that the stoichiometry given by *alternative formula*, $KAlSi_3O_8$ (potassium feldspar), will be added or removed from the aqueous phase until

gibbsite equilibrium is attained. The *alternative formula* and *alternative phase* are mutually exclusive fields.

alternative phase--The chemical formula defined for *alternative phase* is added (or removed) to attain the target saturation index (or log partial pressure). By default, the mineral defined by *phase name* dissolves or precipitates to attain the target saturation index. If *alternative phase* is entered, *phase name* does not react; the stoichiometry of the *alternative phase* is added or removed from the aqueous phase to attain the target saturation index. *Alternative phase* must be defined through **PHASES** input (either in the database file or in the present or previous simulations). Line 1d indicates that the phase gypsum will be added to or removed from the aqueous phase until calcite equilibrium is attained. The *alternative formula* and *alternative phase* are mutually exclusive fields.

amount--Moles of the phase in the phase assemblage or moles of the alternative reaction. This number of moles defines the maximum amount of the mineral or gas that can dissolve. It may be possible to dissolve the entire amount without reaching the target saturation index, in which case the solution will have a smaller saturation index for this phase than the target saturation index. If *amount* is equal to zero, then the phase can not dissolve, but will precipitate if the solution becomes supersaturated with the phase. Default is 10.0 moles.

Notes

If just one number is included on line 1, it is assumed to be the target saturation index (or log partial pressure) and the amount of the phase defaults to 10.0 mol. If two numbers are included on the line, the first is the target saturation index and the second is the amount of the phase present. Line 1 may be repeated to define all pure phases that are assumed to react reversibly. It is possible to include a pure phase that has an amount of zero (line 1a). In this case, chalcedony can not dissolve, but can precipitate if the solution is supersaturated with chalcedony, either by initial conditions, or through dissolution of pure phases or other specified reactions (mixing, stoichiometric or kinetic reactions). It is possible to maintain constant pH conditions by specification of an *alternative formula* and a special phase (**PHASES** input). Line 1e would maintain a pH of 5.0 by adding HCl, provided a phase named "pH_Fix" were defined with reaction $H^+ = H^+$ and log $K = 0.0$ (see example 8, in "Examples"). (Note: If the acid, HCl, is specified and, in fact, a base is needed to attain pH 5.0, it is possible the program will fail to find a solution to the algebraic equations.)

The number of exchange sites can be related to the moles of a phase that are present in an **EQUILIBRIUM_PHASES** phase assemblage (see **EXCHANGE**). As the moles of the phase increase or decrease, the number of exchange sites will increase or decrease. Likewise, the number of surface sites can be related to the moles of a phase that are present in an **EQUILIBRIUM_PHASES** phase assemblage (see **SURFACE**).

For batch reactions, after a pure-phase assemblage has reacted with the solution, it is possible to save the resulting assemblage composition (that is, the identity, target saturation index, and moles of each phase) with the **SAVE** keyword. If the new composition is not saved, the assemblage composition will remain the same as it was before the batch reaction. After it has been defined or saved, the assemblage may be used in subsequent simulations

by the **USE** keyword. **TRANSPORT** and **ADVECTION** calculations automatically update the pure-phase assemblage and **SAVE** has no effect during these calculations.

Example problems

The keyword **EQUILIBRIUM_PHASES** is used in example problems 2, 3, 5, 6, 7, 8, 9, 10, and 14.

Related keywords

ADVECTION, EXCHANGE, PHASES, SAVE equilibrium_phases, SURFACE, TRANSPORT, and **USE equilibrium_phases**.

EXCHANGE

This keyword data block is used to define the amount and composition of an assemblage of exchangers. The initial composition of the exchange assemblage can be defined in two ways, (1) explicitly by listing the composition of each exchange component or (2) implicitly, by specifying that each exchanger is in equilibrium with a solution of fixed composition. The exchange master species, stoichiometries, and log K's for the exchange reactions are defined with the keywords **EXCHANGE_MASTER_SPECIES** and **EXCHANGE_SPECIES**. The number of exchange sites can be fixed; can be related to the amount of a phase in a phase assemblage; or can be related to the amount of a kinetic reactant.

Example data block 1

```
Line 0:  EXCHANGE 10 Measured exchange composition
Line 1a:       CaX2     0.3
Line 1b:       MgX2     0.2
Line 1c:       NaX      0.5
Line 2a:       CaY2     Ca-Montmorillonite    equilibrium_phase 0.165
Line 2b:       NaZ      Kinetic_clay          kinetic_reactant  0.1
```

Explanation 1

Line 0: **EXCHANGE** [*number*] [*description*]

EXCHANGE is the keyword for the data block.

number--Positive number to designate the following exchange assemblage and its composition. A range of numbers may also be given in the form *m-n*, where *m* and *n* are positive integers, *m* is less than *n*, and the two numbers are separated by a hyphen without intervening spaces. Default is 1.

description--Optional comment that describes the exchanger.

Line 1: *exchange formula, amount*

exchange formula--Exchange species including stoichiometry of exchange ion and exchanger.

amount--Quantity of exchange species, in moles.

Line 2: *exchange formula, name,* [(**equilibrium_phase** or **kinetic_reactant**)], *exchange_per_mole*

exchange formula--Exchange species including stoichiometry of exchange ion and exchange site(s). The *exchange formula* must be charge balanced; if no exchange ions are included in the formula, then the exchange site must be uncharged.

name--Name of the pure phase or kinetic reactant that has this kind of exchange site. If *name* is a phase, the amount of the phase in an **EQUILIBRIUM_PHASES** data block with the same number as this exchange number (10, in the example data block) will be used to determine the number of exchange sites. If *name* is a kinetic reactant, the amount of the reactant in a **KINETICS** data block with the same number as this exchange number (10, in the example data block) will be used to determine the number of exchange sites. Some care is needed in defining the stoichiometry of the exchange species if the exchangeable ions are related to a phase or kinetic reactant. The assumption is that some of the ions in the pure phase or kinetic reactant are available for exchange and these ions are defined through one or more entries of Line 2. The stoichi-

ometry of the phase (defined in a **PHASES** data block) or kinetic reactant (defined in a **KINETICS** data block) must contain sufficient amounts of the exchangeable ions. From the example data block, Line 2a, there must be at least 0.165 mol of calcium per mole of Ca-Montmorillonite. From the example data block, Line 2b, there must be at least 0.1 mol of sodium per mole of the reactant "kinetic_clay".

equilibrium_phase or **kinetic_reactant**--If **equilibrium_phase** is used, the *name* on the line is a phase defined in an **EQUILIBRIUM_PHASES** data block. If **kinetic_reactant** is used, the name on the line is the rate name for a kinetic reactant defined in a **KINETICS** data block. Optionally, **e** or **k**, only the first letter is checked. Default is **equilibrium_phase**.

exchange_per_mole--Number of moles of the exchange species per mole of phase or kinetic reactant, unitless (mol/mol).

Notes 1

Line 1 may be repeated to define the entire composition of each exchanger. This example data block defines the amount and composition of three exchangers, X, Y, and Z. Line 2 should be entered only once for each type of exchange site. The total number of exchange sites of X is 1.5 mol and the total concentrations of calcium, magnesium, and sodium on exchanger X are 0.3, 0.2, and 0.5 mol, respectively. When the composition of the exchanger is defined explicitly, such as in this example data block, the exchanger will almost certainly not be in equilibrium with any of the solutions that have been defined. Any batch reaction that includes an explicitly defined exchanger will produce a reaction that causes change in solution and exchange composition.

Exchanger Y is related to the amount of Ca-Montmorillonite in **EQUILIBRIUM_PHASES** 10, where 10 is the same number as the exchange-assemblage number. If m represents the moles of Ca-Montmorillonite in **EQUILIBRIUM_PHASES** 10, then the number of moles of exchangeable component CaY_2 is $0.165m$, and the total number of exchange sites (Y) is $0.33m$ (0.165x2). The stoichiometry of Ca must be at least 0.165 in the formula for Ca-Montmorillonite. During batch-reaction simulations the exchange composition, including the moles of Ca exchanged, will change depending on competing species defined in **EXCHANGE_SPECIES**. In addition, the moles of Ca-Montmorillonite in **EQUILIBRIUM_PHASES** 10 may change, in which case the total moles of the exchange sites (Y) will change.

Exchanger Z is related to the amount of a kinetic reactant that dissolves and precipitates according to a rate expression named "kinetic_clay". The formula for the kinetic reactant is defined in **KINETICS** 10, where 10 is the same number as the exchange-assemblage number. If m represents the moles of kinetic_clay in **KINETICS** 10, then the number of moles of exchangeable component NaZ is $0.1m$, which is equal to the total number of exchange sites. The stoichiometry of Na must be at least 0.1 in the formula for the kinetic reactant. The exchange composition will change during reaction calculations, depending on competing species defined in **EXCHANGE_SPECIES**. In addition, the moles of kinetic_clay in **KINETICS** 10 may change, in which case the total moles of the exchange sites (Z) will change.

Example data block 2

```
Line 0: EXCHANGE 1 Exchanger in equilibrium with solution 1
Line 1a:      X      1.0
Line 1b:      Xa     0.5
```

```
Line 2:      CaY2     Ca-Montmorillonite   equilibrium_phase  0.165
Line 3:      -equilibrate with solution 1
```

Explanation 2

Line 0: **EXCHANGE** [*number*] [*description*]

Same as example data block 1.

Line 1: *exchange site, amount*

exchange site--Only the name of the exchange site needs to be entered.

amount--Quantity of exchange site, in moles.

Line 2: *exchange formula, name,* [(**equilibrium_phase** or **kinetic_reactant**)], *exchange_per_mole* (same as example data block 1).

Line 3: **-equilibrate** *number*

-equilibrate--This string at the beginning of the line indicates that the exchange assemblage is defined to be in equilibrium with a given solution composition. Optionally, **equil**, **equilibrate**, or **-e[quilibrate]**.

number--Solution number with which the exchange assemblage is to be in equilibrium. Any alphabetic characters following the identifier and preceding an integer ("with solution" in line 1) are ignored.

Notes 2

The order of lines 1, 2, and 3 is not important. Line 3 should occur only once within the data block. Lines 1 and 2 may be repeated to define the amounts of other exchangers, if more than one exchanger is present in the assemblage. Example data block 2 requires the program to make a calculation to determine the composition of the exchange assemblage. The calculation will be performed before any batch-reaction calculations to determine the concentrations of each exchange component [such as CaX_2, MgX_2, or NaX (from the default database) provided calcium, magnesium, and sodium are present in solution] that would exist in equilibrium with the specified solution (solution 1 in this example data block). The composition of the solution will not change during this calculation. When an exchange assemblage (defined as in example data block 1 or example data block 2) is placed in contact with a solution during a batch reaction, both the exchange composition and the solution composition will adjust to reach a new equilibrium.

The exchange ions given by the formulas in Lines 2 are not used in the initial exchange-composition calculation. However, the definition of the exchange ions is important for batch-reaction and transport calculations. As the reactant, either a pure phase or a kinetic reactant, dissolves or precipitates, the number of exchange sites varies. Any new sites are initially filled with the exchangeable ions given in Lines 2. When exchange sites are removed, for example when a pure phase dissolves, then the net effect is to subtract from the pure phase formula the amount of the exchange ions defined in Lines 2 and add an equivalent amount of ions as defined by the exchanger composition. As an example, suppose some Ca-montmorillonite forms. Initially, calcium is in the exchange positions, but sodium replaces part of the calcium on the exchanger. When the montmorillonite dissolves again, the calcium in the formula for the phase is added to solution, the exchange ion (calcium from Line 2) is removed from solution, and the sodium and calcium from the exchanger is added to solution; the net effect is dissolution of (Na, Ca)-montmorillonite. Note that equilibrium for Ca-montmorillonite always uses the same

mass-action equation, which include [illegible]y calcium, even though the composition of the phase is changing. Note also that this formulation implies that a pure Na-montmorillonite can never be attained because calcium must always be present to attain equilibrium with Ca-montmorillonite.

It is possible to realize a complete exchange of sodium and calcium by defining Y without cations under **EXCHANGE**, and a new equilibrium with only the structural ions of montmorillonite under **PHASES**. The combined reaction of exchanger and equilibrium phase must be electrically neutral. In the example data block, the montmorillonite would be defined with a positive charge deficit of 0.165. When montmorillonite forms, the exchange sites Y increase in proportion and take cations from solution to exactly balance the charge deficit. Note that log_k for montmorillonite is adjusted by $\log_{10}(0.001^{0.165})$ to account for an estimated contribution of 1 mmol/kgw Ca in solution. Yet another possibility is to use the capabilities of the **SOLID_SOLUTIONS** data block to define a variable composition solid solution between calcium and sodium montmorillonite end members.

```
EXCHANGE 1 Exchanger in equilibrium with solution 1
    Y Montmorillonite equilibrium_phase  0.165
    -equilibrate with solution 1
PHASES
    -no_check          # must use no_check because of unbalanced equation
Montmorillonite        # Montmorillonite has 0.165 mol Y-/mol
Al2.33Si3.67O10(OH)2 + 12 H2O = 2.33 Al(OH)4- + 3.67 H4SiO4 + 2 H+
    log_k     -44.532 #Assume a_Ca = 0.001 at equilibrium
    delta_h   58.373  kcal
```

An exchanger can be defined with a fixed number of sites initially, but through special definition of a kinetic reactant, the number of sites can vary with reaction progress. Concentration changes in the number of exchange sites can be included in the **KINETICS** keyword, under **-formula**. The combination of exchanger and kinetic reaction must be neutral.

```
EXCHANGE 1
    # Z+ is related to Goethite, initial amount is 0.2 * m_go = 0.02
    Z   0.02
    -equil 1
KINETICS 1
    # Z has a charge of +1.0, Fe(OH)2+ sorbs anions.
    -formula  FeOOH 0.8  Fe(OH)2 0.2  Z  -0.2
    m    0.1
```

After a batch reaction has been simulated, it is possible to save the resulting exchange assemblage composition with the **SAVE** keyword. If the new composition is not saved, the exchange assemblage composition will remain the same as it was before the batch reaction. After it has been defined or saved, the exchange assemblage can be used in subsequent simulations through the **USE** keyword.

Example problems

The keyword **EXCHANGE** is used in example problems 11, 12, 13 and 14.

Related keywords

EQUILIBRIUM_PHASES, **EXCHANGE_MASTER_SPECIES**, **EXCHANGE_SPECIES**, **KINETICS**, **SAVE exchange**, and **USE exchange.**

EXCHANGE_MASTER_SPECIES

This keyword data block is used to define the correspondence between the name of an exchange site and an exchange species that is used as the master species in calculations. Normally, this data block is included in the database file and only additions and modifications are included in the input file.

Example data block

```
Line 0: EXCHANGE_MASTER_SPECIES
Line 1a:      X      X-
Line 1b:      Xa     Xa-
```

Explanation

Line 0: **EXCHANGE_MASTER_SPECIES**

Keyword for the data block. No other data are input on the keyword line.

Line 1: *exchange name, exchange master species*

exchange name--Name of an exchange site, X and Xa in this example data block. It must begin with a capital letter, followed by zero or more lower case letters or underscores ("_").

exchange master species--Formula for the master exchange species, X^- and Xa^- in this example data block.

Notes

All half-reactions for the exchanger (X and Xa, in this example data block) must be written in terms of the master exchange species (X^- and Xa^- in this example data block). Each exchange master species must be defined by an identity reaction with log *K* of 0.0 in **EXCHANGE_SPECIES** input. Any additional exchange species and associated reactions must be defined with **EXCHANGE_SPECIES** input.

Example problems

The keyword **EXCHANGE_MASTER_SPECIES** is not used in the example problems. See listing of default database file in Attachment B for an example.

Related keywords

EXCHANGE, **EXCHANGE_SPECIES**, **SAVE exchange**, and **USE exchange**.

EXCHANGE_SPECIES

This keyword data block is used to define a half-reaction and relative log *K* for each exchange species. Normally, this data block is included in the database file and only additions and modifications are included in the input file.

Example data block

```
Line 0:  EXCHANGE_SPECIES
Line 1a:       X- = X-
Line 2a:            log_k      0.0
Line 1b:       X- + Na+ = NaX
Line 2b:            log_k      0.0
Line 3b:            -gamma     4.     0.075
Line 1c:       2X- + Ca+2 = CaX2
Line 2c:            log_k      0.8
Line 4c:            -davies
Line 1d:       Xa- = Xa-
Line 2d:            log_k      0.0
Line 1e:       Xa- + Na+ = NaXa
Line 2e:            log_k      0.0
Line 1f:       2Xa- + Ca+2 = CaXa2
Line 2f:            log_k      2.0
```

Explanation

Line 0: **EXCHANGE_SPECIES**

Keyword for the data block. No other data are input on the keyword line.

Line 1: *Association reaction*

Association reaction for exchange species. The defined species must be the first species to the right of the equal sign. The association reaction must precede any identifiers related to the exchange species. Master species have an identity reaction (lines 1a and 1d).

Line 2: **log_k** log *K*

log_k--Identifier for log *K* at 25°C. Optionally, **-log_k**, **logk**, **-l[og_k]**, or **-l[ogk]**.

log K--Log *K* at 25°C for the reaction. Unlike log *K* for aqueous species, the log *K* for exchange species is implicitly relative to a single exchange species. In the default database file, sodium (NaX) is used as the reference and the reaction $X^- + Na^+ = NaX$ is given a log *K* of 0.0 (line 2b). The log *K* for the exchange reaction for the reaction given in line 2c is then numerically equal to the log *K* for the reaction $2NaX + Ca^{+2} = CaX_2 + 2Na^+$. Master species have log *K* of 0.0 (lines 2a and 2d); reactions for reference species also have log *K* of 0.0 (lines 2b and 2e). Default is 0.0.

Line 3: **-gamma** *Debye-Hückel a, Debye-Hückel b*

-gamma--Indicates WATEQ Debye-Hückel equation will be used to calculate an activity coefficient for the exchange species. If **-gamma** or **-davies** is not input for an exchange species, the activity of the species is equal to its equivalent fraction. If **-gamma** is entered, then an activity coefficient of

the form of WATEQ (Truesdell and Jones, 1974), $\log\gamma = \frac{-Az_e^2\sqrt{\mu}}{1+Ba^o\sqrt{\mu}} + b\mu$, is multiplied times the equivalent fraction to obtain activity for the exchange species. In this equation, γ is the activity coefficient, μ is ionic strength, A and B are constants at a given temperature, and z_e is the number of equivalents of exchanger in the exchange species. Optionally, **gamma** or **-g[amma]**.

Debye-Hückel a--Parameter a^o in the WATEQ activity-coefficient equation.

Debye-Hückel b--Parameter b in the WATEQ activity-coefficient equation.

Line 4: **-davies**

-davies--Indicates the Davies equation will be used to calculate an activity coefficient. If **-gamma** or **-davies** is not input for an exchange species, the activity of the species is equal to its equivalent fraction. If **-davies** is entered, then an activity coefficient of the form of the Davies equation, $\log\gamma = -Az_e^2\left(\frac{\sqrt{\mu}}{1+\sqrt{\mu}} - 0.3\mu\right)$, is multiplied times the equivalent fraction to obtain activity for the exchange species. In this equation, γ is the activity coefficient, μ is ionic strength, A is a constant at a given temperature, and z_e is the number of equivalents of exchanger in the exchange species. Optionally, **davies** or **-d[avies]**.

Notes

Lines 1 and 2 may be repeated as necessary to define all of the exchange reactions, with line 1 preceding line 2 for each exchange species. One identity reaction that defines the exchange master species (in example data block, lines 1a and 2a, 1d and 2d) and one reference half-reaction are needed for each exchanger. The identity reaction has a log K of 0.0. The reference half-reaction for each exchanger also will have a log K of 0.0 (in example data block, lines 1b and 2b, 1e and 2e); in the default database file the reference half-reaction is $Na^+ + X^- = NaX$. Multiple exchangers may be defined simply by defining multiple exchange master species and additional half-reactions involving these master species, as in this example data block.

The theory for activities of exchange species is not well developed. In PHREEQC, the activity of an exchange species is by default assumed to be equal to the equivalent fraction of the species relative to the total equivalents of exchanger. The **-gamma** identifier allows the equivalent fraction to be multiplied by an activity coefficient to obtain the activity of an exchange species. This activity coefficient is identical to the activity coefficient for an aqueous species calculated by using the WATEQ Debye-Hückel equation. The Davies equation can be used to calculate the activity coefficient of the exchange species by specifying the **-davies** identifier. The use of these equations is strictly empirical and is motivated by the observation that these activity corrections provide a better fit to some experimental data.

Temperature dependence of log K can be defined with the standard enthalpy of reaction (identifier **-delta_h**) using the van't Hoff equation or with an analytical expression (**-analytical_expression**). See **SOLUTION_SPECIES** or **PHASES** for examples.

The identifier **-no_check** can be used to disable checking charge and elemental balances (see **SOLUTION_SPECIES**). The use of **-no_check** is not recommended. By default, the equation given for the exchange species (line 1) is used to determine the mass-action equation and the contribution of the species to each

mole-balance equation. Alternatively, the contribution of the species to each mole-balance equation can be defined using the **-mole_balance** identifier. See **SOLUTION_SPECIES** and **SURFACE_SPECIES** for an example. If the **-no_check** identifier is used, then the **-mole_balance** identifier is required.

Example problems

The keyword **EXCHANGE_SPECIES** is used in example problems 12, 13, and 18. See also listing of default database file in Attachment B for examples.

Related keywords

EXCHANGE, EXCHANGE_MASTER_SPECIES, SAVE exchange, and **USE exchange.**

GAS_PHASE

This keyword data block is used to define the composition of a fixed-total-pressure or a fixed-volume multicomponent gas phase. A **GAS_PHASE** data block is not needed if fixed partial pressures of gas components are desired; use **EQUILIBRIUM_PHASES** instead. The gas phase defined with this keyword data block subsequently may be equilibrated with an aqueous phase in combination with pure-phase, surface, exchange, and solid-solution assemblages in batch-reaction calculations. As a consequence of batch reactions, a fixed-pressure gas phase may exist or not, depending on the sum of the partial pressures of the dissolved gases in solution. A fixed-volume gas phase always contains some amount of each gas component that is present in solution. The initial composition of a fixed-pressure gas phase is defined by the partial pressures of each gas component. The initial composition of a fixed-volume gas may be defined by the partial pressures of each gas component or may be defined to be that which is in equilibrium with a fixed-composition aqueous phase. The thermodynamic properties of the gas components are defined with **PHASES** input.

Example data block 1. Fixed-pressure gas phase

```
Line 0:  GAS_PHASE 1-5  Air
Line 1:        -fixed_pressure
Line 2:        -pressure        1.0
Line 3:        -volume          1.0
Line 4:        -temperature     25.0
Line 5a:       CH4(g)           0.0
Line 5b:       CO2(g)           0.000316
Line 5c:       O2(g)            0.2
Line 5d:       N2(g)            0.78
```

Explanation 1

Line 0: **GAS_PHASE** [*number*] [*description*]

GAS_PHASE is the keyword for the data block.

number--Positive number to designate the following gas phase and its composition. A range of numbers may also be given in the form *m-n*, where *m* and *n* are positive integers, *m* is less than *n*, and the two numbers are separated by a hyphen without intervening spaces. Default is 1.

description--Optional comment that describes the gas phase.

Line 1: **-fixed_pressure**

-fixed_pressure--Identifier defining the gas phase to be one that has a fixed total pressure, that is a gas bubble. A fixed-pressure gas phase is the default if neither the **-fixed_pressure** nor the **-fixed_volume** identifier is used. Optionally **fixed_pressure**, or **-fixed_p[ressure]**.

Line 2: **-pressure** *pressure*

-pressure--Identifier defining the fixed pressure of the gas phase that applies during all batch-reaction and transport calculations. Optionally **pressure**, or **-p[ressure]**.

pressure--The pressure of the gas phase, in atmospheres. Default is 1.0 atm.

Line 3: **-volume** *volume*

-volume--Identifier defining the initial volume of the fixed-pressure gas phase. Optionally, **volume**, or **-v[olume]**.

volume--The initial volume of the fixed-pressure gas phase, in liters. The *volume* along with *temp* and *partial pressure* are used to calculate the initial moles of each gas component in the fixed-pressure gas phase. Default is 1.0 liter.

Line 4: **-temperature** *temp*

-temperature--Identifier defining the initial temperature of the gas phase. Optionally, **temperature**, or **-t[emperature]**.

temp--The initial temperature of the gas phase, in Celsius. The *temp* along with *volume* and *partial pressure* are used to calculate the initial moles of each gas component in the fixed-pressure gas phase. Default is 25.0.

Line 5: *phase name, partial pressure*

phase name--Name of a gas component. A phase with this name must be defined by **PHASES** input in the database or input file.

partial pressure--Initial partial pressure of this component in the gas phase, in atmospheres. The *partial pressure* along with *volume* and *temp* are used to calculate the initial moles of this gas component in the fixed-pressure gas phase.

Notes 1

Line 5 may be repeated as necessary to define all of the components initially present in the fixed-pressure gas phase as well as any components which may subsequently enter the gas phase. The initial moles of any gas component that is defined to have a positive partial pressure in **GAS_PHASE** input will be computed using the ideal gas law, $n = PV/RT$, where n is the moles of the gas, P is the defined partial pressure (line 5), V is given by **-volume**, and T is given by **-temperature** (converted to Kelvin). It is likely that the sum of the partial pressures of the defined gases will not be equal to the pressure given by **-pressure**. However, when the initial moles of gas components are brought in contact with a solution during a batch-reaction simulation, the moles of gases and volume of the gas phase will adjust so that each component is in equilibrium with the solution and the total pressure (sum of the partial pressures) is that specified by **-pressure**. It is possible that the gas phase will not exist if the sum of the partial pressures of dissolved gases does not exceed the pressure given by **-pressure**.

Some gas components may be defined to have initial partial pressures of zero. In this case, no moles of that component will be present initially, but the component may enter the gas phase when in contact with a solution that contains that component. If no gas phase exists initially, the initial partial pressures of all components should be set to 0.0; a gas phase may subsequently form if batch reactions cause the sum of the partial pressures of the gas components to exceed *pressure*.

Example data block 2. Fixed-volume gas phase: Define initial moles of components with partial pressures

```
Line 0:  GAS_PHASE 1-5  Air
Line 1:        -fixed_volume
Line 2:        -volume          1.0
Line 3:        -temperature     25.0
Line 4a:       CH4(g)           0.0
```

```
Line 4b:        CO2(g)          0.000316
Line 4c:        O2(g)           0.2
Line 4d:        N2(g)           0.78
```

Explanation 2

Line 0: **GAS_PHASE** [*number*] [*description*]

GAS_PHASE is the keyword for the data block.

number--positive number to designate the following gas phase and its composition. A range of numbers may also be given in the form *m-n*, where *m* and *n* are positive integers, *m* is less than *n*, and the two numbers are separated by a hyphen without intervening spaces. Default is 1.

description--Optional comment that describes the gas phase.

Line 1: **-fixed_volume**

-fixed_volume--Identifier defining the gas phase to be one that has a fixed volume (not a gas bubble). A fixed-pressure gas phase is the default if neither the **-fixed_pressure** nor the **-fixed_volume** identifier is used. Optionally **fixed_volume**, or **-fixed_v[olume]**.

Line 2: **-volume** *volume*

-volume--Identifier defining the volume of the fixed-volume gas phase, which applies for all batch-reaction or transport calculations. Optionally, **volume**, or **-v[olume]**.

volume--The volume of the fixed-volume gas phase, in liters. Default is 1.0 liter.

Line 3: **-temperature** *temp*

-temperature--Identifier defining the initial temperature of the gas phase. Optionally, **temperature**, or **-t[emperature]**.

temp--The initial temperature of the gas phase, in Celsius. Default is 25.0.

Line 4: *phase name, partial pressure*

phase name--Name of a gas component. A phase with this name must be defined by **PHASES** input in the database or input file.

partial pressure--Initial partial pressure of this component in the gas phase, in atmospheres. The *partial pressure* along with *volume* and *temp* are used to calculate the initial moles of this gas component in the fixed-volume gas phase.

Notes 2

Line 4 may be repeated as necessary to define all of the components initially present in the fixed-volume gas phase as well as any components which may subsequently enter the gas phase. The initial moles of any gas component that is defined to have a positive partial pressure will be computed using the ideal gas law, $n = PV/RT$, where n is the moles of the gas, P is the defined partial pressure (line 4), V is given by **-volume**, and T is given by **-temperature** (converted to Kelvin). When the initial moles of gas components are brought in contact with a solution during a batch-reaction simulation, the total pressure, the partial pressures of the gas components in the gas phase, and the partial pressures of the gas components in the aqueous phase will adjust so that equilibrium is established for each component. A constant-volume gas phase always exists unless all of the gas components are absent from the system. The identifier **-pressure** is not used for a fixed-volume gas phase.

Some gas components may be defined to have initial partial pressures of zero. In this case, no moles of that component will be present initially, but the component will enter the gas phase when in contact with a solution containing the component.

Example data block 3. Fixed-volume gas phase: Define initial moles of components by equilibrium with a solution

```
Line 0:  GAS_PHASE 1-5  Air
Line 1:        -fixed_volume
Line 2:        -equilibrium with solution 10
Line 3:        -volume          1.0
Line 4a:       CH4(g)
Line 4b:       CO2(g)
Line 4c:       O2(g)
Line 4d:       N2(g)
```

Explanation 3

Line 0: **GAS_PHASE** [*number*] [*description*]

GAS_PHASE is the keyword for the data block.

number--Positive number to designate the following gas phase and its composition. A range of numbers may also be given in the form *m-n*, where *m* and *n* are positive integers, *m* is less than *n*, and the two numbers are separated by a hyphen without intervening spaces. Default is 1.

description--Optional comment that describes the gas phase.

Line 1: **-fixed_volume**

-fixed_volume--Identifier defining the gas phase to be one that has a fixed volume (not a gas bubble). A fixed-pressure gas phase is the default if neither the **-fixed_pressure** nor the **-fixed_volume** identifier is used. Optionally **fixed_volume**, or **-fixed_v[olume]**.

Line 2: **-equilibrium** *number*

-equilibrium--Identifier indicates that the fixed-volume gas phase is defined to be in equilibrium with a solution of a fixed composition. This identifier may only be used with the **-fixed_volume** identifier. Optionally, **equil**, **equilibrium**, **-e[quilibrium]**, **equilibrate**, **-e[quilibrate]**.

number--Solution number with which the fixed-volume gas phase is to be in equilibrium. Any alphabetic characters following the identifier and preceding an integer ("with solution" in line 2) are ignored.

Line 3: **-volume** *volume*

-volume--Identifier defining the volume of the fixed-volume gas phase, which applies for all batch-reaction or transport calculations. Optionally, **volume**, or **-v[olume]**.

volume--The volume of the fixed-volume gas phase, in liters. Default is 1.0 liter.

Line 4: *phase name*

phase name--Name of a gas component. A phase with this name must be defined by **PHASES** input in the database or input file.

Notes 3

Line 4 may be repeated as necessary to define all of the components that may be present in the fixed-volume gas phase. The **-equilibrate** identifier specifies that the initial moles of the gas components are to be calculated by equilibrium with solution 10. This calculation is termed an "initial gas-phase-composition calculation". During this calculation, the composition of solution 10 does not change, only the moles of each component in the gas phase are calculated. A constant-volume gas phase always exists unless all of the gas components are absent from the system. When the **-equilibrate** identifier is used, the identifiers **-pressure** and **-temperature** are not needed and initial partial pressures for each gas component need not be specified; the partial pressures for the gas components are calculated from the partial pressures in solution and the temperature is equal to the solution temperature. The **-equilibrate** identifier cannot be used with a fixed-pressure gas phase.

A gas component may have an initial partial pressure of zero, because the solution with which the gas phase is in equilibrium does not contain that gas component. In this case, no moles of that component will be present initially, but the component may enter the gas phase when the gas is in contact with another solution that does contain that component.

Example problems

The keyword **GAS_PHASE** is used in example problem 7.

Related keywords

EQUILIBRIUM_PHASES, PHASES, SAVE gas_phase, and **USE gas_phase.**

INCREMENTAL_REACTIONS

This keyword data block is included mainly to speed up batch-reaction calculations that include kinetic reactions (**KINETICS** keyword). The keyword has no effect on transport calculations. By default (**INCREMENTAL_REACTIONS false**), for each time t_i given by **-steps** in the **KINETICS** keyword data block, rates of kinetic reactions are integrated from time 0 to t_i. This default repeats the integration over early times for each reaction step even though the early times may be the most CPU-intensive part. If **INCREMENTAL_REACTIONS** is set to true, the values of t_i are the incremental times for which to integrate the rates; each kinetic calculation (denoted by i) integrates over the time interval from $\sum_{n=0}^{i-1} t_n$ to $\sum_{n=0}^{i} t_n$. **INCREMENTAL_REACTIONS** has a similar effect for **-steps** in the **REACTION** data block.

Example data block

```
Line 0:  INCREMENTAL_REACTIONS True
```

Explanation

Line 0: **INCREMENTAL_REACTIONS** [(*True* or *False*)]

INCREMENTAL_REACTIONS is the keyword for the data block. If value is **true**, reaction steps for **REACTION** and time steps for **KINETICS** data blocks are incremental amounts of reaction and time that add to previous reactions steps. If value is **false**, reaction steps and time steps are total amounts of reaction and time, independent of previous reaction steps. Default (if neither true nor false is entered) is **true**. Initial setting at beginning of run is **false**.

Notes

Frequently, kinetic reactions are faster at early times and slower at later times. The integration of kinetic reactions for the early times is CPU intensive because the rates must be evaluated at many time subintervals to achieve an accurate integration of the rate equations when reactions are fast. If the time steps in the **KINETICS** data block are 0.1, 1, 10, and 100 s and the time steps are not incremental (default at initialization of a run), then the kinetic reactions will be integrated from 0 to 0.1, 0 to 1, 0 to 10, and 0 to 100 s; the early part of the reactions (0 to 0.1 s) must be integrated for each specified time. By using incremental time steps, the kinetic reactions will be integrated from 0 to 0.1, 0.1 to 1.1, 1.1 to 11.1, and 11.1 to 111.1; the results from the previous time step are used as the starting point for the next time step, and integrating over the same early time interval is avoided.

If the time steps in the **KINETICS** data block are defined as "**-steps** 100 **in** 2 **steps**" and **INCREMENTAL_REACTIONS false**, then the kinetic reactions will be integrated from 0 to 50 and 0 to 100 s. By using **INCREMENTAL_REACTIONS true**, the kinetic reactions will be integrated from 0 to 50 and 50 to 100 s. Although the calculation procedure differs, results of calculations using the "**in**" form of data input should be the same for **INCREMENTAL_REACTIONS true** or **false**.

For consistency, the **INCREMENTAL_REACTIONS** keyword also has an effect on the interpretation of steps defined in the **REACTION** data block. If the steps in the **REACTION** data block were 0.1, 1, 10, and 100

mmol, then by default, solution compositions would be calculated after a total of 0.1, 1, 10, and 100 mmol of reaction had been added to the initial solution. By using incremental reaction steps, solution compositions would be calculated after a total of 0.1, 1.1, 11.1, and 111.1 mmol of reaction had been added.

If the time steps in the **REACTION** data block are defined as "**-steps** 1 **in** 2 **steps**" and **INCREMENTAL_REACTIONS false** (default), then the solution composition will be calculated after 0.5 moles of reaction are added to the initial solution and after 1 mole of reaction has been added to the initial solution. By using **INCREMENTAL_REACTIONS true**, the solution composition will be calculated after 0.5 moles of reaction are added to the initial solution and again after an additional 0.5 moles of reaction are added to the reacted solution. Although the calculation procedure differs, results of calculations using the "**in**" form of data input should be the same for **INCREMENTAL_REACTIONS true** or **false**.

If **INCREMENTAL_REACTIONS true**, **REACTION** is defined with a list of steps, and more batch-reaction steps (maximum number of steps defined in **KINETICS**, **REACTION**, and **REACTION_TEMPERATURE**) than **REACTION** steps are defined, then the last reaction step is repeated for the additional batch-reaction steps. Thus the reaction continues to be added to solution during the final batch-reaction steps. If no additional reaction is desired in these final batch-reaction steps, then additional reaction amounts equal to zero should be entered in the **REACTION** data block. Similarly, if more batch-reaction steps are defined than kinetic steps, the final time step from the **KINETICS** data block will be used for the final batch-reaction steps.

If "**in**" is used in **-steps** in the **REACTION** data block and the number of batch-reaction steps is greater than the number of steps defined in the **REACTION** data block, then the reaction step is zero for **REACTION** in the remaining batch-reaction steps. Likewise, if "**in**" is used in **-steps** in the **KINETICS** data block, and the number of batch-reaction steps is greater than the number steps defined in the **KINETICS** data block, then the time step for kinetic reactions in the remaining batch-reaction steps will be zero.

The incremental approach is not implemented for the **MIX** keyword. If a **MIX** data block is used, then solutions are mixed only once before any reaction or kinetic steps. **REACTION_TEMPERATURE** steps are always non-incremental.

Example problems

The keyword **INCREMENTAL_REACTIONS** is used in example problems 6C and 9.

Related keywords

KINETICS, **MIX**, and **REACTION**.

INVERSE_MODELING

This keyword data block is used to specify the information needed for an inverse modeling calculation. Inverse modeling attempts to determine sets of mole transfers of phases that account for changes in water chemistry between one or a mixture of initial water compositions and a final water composition. Isotope mole balance, but not isotope fractionation, can be included in the calculations. The data block includes definition of the solutions, phases, and uncertainty limits used in the calculations.

Example data block

```
Line 0:  INVERSE_MODELING 1
Line 1:       -solutions        10 3 5
Line 2:       -uncertainty      0.02   0.04
Line 3:       -phases
Line 4a:           Calcite      force  pre   13C   -1.0   1
Line 4b:           Anhydrite    force  dis   34S   13.5   2
Line 4c:           CaX2
Line 4d:           NaX
Line 5:       -balances
Line 6a            pH           0.1
Line 6b:           Ca           0.01      -0.005
Line 6c:           Alkalinity   -1.0e-6
Line 6d:           Fe           0.05      0.1       0.2
Line 7:       -isotopes
Line 8a:            13C         0.05      0.1       0.05
Line 8b:            34S         1.0
Line 9:       -range            10000
Line 10:      -minimal
Line 11:      -tolerance        1e-10
Line 12:      -force_solutions  true      false
Line 13:      -uncertainty_water  0.55    # moles (~1%)
Line 14:      -mineral_water    false
```

Explanation

Line 0: **INVERSE_MODELING** [*number*] [*description*]

INVERSE_MODELING is the keyword for the data block.

number--Positive number to designate the following inverse-modeling definition. Default is 1.

description--Optional comment that describes the inverse-modeling calculation.

Line 1: **-solutions**, *list of solution numbers*

-solutions--Identifier that indicates a list of solution numbers follows on the same line. Optionally, **sol**, or **-s[olutions]**. Note the hyphen is required to avoid conflict with the keyword **SOLUTION**.

list of solution numbers--List of solution numbers to use in mole-balance calculations. At least two solution numbers are required and these solutions must be defined by **SOLUTION** input or by **SAVE** after a batch-reaction calculation in the current or previous simulations. The final solution number is listed last, all but the final solution are termed "initial solutions". If more than one initial solution is listed, the initial solutions are assumed to mix to form the final solution. The

mixing proportions of the initial solutions are calculated in the modeling process. In the example data block (line 1), solution 5 is to be made by mixing solutions 10 and 3 in combination with phase mole transfers.

Line 2: **-uncertainty**, *list of uncertainty limits*

-uncertainty--Identifier that indicates a list of default uncertainty limits for each solution follows on the same line. The uncertainty limits defined with **-uncertainty** do not apply to pH; default for pH is 0.05 pH units and may be changed with the **-balances** identifier. If **-uncertainty** is not entered, the program uses 0.05. The default uncertainty limits can be overridden for individual elements or element valence states using **-balances** identifier. Optionally, **uncertainty, uncertainties, -u[ncertainty]**, or **-u[ncertainties]**.

list of uncertainty limits--List of default uncertainty limits that are applied to each solution in the order given by **-solutions**. The first uncertainty limit in the list is applied to all the element and element valence states in the first solution listed in **-solutions**. The second uncertainty limit in the list is applied to all the element and element valence states in the second solution listed in **-solutions** and so on. If fewer uncertainty limits are entered than the number of solutions, the final uncertainty limit in the list is used for the remaining solutions. Thus, if only one uncertainty limit is entered, it is applied to all solutions. The uncertainty limit may have two forms: (1) if the uncertainty limit is positive, it is interpreted as a fraction to be used to calculate the uncertainty limit for each element or element valence state; a value of 0.02 indicates an uncertainty limit of 2 percent of the moles of each element in solution will be used, and (2) if the uncertainty limit is negative, it is interpreted as an absolute value in moles to use for each mole-balance constraint. The second form is rarely used in **-uncertainty** input. In this example data block, the default uncertainty limit for the first solution is set to 0.02, which indicates that the concentration of each element in the first solution (solution 10) is allowed to vary up to plus or minus 2 percent, and a default uncertainty limit of 4 percent will be applied to each element and valence state in the second solution (solution 3) and all remaining solutions (solution 5 in this case).

Line 3: **-phases**

-phases--Identifier that indicates a list of phases to be used in inverse modeling follows on succeeding lines. Optionally, **phase, phase_data, -p[hases]**, or **-p[hase_data]**. Note the hyphen is required in **-phases** to avoid conflict with the keyword **PHASES**.

Line 4: *phase name* [**force**] [(**dissolve** or **precipitate**)] [*list of isotope name, isotope ratio, isotope uncertainty limit*]

phase name--Name of a phase to be used in inverse modeling. The phase must be defined in **PHASES** input or it must be a charge-balanced exchange species defined in **EXCHANGE_SPECIES** input. Any phases and exchange species defined in the database file or in the current or previous simulations are available for inverse modeling. Only the chemical reaction in **PHASES** or **EXCHANGE_SPECIES** input is important; the log K is not used in inverse-modeling calculations.

force--The phase is included ("forced") to be in the range calculation (see line 9) whether or not the phase mole transfer is nonzero. This will give another degree of freedom to the range calculation

for models that do not include the phase and the resulting range of mole transfers may be larger. The order of this option following the phase name is not important. Optionally, **f[orce]**.

dissolve or **precipitate**--The phase may be constrained only to enter the aqueous phase, "**dissolve**", or leave the aqueous phase, "**precipitate**". Any set of initial letters from these two words are sufficient to define a constraint.

list of isotope name, isotope ratio, isotope uncertainty limit--Isotopic information for the phase may be defined for one or more isotopes by appending (to line 4) triplets of *isotope name, isotope ratio, isotope uncertainty limit.*

isotope name--Isotope name written with mass number first followed by element name with no intervening spaces.

isotope ratio--Isotope ratio for this isotope of this element (*isotope name*) in the phase, frequently permil, but percent or other units can be used. Units must be consistent with the units in which this isotope of the element is defined in **SOLUTION** input.

isotope uncertainty--Uncertainty limit for isotope ratio in the phase. Units must be consistent with the units for *isotope ratio* and units in which this isotope of this element is defined in **SOLUTION** input.

Line 5: **-balances**

-balances--Identifier that indicates a list of element or element-valence-state name follow on succeeding lines. Optionally, **bal, balance, balances,** or **-b[alances]**.

Line 6: *element or valence state name* [*list of uncertainty limits*]

element or valence state name--Name of an element or element valence state to be included as a mole-balance constraint in inverse modeling. The identifier **-balances** is used for two purposes: (1) to include mole-balance equations for elements not contained in any of the phases (**-phases**) and (2) to override the uncertainty limits defined with **-uncertainty** (or the default uncertainty limits) for elements, element valences states, or pH. Mole-balance equations for all elements that are found in the phases of **-phases** input are automatically included in inverse modeling with the default uncertainty limits defined by the **-uncertainties** identifier; mole-balance equations for all valence states of redox elements are included if the element is in any of the phases of **-phases**.

list of uncertainty limits--List of uncertainty limits for the specified element or element valence-state constraint. It is possible to input an uncertainty limit for *element or valence state name* for each solution used in inverse modeling (as defined by **-solutions**). If fewer uncertainty limits are entered than the number of solutions, the final uncertainty limit in the list is used for the remaining solutions. Thus, if only one uncertainty limit is entered, it is used for the given element or element valence state for all solutions. The uncertainty limit for pH must be given in standard units. Thus, the uncertainty limit in pH given on line 6a is 0.1 pH units for all solutions. The uncertainty limits for elements and element valence states (but not for pH) may have two forms: (1) if the uncertainty limit is positive, it is interpreted as a fraction that when multiplied times the moles in solution gives the uncertainty limit in moles. A value of 0.02 would indicate an uncertainty limit of 2 percent of the moles in solution; and (2) if the uncertainty limit is negative,

it is interpreted as an absolute value in moles to use for the solution in the mole-balance equation for *element* or *valence state name*. In the example data block, line 6b, the uncertainty limit for calcium in solution 10 is 1 percent of the moles of calcium in solution 10. The uncertainty limit for calcium in solution 3 and 5 is 0.005 moles. The uncertainty limit for iron (line 6d) is 5 percent in solution 10, 10 percent in solution 3, and 20 percent in solution 5.

Line 7: **-isotopes**

-isotopes--Identifier that specifies mole balance for the isotopes listed on succeeding lines will be included in the calculation. Optionally, **isotopes** or **-i[sotopes]**.

Line 8: *isotope_name, list of uncertainty limits*

isotope_name--Name of an isotope for which mole balance is desired. The name must be written with mass number first followed by element name or redox state with no intervening spaces.

list of uncertainty limits--List of uncertainty limits for the specified isotope for the solutions used in inverse modeling (as defined by **-solutions**). If fewer uncertainty limits are entered than the number of solutions, the final uncertainty limit in the list is used for the remaining solutions. Thus, if only one uncertainty limit is entered, it is used for the given isotope for all solutions. In the example data block, the uncertainty limit for carbon-13 (line 8a) is 0.05 permil in solution 10, 0.1 permil in solution 3, and 0.05 permil in solution 5. The uncertainty limit for sulfur-34 (line 8b) is 1 permil in all solutions. Units of the uncertainty limits for an isotope must be consistent with units used to define the isotope in **SOLUTION** input and with the units used to define isotope values under the **-phases** identifier (line 4).

Line 9: **-range** [*maximum*]

-range--Identifier that specifies that ranges in mole transfer for each phase in each model should be calculated. The range in mole transfer for a phase is the minimum and maximum mole transfers that can be attained for a given inverse model by varying element concentrations within their uncertainty limits. The calculation of these ranges is time consuming, but provides valuable information. In the interest of expediency, it is suggested that models are first identified without using the **-range** identifier and the mole transfers checked for plausibility and geochemical consistency with any additional information such as saturation indices, isotopic compositions, and mineral textures; then the calculation is rerun with the **-range** identifier. Any phase with the **force** option will be included for each range calculation even if the inverse model does not contain this phase. Optionally, **range**, **ranges**, or **-r[anges]**.

maximum--The maximum value for the range is calculated by minimizing the difference between the value of *maximum* and the calculated mole transfer of the phase or the solution fraction. The minimum value of the range is calculated by minimizing the difference between the negative of the value of *maximum* and the calculated mole transfer of the phase or the solution fraction. In some evaporation problems, the solution fraction could be greater than 1000 (over 1000-fold evaporative concentration). In these problems, the default value is not large enough and a larger value of maximum should be entered. Default is 1000.

Line 10: **-minimal**

-minimal--Identifier that specifies that models be reduced to the minimum number of phases that can satisfy all of the constraints within the specified uncertainty limits. Note that two minimal models may have different numbers of phases; minimal models imply that no model with any proper subset of phases and solutions could be found. The **-minimal** identifier minimizes the number of calculations that will be performed and produces the models that contain the most essential geochemical reactions. However, models that are not minimal may also be of interest, so the use of this option is left to the discretion of the user. In the interest of expediency, it is suggested that models are first identified using the **-minimal** identifier, checked for plausibility and geochemical consistency, and then rerun without the **-minimal** identifier. Optionally, **minimal, minimum, -m[inimal]**, or **-m[inimum]**.

Line 11: **-tolerance** *tol*

-tolerance--Identifier that indicates a tolerance for the optimizing solver is to be given. Optionally, **tolerance** or **-t[olerance]**.

tol--Tolerance used by the optimizing solver. The value of *tol* should be greater than the greatest calculated mole transfer or solution fraction multiplied by 1e-15. The default value is adequate unless very large mole transfers (greater than 1000 moles) or solution fractions (greater than 1000-fold evaporative concentration) occur. In these cases, a larger value of *tol* may be needed. Essentially, a value less than *tol* is treated as zero. Thus, the value of *tol* should not be too large or significantly different concentrations will be treated as equal. Uncertainty limits less than *tol* are assumed to be zero. Default is approximately 1e-10 for default compilation, but may be smaller if the program is compiled using long double precision.

Line 12: **-force_solutions** *list of* (*True* or *False*)

-force_solutions--Identifier that indicates one or more solutions will be forced to be included in all range calculations. Optionally, **force_solution, force_solutions**, or **-force_[solutions]**.

list of (*True* or *False*)--**True** values include initial solutions in all range calculations. It is possible to input a **True** or **False** value for each initial solution used in inverse modeling. If fewer values are entered than the number of initial solutions (**-solutions** identifier), then the final value in the list is used for the remaining initial solutions. Thus, if only one **True** or **False** value is entered, it is used for all initial solutions. In the example data block (line 12), solution 10 will be included in all range calculations for all models; even if a model does not include solution 10 (mixing fraction of zero), the range calculation will allow for nonzero mixing fractions of solution 10 in calculating the minimum and maximum mole transfers of phases. Solutions 3 and 5 will be included in range calculations only for models that have a nonzero mixing fractions for these solutions.

Line 13: **-uncertainty_water** *moles*

-uncertainty_water--Identifier for uncertainty term in the water-balance equation. For completeness in the formulation of inverse modeling, an uncertainty term can be added to the water balance equation. The sum of the moles of water derived from each initial solution must balance the moles of water in the final solution plus or minus *moles* of water. Optionally, **uncertainty_water, u_water, -uncertainty_[water]**, or **-u_[water]**.

moles--Uncertainty term for the water-balance equation. Default is 0.0.

Line 14: **-mineral_water** [(*True* or *False*)]

-mineral_water--Identifier to include or exclude water derived from minerals in the water-balance equation. Normally, water from minerals should be included in the water-balance equation. Sometimes unreasonable models are generated that create all the water in solution by dissolution and precipitation of minerals. Setting **-mineral_water** to **False** removes the terms for water derived from minerals from the water-balance equation, which eliminates these unreasonable models. However, removing these terms may introduce errors in some models by ignoring water derived from minerals (for example water from dissolution of gypsum) that should be considered in the water-balance equation. Optionally, **mineral_water** or **-mine[ral_water]**.

(*True* or *False*)--**True** includes terms for water derived from minerals in the water-balance equation, **False** excludes these terms from the equation. Default is **True**.

Notes

Writing of inverse models to the output file can be enabled or disabled with the **-inverse** identifier in the **PRINT** data block. Inverse models can be written to the selected-output file by including the **-inverse** identifier in the **SELECTED_OUTPUT** data block. For each model that is found the following values are written to the selected-output file: (1) the sum of residuals, sum of each residual divided by its uncertainty limit, and the maximum fractional error, (2) for each solution--the mixing fraction, minimum mixing fraction, and maximum mixing fraction, and (3) for each phase in the list of phases (**-phase** identifier)--the mole transfer, minimum mole transfer, and maximum mole transfer. Mixing fractions and mole transfers are zero for solutions and phases not included in the model. Minimum and maximum values are 0.0 unless the **-range** calculation is performed. The result of printing to the selected-output file is columns of numbers, where each row represents a mole-balance model.

The numerical method for inverse modeling *requires* consideration of the uncertainties related to aqueous concentrations. Uncertainties related to mineral compositions may be equally important, but are not automatically considered. To consider uncertainties in mineral compositions, it is possible to include two (or more) phases (under **-phases** identifier and definitions in **PHASES** data block) that represent end-member compositions for minerals. The inverse modeling calculation will attempt to find models considering the entire range of mineral composition. Usually, each model that is found will include only one or the other of the end-members, but any mixture of inverse models, which in this case would represent mixtures of the end members, is also a valid inverse model.

The possibility of evaporation or dilution can be included in inverse modeling by including water as one of the phases under the **-phases** identifier [**H2O(g)** for databases distributed with program]. The mole transfer of this phase will affect only the water-balance equation. If the mole transfer is positive, dilution is simulated; if negative, evaporation is simulated (see example 17 in "Examples").

If **-uncertainty** is not included, a default uncertainty limit of 0.05 (5 percent) is used for elements and 0.05 for pH. Default uncertainty limits, specified by **-uncertainty**, will almost always be specified as positive numbers, indicating fractional uncertainty limits. A default uncertainty limit specified by a negative number, indicating a fixed molal uncertainty limit for all elements in solution, is usually not reasonable because of wide ranges in concentrations among elements present in solution.

No mole-balance equation is used for pH and the uncertainty limit in pH only affects the mole balance on alkalinity. Alkalinity is assumed to co-vary with pH and carbon and an equation relating the uncertainty term for alkalinity and the uncertainty terms for pH and carbon is included in the inverse model (see "Equations and Numerical Methods for Inverse Modeling").

All phase names and phase stoichiometries must be defined through **PHASES** or **EXCHANGE_SPECIES** input. Line 4c and 4d are included to allow ion-exchange reactions in the inverse model; exchange species with the names CaX_2 and NaX are among the exchange species defined in the default database and are thus available for use in inverse modeling. In the example data block and in the examples problems (16, 17, and 18), the composition of the phases is assumed to be relatively simple. In real systems, the composition of reactive phases--for example pyroxenes, amphiboles, or alumino-silicate glasses--may be complex. Application of inverse modeling in these systems will require knowledge of specific mineral compositions or appropriate simplification of the mineral stoichiometries.

By default, mole-balance equations for every element that occurs in the phases listed in **-phases** input are included in the inverse-modeling formulation. If an element is redox active, then mole-balance equations for all valence states of that element are included. The **-balances** identifier is necessary to define (1) uncertainty limits for pH, elements, or element valence states that are different from the default uncertainty limits or (2) mole-balance equations for elements not included in the phases. Mole-balance equations for alkalinity and electrons are always included in the inverse model. In some solutions, such as pure water or pure sodium chloride solutions, the alkalinity may be small (less than 1e-7) in both initial and final solutions. In this case, it may be necessary to use large (relative to 1e-7 equivalents) uncertainty limits (+1.0 or -1e-6) to obtain a mole balance on alkalinity. For most natural waters, alkalinity will not be small in both solutions and special handling of the alkalinity uncertainty will not be necessary (note alkalinity is a negative number in acid solutions). Uncertainty limits for electrons are never used because it is always assumed that no free electrons exist in an aqueous solution.

If isotope mole balances are used, then (1) isotopic values for the aqueous phases must be defined through the **SOLUTION** data block, (2) the **-isotopes** identifier must be used in the **INVERSE_MODELING** data block to specify the isotopes for which mole balances are desired and, optionally, the uncertainty limits in isotopic values associated with each solution, and (3) for each phase listed below the **-phases** identifier of the **INVERSE_MODELING** data block, isotopic values and uncertainty limits must be defined for each isotope that is contained in the phase. In addition, each phase that contains isotopes must be constrained either to dissolve or to precipitate. Default uncertainty limits for isotopes are given in table 5.

The options **-minimal** and **-range** affect the speed of the calculations. The fastest calculation is one that includes the **-minimal** identifier and does not include **-range**. The slowest calculation is one that does not include **-minimal** and does include **-range**.

The **force** option for a phase in **-phases** and the **-force_solutions** identifier affects only the range calculation; it does not affect the number of models that are found. When the **-range** identifier is specified and a model is found by the numerical method, then the model is augmented by any phase for which **force** is specified and by any solution for which **-force_solutions** is **true**; the range calculation is performed with the augmented model. The effect of these options is to calculate wider ranges for mole transfers for some models. If every phase and every solution were forced to be in the range calculation, then the results of the range calculation would be the same for

Table 5.--*Default uncertainty limits for isotopes*

Isotope	Default uncertainty limit
^{13}C	1 permil PDB
$^{13}C(4)$	1 permil PDB
$^{13}C(-4)$	5 permil PDB
^{34}S	1 permil CDT
$^{34}S(5)$	1 permil CDT
$^{34}S(-2)$	5 permil CDT
^{2}H	1 permil VSMOW
^{18}O	0.1 permil VSMOW
^{87}Sr	0.01 ratio

every model and the results would be the maximum possible ranges of mole transfer for any models that could be derived from the given set of solutions and phases.

Example problems

The keyword **INVERSE_MODELING** is used in example problems 16, 17, and 18.

Related keywords

EXCHANGE_SPECIES, PHASES, PRINT, SELECTED_OUTPUT, SOLUTION, and **SAVE.**

KINETICS

This keyword data block is used to identify kinetic reactions and specify reaction parameters for batch-reaction, and transport calculations. Mathematical expressions for the rates of the kinetic reactions are defined with the **RATES** data block. The rate equations are integrated over a time step by a Runge-Kutta method that estimates the error of the integration and uses appropriate time subintervals to maintain the errors within specified tolerances for each time interval.

Example data block 1

```
Line 0:  KINETICS 1 Define 3 explicit time steps
Line 1a: Pyrite
Line 2a:     -formula FeS2 1.0 FeAs2 0.001
Line 3a:     -m       1e-3
Line 4a:     -m0      1e-3
Line 5a:     -parms   3.0   0.67   .5   -0.11
Line 6a:     -tol     1e-9
Line 1b: Calcite
Line 3b:     -m       7.e-4
Line 4b:     -m0      7.e-4
Line 5b:     -parms   5.0      0.3
Line 6b:     -tol     1.e-8
Line 1c: Organic_C
Line 2c:     -formula CH2O(NH3)0.1 0.5
Line 3c:     -m       5.e-3
Line 4c:     -m0      5.e-3
Line 6c:     -tol     1.e-8
Line 7:  -steps       100 200 300  # seconds
Line 8:  -step_divide 100
Line 9:  -runge_kutta 6
```

Explanation 1

Line 0: **KINETICS** [*number*] [*description*]

KINETICS is the keyword for the data block.

number--Positive number to designate the following set of kinetic reactions. A range of numbers may also be given in the form *m-n*, where *m* and *n* are positive integers, *m* is less than *n*, and the two numbers are separated by a hyphen without intervening spaces. Default is 1.

description--Optional comment that describes the kinetic reactions.

Line 1: *rate name*

rate name--Name of a rate expression. The *rate name* and its associated rate expression must be defined within a **RATES** data block, either in the default database file or in the current or previous simulations of the run. The name must be spelled identically to the name used in **RATES** input (except for case).

Line 2: **-formula** *list of formula,* [*stoichiometric coefficient*]

By default, the *rate name* is assumed to be the name of a phase that has been defined in a **PHASES** data block and the formula for that phase is then used for the stoichiometry of the reaction (for example, calcite in case "b" above). However, kinetic reactions are not restricted to mineral phases, any set of elements produced or consumed by the kinetic reaction (relative to the aqueous phase) can be specified through a list of doublets *formula* and *stoichiometric coefficient* (lines 2a and 2c). Optionally, **formula** or **-f[ormula]**.

formula--Chemical formula or the name of a phase to be added by the kinetic reaction. If a chemical formula is used, it must begin with a capital letter and contain element symbols and stoichiometric coefficients (line 2a). A phase name may be entered independent of case. Each *formula* must be a charge-balanced combination of elements. (An exception may be for defining exchangers or surfaces related to kinetic reactants).

stoichiometric coefficient--Defines the mole transfer coefficient for *formula* per mole of reaction progress (evaluated by the rate expression in **RATES**). The product of the coefficient times the moles of reaction progress gives the mole transfer for *formula* relative to the aqueous solution; a negative stoichiometric coefficient and a positive value for reaction progress gives a negative mole transfer, which removes reactants from the aqueous solution. In line 2a, each mole of reaction dissolves 1.0 mole of FeS_2 and 0.001 moles of $FeAs_2$ into the aqueous solution; in line 2c, each mole of reaction (as calculated by the rate expression) adds 0.5 mole of CH_2O and 0.05 mole of NH_3 to the aqueous solution to simulate the degradation of nitrogen-containing organic matter. Default is 1.0.

Line 3: **-m** *moles*

moles--Current moles of reactant. As reactions occur, the *moles* will increase or decrease. Default is equal to *initial moles* if *initial moles* is defined, or 1.0 mol if *initial moles* is not defined. Optionally, **m** or **-m**.

Line 4: **-m0** *initial moles*

initial moles--Initial moles of reactant. This identifier is useful if the rate of reaction is dependent on grain size. Formulations for this dependency often include the ratio of the amount of reactant remaining to the amount of reactant initially present. The quantity *initial moles* does not change as the kinetic reactions proceed. Frequently, the quantity *initial moles* is equal to *moles* at the beginning of a kinetic reaction. Default is equal to *moles* if *moles* is defined, or 1.0 if *moles* is not defined. Optionally, **m0** or **-m0**

Line 5: **-parms** *list of parameters*

list of parameters--A list of numbers may be entered that can be used in the rate expressions, for example constants, exponents, or half saturation constants. In the rate expression defined with the **RATES** keyword, these numbers are available to the Basic interpreter in the array *PARM*; *PARM(1)* is the first number entered, *PARM(2)* the second, and so on. Optionally, **parms**, **-p[arms]**, **parameters**, or **-p[arameters]**.

Line 6: **-tol** *tolerance*

tolerance--Tolerance for integration procedure (moles). For each integration time interval, the difference between the fifth-order and the fourth-order integrals of the rate expression must be less than

this tolerance or the time interval is automatically reduced. The value of *tolerance* is related to the concentration differences that are considered significant for the elements in the reaction. Smaller concentration differences that are considered significant require smaller tolerances. Numerical accuracy of the kinetic integration can be tested by decreasing the tolerance to determine if results change significantly. Default is 1e-8. Optionally, **tol** or **-t[ol]**.

Line 7: **-steps** *list of time steps*

list of time steps--Time steps over which to integrate the rate expressions (seconds). The **-steps** identifier is used only during batch-reaction calculations; it is not needed for transport calculations. By default, the list of time steps are considered to be independent times all starting from zero. The example data block would produce results after 100, 200, and 300 seconds of reaction. However, the **INCREMENTAL_REACTIONS** keyword can be used to make the time steps incremental so that the results of the previous time step are the starting point of the new time step. For incremental time steps, the example data block would produce results after 100, 300, and 600 seconds. Default is 1.0 second. Optionally, **steps** or **-s[teps]**.

Line 8: **-step_divide** *step_divide*

step_divide--If *step_divide* is greater than 1.0, the first time interval of each integration is set to *time step / step_divide*; at least two time intervals must be integrated to reach the total time of *time step*--0 to *time step / step_divide* and *time step / step_divide* to *time step*. If *step_divide* is less than 1.0, then *step_divide* is the maximum moles of reaction that can be added during a kinetic integration subinterval. Frequently reaction rates are fast initially, thus requiring small time intervals to produce an accurate integration of the rate expressions. The Runge-Kutta method will adapt to these fast rates when the integration fails the **-tolerance** criterion, but it may require several reductions in the length of the initial time interval for the integration to meet the criterion; *step_divide* > 1 can be used to make the initial time interval of each integration sufficiently small to satisfy the criterion, which may speed the overall calculation time. However, the smaller time interval will apply to all integrations throughout the simulation, even if reaction rates are slow later in the simulation. Using an appropriate *step_divide* < 1 can also cause sufficiently small initial time intervals when rates are fast, but will not require small time intervals later in the simulation if rates are slow; however, the appropriate value for *step_divide* < 1 is not easily known and usually must be found by trial and error. The default maximal reaction is 0.1 moles during a time subinterval. Normally, **-step_divide** is not used unless run times are long and it is apparent that each integration requires several time intervals. The status line, which is printed to the screen, notes the number of integration intervals that fail the **-tolerance** criterion as "bad" and the number of integration intervals that pass the criterion as "OK". Optionally, **step_divide** or **-step_[divide]**.

Line 9: **-runge_kutta** (**1**, **2**, **3**, or **6**)

(**1**, **2**, **3**, or **6**)--Designates the preferred number of time subintervals to use when integrating rates and is related to the order of the integration method. A value of **6** specifies that a 5th order embedded Runge-Kutta method, which requires 6 intermediate rate evaluations, will be used for all integrations. For values of **1**, **2**, or **3**, the program will try to limit the rate evaluations to this number.

If the **-tolerance** criterion is not satisfied among the evaluations or over the full integration interval, the method will automatically revert to the Runge-Kutta method of order 5. A value of **6** will exclusively use the 5th order method. Values of **1** or **2** are mainly expedient when it is known that the rate is nearly constant in time. Default is **3**. Optionally, **rk**, **-r[k]**, **runge_kutta**, or **-r[unge_kutta]**.

Example data block 2

```
Line 0:  KINETICS 1 Define 3 equal time steps
Line 1a: Calcite
Line 3a:      -m        7.e-4
Line 5a:      -parms    5       0.3
Line 7:  -steps         300 in 3 steps # seconds
```

Explanation 2

Line 0: **KINETICS** [*number*] [*description*]

Same as example data block 1.

Line 1: *rate name*

Same as example data block 1.

Line 3: **-m** *moles*

Same as example data block 1.

Line 5: **-parms** *list of parameters*

Same as example data block 1.

Line 7: **-steps** *total time* [**in** *steps*]

total time--Total time over which to integrate kinetic reactions, in seconds. The total time may be divided into a number of calculations given by *steps*. The **-steps** identifier is used only in batch-reaction calculations; it is not needed for transport calculations. Default is 1.0 second. Optionally, **steps** or **-s[teps]**.

in *steps*--"**in**" indicates that the *total time* will be divided into *steps* number of steps.

INCREMENTAL_REACTIONS has no effect on the output for example data block 2, results will be printed after 100, 200, and 300 seconds of reaction. However, **INCREMENTAL_REACTIONS** does affect the computational method. If **INCREMENTAL_REACTIONS** is **false** the reactions will be integrated over the time intervals from 0 to 100, 0 to 200, and 0 to 300 seconds. If **INCREMENTAL_REACTIONS** is **true** the reactions will be integrated over the time intervals from 0 to 100, 100 to 200, and 200 to 300 seconds.

Notes

Both **KINETICS** and **REACTION** data blocks are used to model irreversible reactions. **REACTION** can only be used to define specified amounts of stoichiometric reactions; essentially the rates of the reactions are constant. **KINETICS** is used to define truly kinetic reactions. To use **KINETICS**, a mathematical rate expression based on the solution composition must be defined and this expression is used to calculate the rate of reaction at any

point in time. The **RATES** data block is used to define a set of general rate expressions that may apply over the entire modeling domain. The **KINETICS** data block is used to identify the subset of general rate expressions that apply to a given batch-reaction or to specified cells of transport calculations. The data block also is used to define specific parameters for the rate expression, such as the moles of reactant initially present in a cell, spatially varying coefficients, or cell-specific exponents for the rate equation. In advective (**ADVECTION** data block) and advective-dispersive transport (**TRANSPORT** data block) calculations, the number(s) assigned with the **KINETICS** keyword defines the cell(s) to which the kinetic reactions apply.

For a batch-reaction calculation, the number of reaction steps is the maximum number of steps defined in any of the following keyword data blocks: **KINETICS**, **REACTION**, and **REACTION_TEMPERATURE**. When the maximum number of steps is greater than the number of steps defined in **KINETICS**, then if **INCREMENTAL_REACTIONS** is false (cumulative reaction steps), the reactions are integrated for the time specified by the final time step for each of the additional steps; if **INCREMENTAL_REACTIONS** is true (incremental reaction steps), kinetic reactions are not included in the additional steps.

Example problems

The keyword **KINETICS** is used in example problems 6C, 9 and 15.

Related keywords

ADVECTION, **PHASES**, **RATES**, **REACTION**, and **TRANSPORT**.

KNOBS

This keyword data block is used to redefine parameters that affect convergence of the numerical method during speciation, batch-reaction, and transport calculations. It also provides the capability to produce long, uninterpretable output files. Hopefully, this data block is seldom used.

Example data block

```
Line 0:  KNOBS
Line 1:       -iterations                150
Line 2:       -convergence_tolerance     1e-8
Line 3:       -tolerance                 1e-14
Line 4:       -step_size                 10.
Line 5:       -pe_step_size              5.
Line 6:       -diagonal_scale            TRUE
Line 7:       -debug_diffuse_layer       TRUE
Line 8:       -debug_inverse             TRUE
Line 9:       -debug_model               TRUE
Line 10:      -debug_prep                TRUE
Line 11:      -debug_set                 TRUE
Line 12:      -logfile                   TRUE
```

Explanation

Line 0: **KNOBS**

KNOBS is the keyword for the data block. Optionally, **DEBUG**.

Line 1: **-iterations** *iterations*

-iterations--Allows changing the maximum number of iterations. Optionally, **iterations**, or **-i[terations]**.

iterations--Positive integer limiting the maximum number of iterations used to solve the set of algebraic equations for a single calculation. Values greater than 200 are not usually effective. Default is 100.

Line 2: **-convergence_tolerance** *convergence_tolerance*

-convergence_tolerance--Changes the convergence criterion used to determine when the algebraic equations have been solved. For an element mole-balance equation, convergence is satisfied when mole balance is within *convergence_tolerance* times the total moles of the element ($convergence\_tolerance \cdot T_m$). When the **-high_precision** identifier of **SELECTED_OUTPUT** is used, the convergence criterion is set to the smaller of *convergence_tolerance* and 1e-12. Default is 1e-8. Optionally, **convergence_tolerance**, or **-c[onvergence_tolerance]**.

Line 3: **-tolerance** *tolerance*

-tolerance--Allows changing the tolerance used by the optimization solver (subroutine C11) to determine numbers equal to zero. This is <u>not</u> the convergence criterion used to determine when the algebraic equations have been solved. Optionally, **tolerance**, or **-t[olerance]**.

tolerance--Positive, decimal number used by the optimization solver (subroutine c11). All numbers smaller than this number are treated as zero. This number should approach the value of the least

significant decimal digit that can be interpreted by the computer. The value of tolerance should be on the order of 1e-12 to 1e-15 for most computers and most simulations. Default is 1e-15 (or possibly smaller if the program is compiled with long double precision).

Line 4: **-step_size** *step_size*

-step_size--Allows changing the maximum step size. Optionally, **step_size**, or **-s[tep_size]**.

step_size--Positive, decimal number limiting the maximum, multiplicative change in the activity of an aqueous master species on each iteration. Default is 100, that is, activities of master species may change by up to 2 orders of magnitude in a single iteration.

Line 5: **-pe_step_size** *pe_step_size*

-pe_step_size--Allows changing the maximum step size for the activity of the electron. Optionally, **pe_step_size**, or **-p[e_step_size]**.

pe_step_size--Positive, decimal number limiting the maximum, multiplicative change in the conventional activity of electrons on each iteration. Normally, *pe_step_size* should be smaller than the *step_size*, because redox species are particularly sensitive to changes in pe. Default is 10, that is, a_{e^-} may change by up to 1 order of magnitude in a single iteration or pe may change by up to 1 unit.

Line 6: **-diagonal_scale** [(*True* or *False*)]

-diagonal_scale--Allows changing the default method for scaling equations. Optionally, **diagonal_scale**, or **-d[iagonal_scale]**.

(*True* or *False*)--A value of **true** (optionally, **t[rue]**) indicates the alternative scaling method is to be used; **false** (optionally, **f[alse]**) indicates the alternative scaling method will not be used. If neither **true** nor **false** are entered, **true** is assumed. At the beginning of the run, the value is set to **false**. Invoking this alternative method of scaling causes any mole-balance equations with the diagonal element (approximately the total concentration of the element or element valence state in solution) less than 1e-11 to be scaled by the factor 1e-11/(diagonal element).

Line 7: **-debug_diffuse_layer** [(*True* or *False*)]

-debug_diffuse_layer--Includes debugging prints for diffuse layer calculations. This identifier applies only when **-diffuse_layer** is used in the **SURFACE** data block. Optionally, **debug_diffuse_layer** or **-debug_d[iffuse_layer]**.

(*True* or *False*)--A value of **true** (optionally, **t[rue]**) indicates the debugging information will be included in the output file; **false** (optionally, **f[alse]**) indicates debugging information will not be printed. If neither **true** nor **false** is entered, a value of **true** is assumed. At the start of the program, the default value is **false**. If this option is set to **true**, values of the *g* function--the surface excess--are printed for each value of charge for aqueous species, the charge(s) for which the value of *g* has not converged are printed, and the number of iterations needed for the integration, by which *g* values are calculated, are printed

Line 8: **-debug_inverse** [(*True* or *False*)]

-debug_inverse--Includes debugging prints for subroutines called by subroutine *inverse_models*. Optionally, **debug_inverse** or **-debug_i[nverse]**.

(*True* or *False*)--A value of **true** (optionally, **t[rue]**) indicates the debugging information will be included in the output file; **false** (optionally, **f[alse]**) indicates debugging information will not be printed. If neither **true** nor **false** is entered, a value of **true** is assumed. At the start of the program, the default value is **false**. If this option is set to **true**, a large amount of information about the process of finding inverse models is printed. The program will print the following for each set of equations and inequalities that are attempted to be solved by the optimizing solver: a list of the unknowns, a list of the equations, the array that is to be solved, any nonnegativity or nonpositivity constraints on the unknowns, the solution vector, and the residual vector for the linear equations and inequality constraints. The printout is very long and very tedious.

Line 9: **-debug_model** [(*True* or *False*)]

-debug_model--Includes debugging prints for subroutines called by subroutine *model*. Optionally, **debug_model** or **-debug_m[odel]**.

(*True* or *False*)--A value of **true** (optionally, **t[rue]**) indicates the debugging information will be included in the output file; **false** (optionally, **f[alse]**) indicates debugging information will not be printed. If neither **true** nor **false** is entered, a value of **true** is assumed. At the start of the program, the default value is **false**. If this option is set to **true**, a large amount of information about the Newton-Raphson equations is printed. The program will print some or all of the following at each iteration: the array that is solved, the solution vector calculated by the solver, the residuals of the linear equations and inequality constraints, the values of all of the master unknowns and their change, the moles of each pure phase and phase mole transfers, the moles of each element in the system minus the amount in pure phases and the change in this quantity. The printout is very long and very tedious. If the numerical method does not converge in *iterations*-1 iterations (default is after 99 iterations), this printout is automatically begun and sent to the log file *phreeqc.log*.

Line 10: **-debug_prep** [(*True* or *False*)]

-debug_prep--Includes debugging prints for subroutine *prep*. Optionally, **debug_prep** or **-debug_p[rep]**.

(*True* or *False*)--A value of **true** (optionally, **t[rue]**) indicates the debugging information will be included in the output file; **false** (optionally, **f[alse]**) indicates debugging information will not be printed. If neither **true** nor **false** is entered, a value of **true** is assumed. At the start of the program, the default value is **false**. If this option is set to **true**, the chemical equation and log K for each species and phase, as rewritten for the current calculation, are written to the output file. The printout is long and tedious.

Line 11: **-debug_set** [(*True* or *False*)]

-debug_set--Includes debugging prints for subroutines called by subroutine *set*. Optionally, **debug_set** or **-debug_s[et]**.

(*True* or *False*)--A value of **true** (optionally, **t[rue]**) indicates the debugging information will be included in the output file; **false** (optionally, **f[alse]**) indicates debugging information will not be printed. If neither **true** nor **false** is entered, a value of **true** is assumed. At the start of the program, the default value is **false**. If this option is set to **true**, the initial revisions of the master unknowns (see equation 84), which occur in subroutine set, are printed for each element or element valence

state that fails the initial convergence criteria. The initial revisions occur before the Newton-Raphson method is invoked and attempt to provide good estimates of the master unknowns to the Newton-Raphson method. The printout is tedious.

Line 12: **-logfile** [(*True* or *False*)]

-logfile--Prints information to a file named *phreeqc.log*. Optionally, **logfile** or **-l[ogfile]**.

(*True* or *False*)--A value of **true** (optionally, **t[rue]**) indicates information will be written to the log file, *phreeqc.log*; **false** (optionally, **f[alse]**) indicates information will not be written. If neither **true** nor **false** is entered, a value of **true** is assumed. At the start of the program, the default value is **false**. If this option is set to **true**, information about each calculation will be written to the log file. The information includes number of iterations in revising the initial estimates of the master unknowns, the number of Newton-Raphson iterations, and the iteration at which any infeasible solution was encountered while solving the system of nonlinear equations. (An infeasible solution occurs if no solution to the equality and inequality constraints can be found.) At each iteration, the identity of any species that exceeds 30 mol (an unreasonably large number) is written to the log file and noted as an "overflow". Any basis switches are noted in the log file. The information about infeasible solutions and overflows can be useful for altering other parameters defined through the **KNOBS** data block, as described below.

Notes

Convergence problems are less frequent with PHREEQC than with PHREEQE; however, they may still occur. The main causes of nonconvergence appear to be (1) calculation of very large molalities in intermediate iterations (2) accumulation of roundoff errors in simulations involving very small concentrations of elements in solution, and (3) loss of precision in problems with no redox buffering. The first cause can be identified by "overflow" messages at iteration 1 or greater that appear in the file *phreeqc.log* (see **-logfile** above). This problem can usually be eliminated by decreasing the maximum allowable step sizes from the default values. The second and third causes of nonconvergence can be identified by messages in *phreeqc.log* that indicate "infeasible solutions". The remedy to these problems is an ongoing investigation, but altering **-tolerance** or **-diagonal_scaling** sometimes fixes the problem, and it should be noted that the program attempts several combinations of these parameters automatically before terminating the calculations. Additional iterations (**-iterations**) beyond 200 usually do not solve nonconvergence problems. A trick that is sometimes helpful with nonconvergence is to include the following fictitious aqueous species that has a concentration of about 1e-9 and produces terms in the charge-, hydrogen-, and oxygen-balance equations of a magnitude great enough for the solver to solve the equations:

```
SOLUTION_SPECIES
H2O + 0.01e- = H2O-0.01
     log_k   -9.0
```

If the numerical method does not converge with the original set of convergence parameters (either default or user specified), six additional sets of parameters are tried automatically to obtain convergence: (1) *iterations* is doubled and smaller values for *step_size* and *pe_step_size* are used; (2) *iterations* is doubled and the value of *diagonal_scale* is switched from false to true or from true to false; (3) *iterations* is doubled and *tol* is decreased by

a factor of 10.0; (4) *iterations* is doubled and *tol* is increased by a factor of 10.0; (5) *iterations* is doubled, *diagonal_scale* is switched, and *tol* is decreased by a factor of 10.0;and (6) *iterations* is doubled and the minimum scaled diagonal is increased by a factor of 10.0.

Example problems

The keyword **KNOBS** is not used in the example problems.

MIX

This keyword data block is used if two or more aqueous solutions are to be mixed together. Normally, the mixing occurs as part of the batch-reaction calculation, but mixing may be applied during advection calculations also.

Example data block

```
Line 0:  MIX 2 Mixing solutions 5, 6, and 7.
Line 1a:       5      1.1
Line 1b:       6      0.5
Line 1c:       7      0.3
```

Explanation

Line 0: **MIX** [*number*] [*description*]

MIX is the keyword for the data block.

number--Positive number to designate the following mixing parameters. Default is 1.

description--Optional comment that describes the mixture.

Line 1: *solution number, mixing fraction*

solution number--Defines a solution to be part of the mixture.

mixing fraction--Decimal number which is multiplied times the moles of each element in the specified solution, to be summed with any other solutions included in the mixture. Mixing fractions may be greater than 1.0.

Notes

In mixing, each solution is multiplied by its mixing fraction and a new solution is calculated by summing over all of the fractional solutions. In the example data block, if the moles of sodium in solutions 5, 6, and 7 were 0.1, 0.2, and 0.3, the moles of sodium in the mixture would be $0.1 \times 1.1 + 0.2 \times 0.5 + 0.3 \times 0.3 = 0.3$. The moles of all elements are multiplied by the solution's mixing fraction, including hydrogen and oxygen. Thus, the mass of water is effectively multiplied by the same fraction. In the example data block, if all solutions have 1 kg of water, the total mass of water in the mixture is $1.1 + 0.5 + 0.3 = 1.9$ kg and the concentration of sodium would be approximately 0.16 mol/kgw (0.3/1.9). The charge imbalance of each solution is multiplied by the mixing fraction and all the imbalances are then summed to calculate the charge imbalance of the mixture. The temperature of the mixture is approximated by multiplying each solution temperature by its mixing fraction, summing these numbers, and dividing by the sum of the mixing fractions. Other intensive properties of the mixture are calculated in the same way as temperature. This approach for calculating the temperature of mixtures is an approximation because enthalpies of reaction are ignored. For example, heat generated by mixing a strong acid with a strong base is not considered.

This formulation of mixing can be used to approximate constant volume processes if the sum of the mixing fractions is 1.0 and all of the solutions have the same mass of water. The calculations are only approximate in terms of mixing volumes because the summation is made in terms of moles (or mass) and no consideration is given to

the partial molar volumes of solutes. Similarly, the formulation for mixing can approximate processes with varying volume, for example, a titration.

When multiple batch-reaction steps are defined in **KINETICS**, **REACTION**, or **REACTION_TEMPERATURE**, and if **INCREMENTAL_REACTIONS** is false (cumulative reaction steps), then each batch-reaction step uses the same mixing factors; if **INCREMENTAL_REACTIONS** is true (incremental reaction steps), then the mixing fractions are applied during the first batch-reaction step only.

Example problems

The keyword **MIX** is used in example problems 3, 4, and 13.

Related keywords

INCREMENTAL_REACTIONS, **SOLUTION**, **SAVE solution**, **USE solution**, and **USE mix.**

PHASES

This keyword data block is used to define a name, chemical reaction, log *K*, and temperature dependence of log *K* for each gas component and mineral that is used for speciation, batch-reaction, transport, or inverse-modeling calculations. Normally, this data block is included in the database file and only additions and modifications are included in the input file.

Example data block

```
Line 0:  PHASES
Line 1a: Gypsum
Line 2a:       CaSO4:2H2O = Ca+2 + SO4-2 + 2H2O
Line 3a:       log_k      -4.58
Line 4a:       delta_h    -0.109
Line 5:        -analytical_expression 68.2401 0.0 -3221.51  -25.0627  0.0
Line 1b: O2(g)
Line 2b:       O2 = O2
Line 3b:       log_k      -2.96
Line 4b:       delta_h    1.844
```

Explanation

Line 0: **PHASES**

Keyword for the data block. No other data are input on the keyword line.

Line 1: *Phase name*

phase name--Alphanumeric name of phase, no spaces are allowed.

Line 2: *Dissolution reaction*

Dissolution reaction for phase to aqueous species. Any aqueous species, including e^-, may be used in the dissolution reaction. The chemical formula for the defined phase must be the first chemical formula on the left-hand side of the equation. The dissolution reaction must precede any identifiers related to the phase. The stoichiometric coefficient for the phase in the chemical reaction must be 1.0.

Line 3: **log_k** *log K*

log_k--Identifier for log *K* at 25°C. Optionally, **-log_k**, **logk**, **-l[og_k]**, or **-l[ogk]**.

log K--Log *K* at 25°C for the reaction. Default is 0.0.

Line 4: **delta_h** *enthalpy,* [*units*]

delta_h--Identifier for enthalpy of reaction at 25°C. Optionally, **-delta_h**, **deltah**, **-d[elta_h]**, or **-d[eltah]**.

enthalpy--Enthalpy of reaction at 25°C for the reaction. Default is 0.0.

units--Units may be calories, kilocalories, joules, or kilojoules per mole. Only the energy unit is needed (per mole is implied) and abbreviations of these units are acceptable. Explicit definition of units for all enthalpy values is recommended. The enthalpy of reaction is used in the van't Hoff equation to determine the temperature dependence of the equilibrium constant. Internally, all enthalpy calculations are performed in the units of kilojoules per mole. Default units are kilojoules per mole.

Line 5: **-analytical_expression** A_1, A_2, A_3, A_4, A_5

-analytical_expression--Identifier for coefficients for an analytical expression for the temperature dependence of log K. If defined, the analytical expression takes precedence over the van't Hoff equation to determine the temperature dependence of the equilibrium constant. Optionally, **analytical_expression, a_e, ae, -a[nalytical_expression], -a[_e], -a[e]**.

A_1, A_2, A_3, A_4, A_5--Five values defining log K as a function of temperature in the expression

$$\log_{10}K = A_1 + A_2T + \frac{A_3}{T} + A_4\log_{10}T + \frac{A_5}{T^2}, \text{ where } T \text{ is in Kelvin.}$$

Notes

The set of lines 1 and 2 must be entered in order, either line 3 (**log_k**) or 5 (**-analytical_expression**) must be entered for each phase. The analytical expression (**-analytical_expression**) takes precedence over the van't Hoff equation (**delta_H**) to determine the temperature dependence of the equilibrium constant. Lines 3, 4, and 5 may be entered as needed in any order. Additional sets of lines 1 through 5 may be added as necessary to define all minerals and gases. The equations for the phases may be written in terms of any aqueous chemical species, including e^-.

The identifiers **-no_check** can be used to disable checking charge and elemental balances (see **SOLUTION_SPECIES**). The use of **-no_check** is not recommended, except in cases where the phase is only to be used for inverse modeling. Even in this case, equations defining phases should be charge balanced. The identifier can also be used to define the mineral formula for an exchanger with an explicit charge imbalance (see explanation under **EXCHANGE**).

Example problems

The keyword **PHASES** is used in example problems 1, 6, 8, 9, 10, 16, and 18.

Related keywords

EQUILIBRIUM_PHASES, EXCHANGE, INVERSE_MODELING, KINETICS, REACTION, SAVE equilibrium_phases, and **USE equilibrium_phases**.

PRINT

This keyword data block is used to select which results are written to the output file. Fourteen blocks of calculation results may be included or excluded in the output file for each simulation. In addition, this data block allows the following to be enabled or disabled: writing of results to the selected-output file and writing a status line to the screen that monitors the type of calculation being performed.

Example data block

```
Line 0:  PRINT
Line 1:       -reset                  false
Line 2:       -eh                     true
Line 3:       -equilibrium_phases     true
Line 4:       -exchange               true
Line 5:       -gas_phase              true
Line 6:       -headings               true
Line 7:       -inverse                true
Line 8:       -kinetics               true
Line 9:       -other                  true
Line 10:      -saturation_indices     true
Line 11:      -solid_solutions        true
Line 12:      -species                true
Line 13:      -surface                true
Line 14:      -totals                 true
Line 15:      -user_print             true
Line 16:      -selected_output        false
Line 17:      -status                 false
```

Explanation

Line 0: **PRINT**

Keyword for the data block. No other data are input on the keyword line.

Line 1: **-reset** [(*True* or *False*)]

-reset--Changes all print options listed on lines 2 through 15 to **true** or **false**. If used, this identifier should be the first identifier of the data block. Individual print options may follow. Default is **true**. Optionally, **reset** or **-r[eset]**.

(*True* or *False*)--**True** causes all data blocks on lines 2 through 15 to be printed to the output file; **false** causes all these data blocks to be excluded from the output file. Optionally, **t[rue]** or **f[alse]**, case independent.

Line 2: **-eh** [(*True* or *False*)]

-eh--Prints eh values derived from redox couples in initial solution calculations if value is **true**, excludes print if value is **false**. Default is **true**. Optionally, **eh**.

Line 3: **-equilibrium_phases** [(*True* or *False*)]

-equilibrium_phases--Prints composition of the pure-phase assemblage if value is **true**, excludes print if value is **false**. Default is **true**. Optionally, **equilibria**, **equilibrium**, **pure**, **-eq[uilibrium_phases]**, **-eq[uilibria]**, **-p[ure_phases]**, or **-p[ure]**. Note the hyphen is required

to avoid a conflict with the keyword **EQUILIBRIUM_PHASES**; the same is true for the synonym **PURE_PHASES**.

Line 4: **-exchange** [(*True* or *False*)]

-exchange--Prints composition of the exchange assemblage if value is **true**, excludes print if value is **false**. Default is **true**. Optionally, **-ex[change]**. Note the hyphen is required to avoid a conflict with the keyword **EXCHANGE**.

Line 5: **-gas_phase** [(*True* or *False*)]

-gas_phase--Prints composition of the gas phase if value is **true**, excludes print if value is **false**. Default is **true**. Optionally, **-g[as_phase]**. Note the hyphen is required to avoid a conflict with the keyword **GAS_PHASE**.

Line 6: **-headings** [(*True* or *False*)]

-headings--Prints title and headings that identify the beginning of each type of calculation if value is **true**, excludes print if value is **false**. Default is **true**. Optionally, **heading**, **headings**, or **-h[eadings]**.

Line 7: **-inverse_modeling** [(*True* or *False*)]

-inverse_modeling--Prints results of inverse modeling if value is **true**, excludes print if value is **false**. Default is **true**. Optionally, **inverse**, or **-i[nverse_modeling]**. Note the hyphen is required to avoid a conflict with the keyword **INVERSE_MODELING**.

Line 8: **-kinetics** [(*True* or *False*)]

-kinetics--Prints information about kinetic reactants if value is **true**, excludes print if value is **false**. Default is **true**. Optionally, **-k[inetics]**. Note the hyphen is required to avoid a conflict with the keyword **KINETICS**.

Line 9: **-other** [(*True* or *False*)]

-other--Controls all printing to the output file not controlled by any of the other identifiers, including lines that identify the solution or mixture, exchange assemblage, solid-solution assemblage, surface assemblage, pure-phase assemblage, kinetic reaction, and gas phase to be used in each calculation; and description of the stoichiometric reaction. Default is **true**. Optionally, **other**, **-o[ther]**, **use**, or **-u[se]**.

Line 10: **-saturation_indices** [(*True* or *False*)]

-saturation_indices--Prints saturation indices for each phase for which a saturation index can be calculated if value is **true**, excludes print if value is **false**. Default is **true**. Optionally, **-si**, **si**, **saturation_indices**, or **-sa[turation_indices]**.

Line 11: **-solid_solutions** [(*True* or *False*)]

-solid_solutions--Prints compositions of solid solutions if value is **true**, excludes print if value is **false**. Default is **true**. Optionally, **-so[lid_solutions]**. Note the hyphen is required to avoid a conflict with the keyword **SOLID_SOLUTIONS**.

Line 12: **-species** [(*True* or *False*)]

-species--Prints the distribution of aqueous species, including molality, activity, and activity coefficient, if value is **true**, excludes print if value is **false**. Default is **true**. Optionally, **species** or **-sp[ecies]**.

Line 13: **-surface** [(*True* or *False*)]

-surface--Prints composition of the surface assemblage if **true**, excludes print if **false**. Default is **true**. Optionally, **-su[rface]**. Note the hyphen is required to avoid a conflict with the keyword **SURFACE**.

Line 14: **-totals** [(*True* or *False*)]

-totals--Prints the total molalities of elements (or element valence states in initial solutions), pH, pe, temperature, and other solution characteristics if **true**, excludes print if **false**. Note printing of molalities and other properties of all of the aqueous species is controlled by the **-species** identifier. Default is **true**. Optionally, **totals** or **-t[otals]**.

Line 15: **-user_print** [(*True* or *False*)]

-user_print--Controls writing of information defined in **USER_PRINT** to the output file. When set to **false**, information defined in **USER_PRINT** will not be written to the output file. Default is **true**. Optionally, **-user[_print]**. Note the hyphen is required to avoid a conflict with the keyword **USER_PRINT**.

Line 16: **-selected_output** [(*True* or *False*)]

-selected_output--Controls writing of information to the selected-output file. This identifier has no effect unless the **SELECTED_OUTPUT** data block is included in the file. If a **SELECTED_OUTPUT** data block is included and **-selected_output** is **false**, no results are written to the selected-output file. Writing to the selected-output file can be resumed if **-selected_output** is set to **true** in a **PRINT** data block in a subsequent simulation. This print-control option is not affected by **-reset**. Default is **true**. Optionally, **-se[lected_output]**. Note the hyphen is required to avoid a conflict with the keyword **SELECTED_OUTPUT**.

Line 17: **-status** [(*True* or *False*)]

-status--Controls printing of information to the screen that monitors calculations. When set to **true**, a status line is printed to the screen identifying the simulation number and the type of calculation that is currently being processed by the program. When set to **false**, no status line will be printed to the screen. This print-control option is not affected by **-reset**. Default is **true**. Optionally, **status** or **-st[atus]**.

Notes

By default, all print options are set to **true** at the beginning of a run. Once set by the keyword data block **PRINT**, options remain in effect until the end of the run or until changed in another **PRINT** data block.

Unlike most of PHREEQC input, the order in which the identifiers are entered is important when using the **-reset** identifier. Any identifier set before the **-reset** in the data block will be reset when **-reset** is encountered. Thus, **-reset** should be the first identifier in the data block. Using **-reset false** will eliminate all printing to the output file except echoes of the input data set and warning and error messages.

For long **TRANSPORT** and **ADVECTION** calculations with **KINETICS**, printing the status line [**-status true** (default)] may cause a significant increase in run time. This has found to be the case on some Macintosh systems. If printing to the screen is unbuffered, the program must wait for the status line to be written before continuing calculations, which slows overall execution time. In this case, setting -**status false** may speed up run times.

The identifiers **-species** and **-saturation_indices** control the longest output data blocks and are the most likely to be selectively excluded from long computer runs. If transport calculations are made, the output file could become very large unless some or all of the output is excluded though the **PRINT** data block (**-reset false**). Alternatively, the output in transport calculations may be limited by using the **-print_cells** and **-print_frequency** identifiers in the **ADVECTION** and **TRANSPORT** data block. For transport calculations, the **SELECTED_OUTPUT** data block is usually used to produce a compact file of selected results.

Example problems

The keyword **PRINT** is used in example problems 10, 12, 13, 14, and 15.

Related keywords

ADVECTION: **-print_cells** and **-print_frequency**, **SELECTED_OUTPUT**, **TRANSPORT**: **-print_cells** and **-print_frequency**, **USER_PRINT**, and **USER_PUNCH**.

RATES

This keyword data block is used to define mathematical rate expressions for kinetic reactions. General rate formulas are defined in the **RATES** data block and specific parameters for batch-reaction or transport kinetics are defined in **KINETICS** data blocks.

Example data block

```
Line 0: RATES
Line 1a:     Calcite
Line 2a:     -start
Basic: 1   rem M = current number of moles of calcite
Basic: 2   rem M0 = number of moles of calcite initially present
Basic: 3   rem PARM(1) = A/V, cm^2/L
Basic: 4   rem PARM(2) = exponent for M/M0
Basic: 10  si_cc = SI("Calcite")
Basic: 20  if (M <= 0 and si_cc < 0) then goto 200
Basic: 30    k1 = 10^(0.198 - 444.0 / TK )
Basic: 40    k2 = 10^(2.84 - 2177.0 / TK)
Basic: 50    if TC <= 25 then k3 = 10^(-5.86 - 317.0 / TK )
Basic: 60    if TC > 25 then k3  = 10^(-1.1 - 1737.0 / TK )
Basic: 70    t = 1
Basic: 80    if M0 > 0 then t = M/M0
Basic: 90    if t = 0 then t = 1
Basic: 100   area = PARM(1) * (t)^PARM(2)
Basic: 110   rf = k1*ACT("H+")+k2*ACT("CO2")+k3*ACT("H2O")
Basic: 120   rem 1e-3 converts mmol to mol
Basic: 130   rate = area * 1e-3 * rf * (1 - 10^(2/3*si_cc))
Basic: 140   moles = rate * TIME
Basic: 200 SAVE moles
Line 3a:     -end
Line 1b:     Pyrite
Line 2b:     -start
Basic: 1   rem  PARM(1) = log10(A/V, 1/dm)
Basic: 2   rem  PARM(2) = exp for (M/M0)
Basic: 3   rem  PARM(3) = exp for O2
Basic: 4   rem  PARM(4) = exp for H+
Basic: 10  if (M <= 0) then goto 200
Basic: 20  if (SI("Pyrite") >= 0) then goto 200
Basic: 30    lograte = -10.19 + PARM(1) + PARM(2)*LOG10(M/M0)
Basic: 40    lograte = lograte + PARM(3)*LM("O2") + PARM(4)*LM("H+")
Basic: 50    moles = (10^lograte) * TIME
Basic: 60    if (moles > M) then moles = M
Basic: 200 SAVE moles
Line 3b:     -end
```

Explanation

Line 0: **RATES**

RATES is the keyword for the data block. No other data are input on the keyword line.

Line 1: *name of rate expression*

name of rate expression--Alphanumeric character string that identifies the rate expression, no spaces are allowed.

Line 2: **-start**

-start--Identifier marks the beginning of a Basic program by which the moles of reaction for a time subinterval are calculated. Optional.

Basic: *numbered Basic statement*

numbered Basic statement--A valid Basic language statement that must be numbered. The statements are evaluated in numerical order. The sequence of statements must extrapolate the rate of reaction over the time subinterval given by the internally defined variable TIME. The last statement must be "**SAVE** *expression*", where the value of *expression* is the moles of reaction that occur during time subinterval TIME. Statements and functions that are available through the Basic interpreter are listed in tables 8 and 9. Parameters defined in the **KINETICS** data block are also available through the array PARM.

Line 2: **-end**

-end--Identifier marks the end of a Basic program by which the number of moles of a reaction for a time subinterval is calculated. Note the hyphen is required to avoid a conflict with the keyword **END**.

Notes

A Basic interpreter (David Gillespie, Synaptics, Inc., San Jose, CA, written commun., 1997) distributed with the Linux operating system (Free Software Foundation, Inc.) is embedded in PHREEQC. The Basic interpreter is used during the integration of the kinetic reactions to evaluate the moles of reaction progress for a time subinterval. A Basic program for each kinetic reaction must be included in the input or database file. Each program must stand alone with its own set of variables and numbered statement lines. No data is passed between rate programs and there is no conflict using the same variable names or line numbers in separate rate programs. However, it is possible to transfer data among rates with the special Basic statements PUT and GET (see table 8). The programs are used to calculate the instantaneous rate of reaction and extrapolate that rate for a time subinterval given by the variable "TIME" (examples: calcite, line 140; pyrite line 50). TIME is a fraction of the time step that is defined in the **KINETICS** (for batch reactions), **ADVECTION**, or **TRANSPORT** data blocks. The total moles of reaction must be returned to the main program with a SAVE command (line 200 in each example). Note that not the rate, but the moles of reaction are returned, counted positive when the solution concentration of the reactant increases.

The first example estimates the rate of calcite dissolution or precipitation on the basis of a rate expression from Plummer and others (1978) as derived in equations 101 and 106. The forward rate is given by

$$R_f = k_1[H^+] + k_2[CO_{2(aq)}] + k_3[H_2O], \tag{155}$$

where brackets indicate activity and k_1, k_2, and k_3 are functions of temperature (Plummer and others, 1978). In a pure calcite-water system with fixed P_{CO_2}, the overall rate for calcite (forward rate minus backward rate) is approximated by

Table 6.--Description of Basic program for calcite kinetics given in example for **RATES** data block

Line number	Function
1-4	Comments.
10	Calculate calcite saturation index.
20	If undersaturated and no moles of calcite, exit; moles=0 by default.
30-60	Calculate temperature dependence of constants k1, k2, and k3.
70-90	Calculate ratio of current moles of calcite to initial moles of calcite, set ratio to 1 if no moles of calcite are present.
100	Calculate surface area.
110	Calculate forward rate.
130	Calculate overall rate, factor of 1e-3 converts rate to moles from millimoles.
140	Calculate moles of reaction over time interval given by TIME. Note that the multiplication of the rate by TIME must be present in one of the Basic lines.
200	Return moles of reaction for time subinterval with "SAVE". A SAVE statement must always be present in a rate program.

$$R_{Calcite} = r_f\left[1-\left(\frac{IAP}{K_{Calcite}}\right)^{\frac{2}{3}}\right], \tag{156}$$

where $R_{Calcite}$ is mmol cm^{-2} s^{-1}. Equation 156 is implemented in Basic for the first example above. Explanations of the Basic lines for this rate expression are given in table 6.

The second example is for the dissolution of pyrite in the presence of dissolved oxygen from Williamson and Rimstidt (1994):

$$R_{Pyrite} = 10^{-10.19}(O_{2(aq)})^{0.5}(H^+)^{-0.11} \tag{157}$$

where parentheses indicate molality. This rate is based on detailed measurements in solutions of varying compositions and shows a square root dependence on the molality of oxygen and a small dependence on pH. This rate is,

Table 7.--Description of Basic program for pyrite dissolution kinetics given in example for **RATES** data block

Line number	Function
1-4	Comments.
10	Checks that pyrite is still available, otherwise exits with value of moles=0 by default.
20	Checks that the solution is undersaturated (the rate is for dissolution only), otherwise exits with value of moles=0.
30, 40	Calculate log of the rate of pyrite dissolution.
50	Calculate the moles of pyrite dissolution over time interval given by TIME.
60	Limits pyrite dissolution to remaining moles of pyrite.
200	Return moles of reaction for time subinterval with SAVE. A SAVE statement must always be present in a rate program.

Table 8.--Special Basic statements and functions for PHREEQC

Special PHREEQC Statement or Function	Explanation
ACT("HCO3-")	Activity of an aqueous, exchange, or surface species.
ALK	Alkalinity of solution.
CELL_NO	Cell number in **TRANSPORT** or **ADVECTION** calculations
CHARGE_BALANCE	Aqueous charge balance in equivalents.
DIST	Distance to midpoint of cell in **TRANSPORT** calculations, cell number in **ADVECTION** calculations, "-99" in all other calculations.
EQUI("Calcite")	Moles of a phase in the pure-phase (equilibrium-phase) assemblage.
EXISTS(*i1*[, *i2*, ...])	Determines if a value has been stored with a PUT statement for the list of one or more subscripts.The function equals 1 if a value has been stored and 0 if no value has been stored. Values are stored in global storage with PUT and are accessible by any Basic program. See description of PUT for more details.
GAS("CO2(g)")	Moles of a gas component in the gas phase.
GET(*i1*[, *i2*, ...])	Retrieves the value that is identified by the list of one or more subscripts.Value is zero if PUT has not been used to store a value for the set of subscripts. Values stored in global storage with PUT are accessible by any Basic program. See description of PUT for more details.
KIN("CH2O")	Moles of a kinetic reactant.
LA("HCO3-")	Log10 of activity of an aqueous, exchange, or surface species.
LM("HCO3-")	Log10 of molality of an aqueous, exchange, or surface species.
M	Current moles of reactant for which the rate is being calculated (see **KINETICS**).
M0	Initial moles of reactant for which the rate is being calculated (see **KINETICS**).
MISC1("Ca(x)Sr(1-x)SO4")	Mole fraction of component 2 at the beginning of the miscibility gap, returns 1.0 if there is no miscibility gap (see **SOLID_SOLUTIONS**).
MISC2("Ca(x)Sr(1-x)SO4")	Mole fraction of component 2 at the end of the miscibility gap, returns 1.0 if there is no miscibility gap (see **SOLID_SOLUTIONS**).
MOL("HCO3-")	Molality of an aqueous, exchange, or surface species.
MU	Ionic strength of the solution.
PARM(*i*)	Parameter array defined in **KINETICS** data block.
PERCENT_ERROR	Percent charge-balance error [100(cations-|anions|)/(cations + |anions|)].
PRINT	Write to output file.
PUNCH	Write to selected-output file.
PUT(*x*, *i1*[, *i2*, ...])	Saves value of *x* in global storage that is identified by a sequence of one or more subscripts. Value of *x* can be retrieved with GET(*i1*,[, *i2*, ...]) and a set of subscripts can be tested to determine if a value has been stored with EXISTS(*i1*[, *i2*, ...]). PUT may be used in **RATES**, **USER_PRINT**, or **USER_PUNCH** Basic programs to store a value. The value may be retrieved by any of these Basic programs. The value persists until overwritten using a PUT statement with the same set of subscripts, or until the end of the run. For a **KINETICS** data block, the Basic programs for the rate expressions are evaluated in the order in which they are defined in the input file.
RXN	Amount of reaction (moles) as defined in **-steps** in **REACTION** data block for a batch-reaction calculation, otherwise zero.
SAVE	Last statement of Basic program that returns the moles of kinetic reactant, counted positive when the solution concentration of the reactant increases.
SI("Calcite")	Saturation index of a phase, $Log10\left(\frac{IAP}{K}\right)$.
SIM_NO	Simulation number, equals one more than the number of **END** statements before current simulation.

Table 8.--Special Basic statements and functions for PHREEQC

Special PHREEQC Statement or Function	Explanation
SIM_TIME	Time (s) from the beginning of a kinetic batch-reaction or transport calculation.
SR("Calcite")	Saturation ratio of a phase, $\frac{IAP}{K}$.
STEP_NO	Step number in batch-reaction calculations, or shift number in **ADVECTION** and **TRANSPORT** calculations.
S_S("MgCO3")	Current moles of a solid-solution component.
TC	Temperature in Celsius.
TK	Temperature in Kelvin.
TIME	Time interval for which moles of reaction are calculated in rate programs, automatically set in the time-step algorithm of the numerical integration method.
TOT("Fe(2)")	Total molality of element or element redox state. TOT("water") is total mass of water (kg).
TOTAL_TIME	Cumulative time (s) including all advective (for which **-time_step** is defined) and advective-dispersive transport simulations from the beginning of the run or from last **-initial_time** identifier.

applicable only for dissolution in the presence of oxygen and will be incorrect near equilibrium when oxygen is depleted. Explanations of the Basic lines for this rate expression are given in table 7.

Some special statements and functions have been added to the Basic interpreter to allow access to quantities that may be needed in rate expressions. These functions are listed table 8 and the Basic statements that are implemented in the interpreter are listed in table 9. Upper or lower case (case insensitive) may be used for statement, function, and variable names. String variable names must end with the character "$".

The PRINT command in Basic programs is useful for debugging rate expressions. It can be used to write quantities to the output file to check that rates are calculated correctly. However, the PRINT command will write to the output file every time a rate is evaluated, which may be many times per time step. The sequence of information from PRINT statements in **RATES** definitions may be difficult to interpret because of the automatic time-step adjustment of the integration method.

Table 9.--Standard Basic statements and functions

[Character variables in Basic have "$" as the final character of the variable name]

Basic Statements and Functions	Explanation
+, -, *, /	Add, subtract, multiply, and divide.
string1 + *string2*	String concatenation
a ^ *b*	Exponentiation, a^b.
<, >, <=, >=, <>, =, AND, OR, XOR, NOT	Relational and Boolean operators.
ABS(*a*)	Absolute value.
ARCTAN(*a*)	Arctangent function.
ASC(*character*)	Ascii value for *character.*
CHR$(*number*)	Convert Ascii value to character.
COS(*a*)	Cosine function.
DIM *a*(*n*)	Dimension an array

Table 9.--Standard Basic statements and functions

[Character variables in Basic have "$" as the final character of the variable name]

Basic Statements and Functions	Explanation
DATA *list*	List of data.
EXP(*a*)	e^a.
FOR *i* = *n* TO *m* STEP *k* NEXT *i*	"For" loop.
GOTO *line*	Go to line number.
GOSUB *line*	Go to subroutine.
IF (*expr*) THEN *statement* ELSE *statement*	If, then, else statement (on one line; a '\' may be used to concatenate lines).
LEN(*string*)	Number of characters in *string*.
LOG(*a*)	Natural logarithm.
LOG10(*a*)	Base 10 logarithm.
MID$(*string, n*) MID$(*string, n, m*)	Extract characters from position *n* to end of *string*. Extract *m* characters from *string* starting at position *n*.
a MOD *b*	Returns remainder *a* / *b*.
ON *expr* GOTO *line1*, *line2*, ... ON *expr* GOSUB *line1*, *line2*, ...	If the expression's value, rounded to an integer, is *N*, go to the *N*th line number in the list. If *N* is less than one or greater than the number of line numbers listed, execution continues at the next statement after the ON statement.
READ	Read from DATA statement.
REM	At beginning of line, line is a remark with no effect on the calculations.
RESTORE *line*	Set pointer to DATA statement of *line* for subsequent READ.
RETURN	Return from subroutine.
SGN(*a*)	Sign of *a*, +1 or -1.
SIN(*a*)	Sine function.
SQR(*a*)	a^2.
SQRT(*a*)	$\sqrt{a}$.
STR$(*a*)	Convert number to a string.
TAN(*a*)	Tangent function.
VAL(*string*)	Convert string to number.
WHILE (*expression*) WEND	"While" loop.

Example problems

The keyword **RATES** is used in example problems 6C, 9 and 15.

Related keywords

ADVECTION, **KINETICS**, and **TRANSPORT**.

REACTION

This keyword data block is used to define irreversible reactions that transfer specified amounts of elements to or from the aqueous solution during batch-reaction or transport calculations. **REACTION** steps are specified explicitly and do not depend directly on solution composition or time. Use **KINETICS** and **RATES** data blocks instead of the **REACTION** data block to model the rates of irreversible reactions that evolve with time and vary with solution composition.

Example data block 1

```
Line 0:  REACTION 5 Add sodium chloride and calcite to solution.
Line 1a:       NaCl     2.0
Line 1b:       Calcite  0.001
Line 2:        0.25     0.5     0.75     1.0  moles
```

Explanation 1

Line 0: **REACTION** [*number*] [*description*]

REACTION is the keyword for the data block.

number--Positive number to designate the following stoichiometric reaction. A range of numbers may also be given in the form *m-n*, where *m* and *n* are positive integers, *m* is less than *n*, and the two numbers are separated by a hyphen without intervening spaces. Default is 1.

description--Optional comment that describes the stoichiometric reaction.

Line 1: (*phase name* or *formula*), [*relative stoichiometry*]

phase name or *formula*--If a *phase name* is given, the program uses the stoichiometry of that phase as defined by **PHASES** input; otherwise, *formula* is a chemical formula to be used in the stoichiometric reaction. Additional lines can be used to define additional reactants.

relative stoichiometry--Amount of this reactant relative to other reactants, it is a molar ratio between reactants. In the example data block, the reaction contains 2000 times more NaCl than calcite. Default is 1.0.

Line 2: *list of reaction amounts,* [*units*]

list of reaction amounts--A separate calculation will be made for each listed amount. If **INCREMENTAL_REACTIONS** is **false** (default), example data block 1 performs the calculation as follows: the first step adds 0.25 mol of reaction to the initial solution; the second step adds 0.5 mol of reaction to the initial solution; the third 0.75; and the fourth 1.0; each reaction step begins with the same initial solution and adds only the amount of reaction specified. If **INCREMENTAL_REACTIONS** keyword is **true**, the calculations are performed as follows: the first step adds 0.25 mol of reaction and the intermediate results are saved as the starting point for the next step; then 0.5 mol of reaction are added and the intermediate results saved; then 0.75 mol; then 1.0 mol; the total amount of reaction added to the initial solution is 2.5 mol. The total amount of each reactant added at any step in the reaction is the reaction amount times the relative stoichiometric coefficient of the reactant. Additional lines may be used to define all reactant amounts.

units--Units may be moles, millimoles, or micromoles. Units must follow all reaction amounts. Default is moles.

If line 2 is not entered, the default is one step of 1.0 mol.

Example data block 2

```
Line 0: REACTION 5 Add sodium chloride and calcite to reaction solution.
Line 1a:      NaCl        2.0
Line 1b:      Calcite     0.001
Line 2:       1.0 moles in 4 steps
```

Explanation 2

Line 0: **REACTION** [*number*] [*description*]

Same as example data block 1.

Line 1: (*phase name* or *formula*), [*relative stoichiometry*]

Same as example data block 1.

Line 2: *reaction amount* [*units*] [**in** *steps*]

reaction amount--A single reaction amount is entered. This amount of reaction will be added in *steps* steps.

units--Same as example data block 1.

in *steps*--"**in**" indicates that the stoichiometric reaction will be divided into *steps* number of steps. If **INCREMENTAL_REACTIONS** is **false** (default), example data block 2 performs the calculations as follows: the first step adds 0.25 mol of reaction to the initial solution; the second step adds 0.5 mol of reaction to the initial solution; the third 0.75; and the fourth 1.0. If **INCREMENTAL_REACTIONS** keyword is **true**, the calculations are performed as follows: each of the four steps adds 0.25 mol of reaction and the intermediate results are saved as the starting point for the next step.

If line 2 is not entered, the default is one step of 1.0 mol.

Notes

The **REACTION** data block is used to increase or decrease solution concentrations by specified amounts of reaction. The specified reactions are added to or removed from solution without regard to equilibrium, time, or reaction kinetics. Irreversible reactions for which time evolution or concentration-dependent rates are needed must be modeled using the **KINETICS** and **RATES** keywords, however, a kinetic rate expression is needed for this type of modeling.

Example data block 1 with **INCREMENTAL_REACTIONS false** and example data block 2 with **INCREMENTAL_REACTIONS True** or **False** will generate the same results--solutions after a total of 0.25, 0.5, 0.75, and 1.0 mol of reaction have been added. Example data block 1 with **INCREMENTAL_REACTIONS true** generates results after a total of 0.25, 0.75, 1.5, and 2.5 mol of reaction have been added.

If a phase name is used to define the stoichiometry of a reactant, that phase must be defined by **PHASES** input in the database or in the input data file. If negative relative stoichiometries or negative reaction amounts are used, it is possible to remove more of an element than is present in the system, which results in negative concentrations.

Negative concentrations will cause the calculations to fail. It is possible to "evaporate" a solution by removing H_2O or dilute a solution by adding H_2O. If more reaction steps are defined in the **REACTION_TEMPERATURE** or **KINETICS** data blocks than in **REACTION**, then the final reaction amount defined by **REACTION** will be repeated for the additional steps. Suppose only one reaction step of 1.0 mole is specified in a **REACTION** data block and two temperature steps are specified in a **REACTION_TEMPERATURE** data block. If **INCREMENTAL_REACTIONS** is **false**, then the total amount of reaction added by the end of step 1 and 2 is the same, 1.0 mole. However, if **INCREMENTAL_REACTIONS true**, the total amount of reaction added by the end of step 1 will be 1.0 mole and by the end of step two will be 2.0 mole.

Example problems

The keyword **REACTION** is used in example problems 4, 5, 6, 7, and 10.

Related keywords

INCREMENTAL_REACTIONS, KINETICS, PHASES, RATES, and **REACTION_TEMPERATURE.**

REACTION_TEMPERATURE

This keyword data block is used to define temperature during batch-reaction steps. It is necessary to enter this data block if a temperature other than the default temperature is needed for batch-reaction calculations. This data block can also be used to specify the temperature in a cell or range of cells during advective transport calculations (**ADVECTION**) and the initial temperature for a cell or range of cells in advective-dispersive transport calculations (**TRANSPORT**).

Example data block 1

```
Line 0: REACTION_TEMPERATURE 1 Three explicit reaction temperatures.
Line 1:      15.0      25.0      35.0
```

Explanation 1

Line 0: **REACTION_TEMPERATURE** [*number*] [*description*]

REACTION_TEMPERATURE is the keyword for the data block.

number--Positive number to designate the following temperature data. A range of numbers may also be given in the form *m-n*, where *m* and *n* are positive integers, *m* is less than *n*, and the two numbers are separated by a hyphen without intervening spaces. Default is 1.

description--Optional comment that describes the temperature data.

Line 1: *list of temperatures*

list of temperatures--A list of temperatures, in Celsius, that will be applied to batch-reaction calculations. More lines may be used to supply additional temperatures. One batch-reaction calculation will be performed for each listed temperature.

Example data block 2

```
Line 0: REACTION_TEMPERATURE 1 Three implicit reaction temperatures.
Line 1:      15.0      35.0 in 3 steps
```

Explanation 2

Line 0: **REACTION_TEMPERATURE** [*number*] [*description*]

Same as example data block 1.

Line 1: $temp_1$, $temp_2$, **in** *steps*

$temp_1$--Temperature of first reaction step, in Celsius.

$temp_2$--Temperature of final reaction step, in Celsius.

in *steps*--"**in**" indicates that the temperature will be calculated for each of *steps* number of steps. The temperature at each step, *i*, will be calculated by the formula

$$temp_i = temp_1 + \frac{(i-1)}{(steps-1)}(temp_2 - temp_1)$$; if *steps* = 1, then the temperature of the batch reaction will be $temp_1$. Example data block 2 performs exactly the same calculations as example data block 1. If more batch-reaction steps are defined by **REACTION** or **KINETICS** input, the temperature of the additional steps will be $temp_2$.

Notes

If more batch-reaction steps are defined in **REACTION** or **KINETICS** than temperature steps in **REACTION_TEMPERATURE**, then the final temperature will be used for all of the additional batch-reaction steps. **INCREMENTAL_REACTIONS** keyword has no effect on the **REACTION_TEMPERATURE** data block. The default temperature of a reaction step is equal to the temperature of the initial solution or the mixing-fraction-averaged temperature of a mixture. **REACTION_TEMPERATURE** input can be used even if there is no **REACTION** input. The method of calculation of temperature steps using "**in**" is slightly different than that for reaction steps. If *n* temperature steps are defined with "**in** *n*" in a **REACTION_TEMPERATURE** data block, then the temperature of the first reaction step is equal to $temp_1$; temperatures in the remaining steps changes in *n-1* equal increments. If *n* reaction steps are defined with "**in** *n*" in a **REACTION** data block, then the reaction is added in *n* equal increments.

In an advective transport calculation (**ADVECTION**), if **REACTION_TEMPERATURE** *n* is defined (or a range is defined *n-m*), and *n* is less than or equal to the number of cells in the simulation, then the first temperature in the data block of **REACTION_TEMPERATURE** *n* is used as the temperature in cell *n* (or cells *n-m*) for all shifts in the advective transport calculation. In advective-dispersive transport simulations (**TRANSPORT**), the initial equilibration also occurs at the first temperature of **REACTION_TEMPERATURE** *n* in cell *n*. However, depending on the setting of **temperature_retardation_factor**, an exchange of heat may take place that will cause the temperature of the cell to change as the advective-dispersive transport calculation progresses.

Example problems

The keyword **REACTION_TEMPERATURE** is used in example problem 2.

Related keywords

ADVECTION, **KINETICS**, **TRANSPORT**, and **REACTION**.

SAVE

This keyword data block is used to save the composition of a solution, exchange assemblage, gas phase, pure-phase assemblage, solid-solution assemblage, or surface assemblage following a batch-reaction calculation. The composition is stored internally in computer memory and can be retrieved subsequently with the **USE** keyword during the remainder of the computer run.

Example data block

```
Line 0a: SAVE equilibrium_phases 2
Line 0b: SAVE exchange 2
Line 0c: SAVE gas_phase 2
Line 0d: SAVE solid_solution 1
Line 0e: SAVE solution 2
Line 0f: SAVE surface 1
```

Explanation

Line 0: **SAVE** *keyword, number*

SAVE is the keyword for the data block.

keyword--One of six keywords, **exchange**, **equilibrium_phases**, **gas_phase**, **solid_solution**, **solution**, or **surface**. Options for **equilibrium_phases**: **equilibrium**, **equilibria**, **pure_phases**, or **pure**.

number--User defined positive integer to be associated with the respective composition. A range of numbers may also be given in the form *m-n*, where *m* and *n* are positive integers, *m* is less than *n*, and the two numbers are separated by a hyphen without intervening spaces.

Notes

SAVE affects only the internal storage of chemical-composition information and has effect only for the duration of a run. To save results to a permanent file, see **SELECTED_OUTPUT**. The **SAVE** data block applies only at the end of batch-reaction calculations and has no effect following initial solution, initial exchange-composition, initial surface-composition, initial gas-phase-composition, transport, or inverse calculations. During batch-reaction calculations, the compositions of the solution, exchange assemblage, gas phase, pure-phase assemblage, solid-solution assemblage, and surface assemblage vary to attain equilibrium. The compositions which exist at the end of a simulation in temporary storage locations are overwritten by the next simulation. These compositions are not automatically saved; however, they may be saved explicitly for use in subsequent simulations within the run by using the **SAVE** keyword. The **SAVE** keyword must be used for each type of composition that is to be saved (solution, exchange assemblage, gas phase, pure-phase assemblage, solid-solution assemblage, or surface assemblage). **SAVE** assigns *number* to the corresponding composition. If one of the compositions is saved in a *number* that already exists, the old composition is deleted. There is no need to save the compositions unless they are to be used in subsequent simulations within the run. Transport calculations automatically save the results of calculations after each step and the **SAVE** keyword has no effect in these calculations. Amounts of kinetic reactions (**KINETICS**) are automatically saved during all batch-reaction and transport calculations and can not be saved with the **SAVE** keyword. The **USE** keyword can be invoked in subsequent simulations to use the saved compositions in additional batch-reaction calculations.

Example problems

The keyword **SAVE** is used in example problems 3, 4, 7, 10, and 14.

Related keywords

EXCHANGE, EQUILIBRIUM_PHASES, GAS_PHASE, SELECTED_OUTPUT, SOLID_SOLUTIONS, SOLUTION, SURFACE, and **USE.**

SELECTED_OUTPUT

This keyword data block is used to produce a file that is suitable for processing by spreadsheets and other data-management software. It is possible to print selected entities from the compositions of solution, exchange assemblage, gas phase, pure-phase assemblage, solid-solution assemblage, and surface assemblage after the completion of each type of calculation. The selected-output file contains a column for each data item defined through the identifiers of **SELECTED_OUTPUT**. Initial print settings are shown in lines 1-4, 6-20 and 29 of the following example data block.

Example data block

```
Line 0: SELECTED_OUTPUT
Line 1:     -file      selected.out
Line 2:     -selected_out          true
Line 3:     -user_punch            true
Line 4:     -high_precision        false

          # set value for all identifiers on lines 6 through 20
Line 5:     -reset                 true
          # By default, data for the identifiers marked "true"
          # will be printed in order of line numbers,
          # By default, data for the identifiers marked "false"
          # will not be printed
Line 6:     -simulation            true
Line 7:     -state                 true
Line 8:     -solution              true
Line 9:     -distance              true
Line 10:    -time                  true
Line 11:    -step                  true
Line 12:    -ph                    true
Line 13:    -pe                    true
Line 14:    -reaction              false
Line 15:    -temperature           false
Line 16:    -alkalinity            false
Line 17:    -ionic_strength        false
Line 18:    -water                 false
Line 19:    -charge_balance        false
Line 20:    -percent_error         false
          # define printout of selected properties
Line 21:     -totals   Hfo_s  C  C(4)  C(-4)  N  N(0)
Line 21a:               Fe  Fe(3)  Fe(2)  Ca  Mg  Na  Cl
Line 22:     -molalities   Fe+2  Hfo_sOZn+  ZnX2
Line 23:     -activities   H+  Ca+2  CO2  HCO3-  CO3-2
Line 24:     -equilibrium_phases   Calcite Dolomite  Sphalerite
Line 25:     -saturation_indices   CO2(g)  Siderite
Line 26:     -gases                CO2(g)  N2(g)     O2(g)
Line 27:     -kinetic_reactants    CH2O    Pyrite
Line 28:     -solid_solutions      CaSO4   SrSO4
Line 29:     -inverse_modeling     true
```

Explanation

Line 0: **SELECTED_OUTPUT**

SELECTED_OUTPUT is the keyword for the data block. No additional data are read on this line. Optionally, **SELECTED_OUT**, **SELECT_OUTPUT**, or **SELECT_OUT**.

Line 1: **-file** *file name*

-file--Identifier allows definition of the name of the file where the selected results are written. Optionally, **file**, or **-f[ile]**.

file name--File name where selected results are written. If the file exists, the contents will be overwritten. File names must conform to operating system conventions. Default is **selected.out**.

Line 2: **-selected_out** [(*True* or *False*)]

-selected_out--Controls printing to the selected-output file. When **-selected_out** is set to **false**, all printing to the selected-output file is halted. Printing can be resumed if **-selected_out** is set to **true** in a **SELECTED_OUTPUT** data block in a subsequent simulation. Default is **true**. Optionally, **-se[lected_out]**. Note the hyphen is required to avoid a conflict with synonym of keyword **SELECTED_OUTPUT**.

Line 3: **-user_punch** [(*True* or *False*)]

-user_punch--Controls printing of information defined in **USER_PUNCH** to the selected-output file. When **-user_punch** is set to **false**, information defined in **USER_PUNCH** will not be written to the selected-output file. Writing this information to the selected-output file can be resumed if **-user_punch** is set to **true** in a **SELECTED_OUTPUT** data block in a subsequent simulation. Default is **true**. Optionally, **-u[ser_punch]**. Note the hyphen is required to avoid a conflict with the keyword **USER_PUNCH**.

Line 4: **-high_precision** [(*True* or *False*)]

-high_precision--Prints results to the selected-output file with extra numerical precision (12 decimal places, default is 3 or 4). In addition, the criterion for convergence of the calculations is set to 1e-12 (default is 1e-8, **-convergence_tolerance** in **KNOBS** data block). Default is **true**. Initial value at start of program is **false**. Optionally, **high_precision**, or **-h[igh_precision]**.

Line 5: **-reset** [(*True* or *False*)]

-reset--Resets all identifiers listed in lines 6-20 to **true** or **false**. Default is **true**. Optionally, **reset**, or **-r[eset]**.

Line 6: **-simulation** [(*True* or *False*)]

-simulation--Prints simulation number, or for advective-dispersive transport calculations the number of advective-dispersive transport simulations, to each line of the selected-output file if value is **true**, excludes print if value is **false**. Default is **true**. Initial value at start of program is **true**. Optionally, **simulation**, **sim**, or **-sim[ulation]**.

Line 7: **-state** [(*True* or *False*)]

-state--Prints type of calculation in each line of the selected-output file if value is **true**, excludes print if value is **false**. The following character strings are used to identify each calculation type: initial solution, "i_soln"; initial exchange composition, "i_exch"; initial surface composition, "i_surf"; initial gas-phase composition, "i_gas"; batch-reaction, "react"; inverse, "inverse"; advection,

"advect"; and transport, "transp". Default is **true**. Initial value at start of program is **true**. Optionally, **state**, or **-st[ate]**.

Line 8: **-solution** [(*True* or *False*)]

-solution--Prints solution number used for the calculation in each line of the selected-output file if value is **true**, excludes print if value is **false**. Default is **true**. Initial value at start of program is **true**. Optionally, **soln**, **-solu[tion]**, or **-soln**. Note the hyphen is required to avoid a conflict with the keyword **SOLUTION**.

Line 9: **-distance** [(*True* or *False*)]

-distance--Prints to the selected-output file (1) the X-coordinate of the cell for advective-dispersive transport calculations (**TRANSPORT**), (2) the cell number for advection calculations (**ADVECTION**), or (3) -99 for other calculations if value is **true**, excludes print if value is **false**. Default is **true**. Initial value at start of program is **true**. Optionally, **distance**, **dist**, or **-d[istance]**.

Line 10: **-time** [(*True* or *False*)]

-time--Prints to the selected-output file (1) the cumulative model time since the beginning of the simulation for batch-reaction calculations with kinetics, (2) the cumulative transport time since the beginning of the run (or since **-initial_time** identifier was last defined) for advective-dispersive transport calculations and advective transport calculations for which **-time_step** is defined, (3) the advection shift number for advective transport calculations for which **-time_step** is not defined, or (4) -99 for other calculations if value is **true**, excludes print if value is **false**. Default is **true**. Initial value at start of program is **true**. Optionally, **time**, or **-ti[me]**.

Line 11: **-step** [(*True* or *False*)]

-step--Prints to the selected-output file (1) advection shift number for transport calculations, (2) reaction step for batch-reaction calculations, or (3) -99 for other calculations if value is **true**, excludes print if value is **false**. Default is **true**. Initial value at start of program is **true**. Optionally, **step**, or **-ste[p]**.

Line 12: **-pH** [(*True* or *False*)]

-pH--Prints pH to each line of the selected-output file if value is **true**, excludes print if value is **false**. Default is **true**. Initial value at start of program is **true**. Optionally, **pH** (case insensitive).

Line 13: **-pe** [(*True* or *False*)]

-pe--Prints pe to each line of the selected-output file if value is **true**, excludes print if value is **false**. Default is **true**. Initial value at start of program is **true**. Optionally, **pe**.

Line 14: **-reaction** [(*True* or *False*)]

-reaction--Prints (1) reaction increment to the selected-output file if **REACTION** is used in the calculation or (2) -99 for other calculations if value is **true**, excludes print if value is **false**. Default is **true**. Initial value at start of program is **false**. Optionally, **rxn**, **-rea[ction]**, or **-rx[n]**. Note the hyphen is required to avoid a conflict with the keyword **REACTION**.

Line 15: **-temperature** [(*True* or *False*)]

-temperature--Prints temperature (Celsius) to the selected-output file if value is **true**, excludes print if value is **false**. Default is **true**. Initial value at start of program is **false**. Optionally, **temp**, **temperature**, or **-te[mperature]**.

Line 16: **-alkalinity** [(*True* or *False*)]

-alkalinity--Prints alkalinity (eq/kgw) to the selected-output file if value is **true**, excludes print if value is **false**. Default is **true**. Initial value at start of program is **false**. Optionally, **alkalinity**, **alk**, or **-al[kalinity]**.

Line 17: **-ionic_strength** [(*True* or *False*)]

-ionic_strength--Prints ionic strength to the selected-output file if value is **true**, excludes print if value is **false**. Default is **true**. Initial value at start of program is **false**. Optionally, **ionic_strength**, **mu**, **-io[nic_strength]**, or **-mu**.

Line 18: **-water** [(*True* or *False*)]

-water--Writes mass of water to the selected-output file if value is **true**, excludes print if value is **false**. Default is **true**. Initial value at start of program is **false**. Optionally, **water** or **-w[ater]**.

Line 19: **-charge_balance** [(*True* or *False*)]

-charge_balance--Writes charge balance of solution (eq) to the selected-output file if value is **true**, excludes print if value is **false**. Default is **true**. Initial value at start of program is **false**. Optionally, **charge_balance** or **-c[harge_balance]**.

Line 20: **-percent_error** [(*True* or *False*)]

-percent_error--Writes percent error in charge balance ($100\frac{cations - |anions|}{cations + |anions|}$) to the selected-output file if value is **true**, excludes print if value is **false**. Default is **true**. Initial value at start of program is **false**. Optionally, **percent_error** or **-per[cent_error]**.

Line 21: **-totals** *element list*

-totals--Identifier allows definition of a list of total concentrations that will be written to the selected-output file. Optionally, **totals**, or **-t[otals]**.

element list--List of elements, element valence states, exchange sites, or surface sites for which total concentrations will be written. The list may continue on subsequent line(s) (line 2a). Elements, valence states, exchange sites, and surface sites must have been defined in the first column of **SOLUTION_MASTER_SPECIES**, **EXCHANGE_MASTER_SPECIES**, or **SURFACE_MASTER_SPECIES** input. After each calculation, the concentration (mol/kgw) of each of the selected elements, element valence states, exchange sites, and surface sites will be written to the selected-output file. If an element is not defined or is not present in the calculation, its concentration will be printed as 0.

Line 22: **-molalities** *species list*

-molalities--Identifier allows definition of a list of species for which concentrations will be written to the selected-output file. Optionally, **molalities**, **mol**, or **-m[olalities]**.

species list--List of aqueous, exchange, or surface species for which concentrations will be written to the selected-output file. The list may continue on subsequent line(s). Species must have been defined by **SOLUTION_SPECIES**, **EXCHANGE_SPECIES**, or **SURFACE_SPECIES** input. After each calculation, the concentration (mol/kgw) of each species in the list will be written to the selected-output file. If a species is not defined or is not present in the calculation, its concentration will be printed as 0.

Line 23: **-activities** *species list*

-activities--Identifier allows definition of a list of species for which log of activity will be written to the selected-output file. Optionally, **activities**, or **-a[ctivities]**.

species list--List of aqueous, exchange, or surface species for which log of activity will be written to the selected-output file. The list may continue on subsequent line(s). Species must have been defined by **SOLUTION_SPECIES**, **EXCHANGE_SPECIES**, or **SURFACE_SPECIES** input. After each calculation, the log (base 10) of the activity of each of the species will be written to the selected-output file. If a species is not defined or is not present in the calculation, its log activity will be printed as -999.999.

Line 24: **-equilibrium_phases** *phase list*

-equilibrium_phases--Identifier allows definition of a list of pure phases for which (1) total amounts in the pure-phase assemblage and (2) moles transferred will be written to the selected-output file. Optionally, **-e[quilibrium_phases]**, or **-p[ure_phases]**. Note the hyphen is required to avoid a conflict with the keyword **EQUILIBRIUM_PHASES** and its synonyms.

phase list--List of phases for which data will be written to the selected-output file. The list may continue on subsequent line(s). Each phase must have been defined by **PHASES** input. After each calculation, two values are written to the selected-output file, (1) the moles of each of the phases (defined by **EQUILIBRIUM_PHASES**), and (2) the moles transferred. If the phase is not defined or is not present in the pure-phase assemblage, the amounts will be printed as 0.

Line 25: **-saturation_indices** *phase list*

-saturation_indices--Identifier allows definition of a list of phases for which saturation indices [or log (base 10) partial pressure for gases] will be written to the selected-output file. Optionally, **saturation_indices**, **si**, **-s[aturation_indices]**, or **-s[i]**.

phase list--List of phases for which saturation indices [or log (base 10) partial pressure for gases] will be written to the selected-output file. The list may continue on subsequent line(s). Each phase must have been defined by **PHASES** input. After each calculation, the saturation index of each of the phases will be written to the selected-output file. If the phase is not defined or if one or more of its constituent elements is not in solution, the saturation index will be printed as -999.999.

Line 26: **-gases** *gas-component list*

-gases--Identifier allows definition of a list of gas components for which the amount in the gas phase (moles) will be written to the selected-output file. Optionally, **gases**, or **-g[ases]**.

gas-component list--List of gas components. The list may continue on subsequent line(s). Each gas component must have been defined by **PHASES** input. After each calculation, the moles of each of the selected gas components in the gas phase will be written to the selected-output file. If a gas component is not defined or is not present in the gas phase, the amount will be printed as 0. Before the columns for the gas components, the flat file will contain the total pressure, total moles of gas components, and the volume of the gas phase. Partial pressures of any gas, including the components in the gas phase, can be obtained by use of the **-saturation_indices** identifier.

Line 27: **-kinetic_reactants** *reactant list*

-kinetic_reactants--Identifier allows definition of a list of kinetically controlled reactants for which two values are written to the selected-output file, (1) the current moles of the reactant, and (2) the moles transferred of the reactant. Optionally, **kin**, **-k[inetics]**, **kinetic_reactants**, or **-k[inetic_reactants]**. Note the hyphen is required to avoid a conflict with the keyword **KINETICS**.

reactant list--List of kinetically controlled reactants. The list may continue on subsequent line(s). Each reactant is identified by the rate name in the **KINETICS** data block. (The rate name in turn refers to a rate expression defined with **RATES** data block.) After each calculation, the moles and the moles transferred of each of the kinetically controlled reactants will be written to the selected-output file. If the reactant is not defined, the amount will be printed as 0.

Line 28: **-solid_solutions** *component list*

-solid_solutions--Identifier allows definition of a list of solid-solution components for which the moles in a solid solution is written to the selected-output file. Optionally, **-so[lid_solutions]**. Note the hyphen is required to avoid a conflict with the keyword **SOLID_SOLUTIONS**.

component list--List of solid-solution components. The list may continue on subsequent line(s). Each component is identified by the component name defined in the **SOLID_SOLUTIONS** data block. (The component names are also phase names which have been defined in the **PHASES** data block.) After each calculation, the moles of each solid-solution component in the list will be written to the selected-output file. If the component is not defined in any of the solid solutions, the amount will be printed as 0.

Line 29: **-inverse_modeling** [(*True* or *False*)]

-inverse_modeling--Prints results of inverse modeling to the selected-output file if value is **true**, excludes print if value is **false**. For each inverse model, three values are printed for each solution and phase defined in the **INVERSE_MODELING** data block: the central value of the mixing fraction or mole transfer, and the minimum and maximum of the mixing fraction or mole transfer, which are zero unless **-range** is specified in the **INVERSE_MODELING** data block. Default is **true**. Initial value at start of program is **true**. Optionally, **inverse** or **-i[nverse_modeling]**. Note the hyphen is required to avoid a conflict with the keyword **INVERSE_MODELING**.

Notes

The selected-output file contains a column for each data item defined through the identifiers of **SELECTED_OUTPUT**. Additional columns may be defined through the **USER_PUNCH** data block. In the input for the **SELECTED_OUTPUT** data block, all element names, species names, and phase names must be spelled exactly, including the case and charge for the species names. One line containing an entry for each of the items will be written to the selected-output file after each calculation--that is, after any initial solution, initial exchange-composition, initial surface-composition, or initial gas-phase-composition calculation and after each step in batch-reaction or each shift in transport calculations. The **-selected_output** identifier in the **PRINT** data block can be used to selectively suspend and resume writing results to the selected-output file. In transport

simulations, the frequency at which results are written to the selected-output file can be controlled by the **-punch_cells** and **-punch_frequency** identifiers in **ADVECTION** and **TRANSPORT** data blocks).

Several data items are included by default at the beginning of each line in the selected-output file to identify the type of calculation that has been performed. These data are described in lines 14 through 21. Data described in lines 22 through 28 may also be included in the output file. All of the data described by lines 14 through 28 may be simultaneously included or excluded from the selected-output file with the **-reset** identifier. Unlike most of PHREEQC input, the order in which the identifiers are entered is important when using the **-reset** identifier. Any identifier set before the **-reset** in the data block will be reset when **-reset** is encountered. Thus, **-reset** should be specified before any of the identifiers described in lines 14 through 28.

The first line of the selected-output file contains a description of each data column. The columns of data are written in the following order: items described by lines 14 through 28, totals, molalities, log activities, pure phases (two columns for each phase--total amount of phase and mole transfer for current calculation), saturation indices, gas-phase data (multiple columns), kinetically controlled reactants (two columns for each reactant--total amount of reactant and mole transfer for current calculation), solid-solution components, and data defined by the **USER_PUNCH** data block. A data item within an input list (for example an aqueous species within the **-molalities** list) is printed in the order of the list. If the selected-output file contains data for gases (**-gases** identifier), the total pressure, total moles in the gas phase, and the total volume of the gas phase precede the moles of each gas component specified by the identifier.

Example problems

The keyword **SELECTED_OUTPUT** is used in example problems 2, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, and 15.

Related keywords

ADVECTION, EQUILIBRIUM_PHASES, EXCHANGE_MASTER_SPECIES, EXCHANGE_SPECIES, GAS_PHASE, INVERSE_MODELING, KINETICS, KNOBS, PHASES, PRINT, REACTION, SOLUTION_MASTER_SPECIES, SOLID_SOLUTIONS, SOLUTION_SPECIES, SURFACE_MASTER_SPECIES, SURFACE_SPECIES, TRANSPORT, and **USER_PUNCH.**

SOLID_SOLUTIONS

This keyword data block is used to define a solid-solution assemblage. Each solid solution may be nonideal with two components or ideal with any number of components. The amount of each component in each solid solution is defined in this keyword data block. Any calculation involving solid solutions assumes that all solid solutions dissolve entirely and reprecipitate in equilibrium with the solution.

Example data block

```
Line 0:  SOLID_SOLUTIONS 1 Two solid solutions
Line 1a:       CaSrBaSO4     # greater than 2 components, ideal
Line 2a:            -comp    Anhydrite    1.500
Line 2b:            -comp    Celestite    0.05
Line 2c:            -comp    Barite       0.05
Line 1b:       Ca(x)Mg(1-x)CO3     # Binary, nonideal
Line 3:            -comp1    Calcite           0.097
Line 4:            -comp2    Ca.5Mg.5CO3       0.003
Line 5:            -temp     25.0
Line 6:            -tempk    298.15
Line 7:            -Gugg_nondim   5.08         1.90
```

Optional definitions of excess free-energy parameters for nonideal solid solutions:

```
Line 8:            -Gugg_kj              12.593  4.70
Line 9:            -activity_coefficients 24.05 1075. 0.0001 0.9999
Line 10:           -distribution_coefficients .0483 1248. .0001 .9999
Line 11:           -miscibility_gap  0.0428  0.9991
Line 12:           -spinodal_gap     0.2746  0.9483
Line 13:           -critical_point   0.6761  925.51
Line 14:           -alyotropic_point 0.5768  -8.363
Line 15:           -Thompson         17.303  7.883
Line 16:           -Margules         -0.62   7.6
```

Explanation

Line 0: **SOLID_SOLUTIONS** [*number*] [*description*]

SOLID_SOLUTIONS is the keyword for the data block. Optionally, **SOLID_SOLUTION**.

number--Positive number to designate the following solid-solution assemblage and its composition. A range of numbers may also be given in the form *m-n*, where *m* and *n* are positive integers, *m* is less than *n*, and the two numbers are separated by a hyphen without intervening spaces. Default is 1.

description--Optional comment that describes the surface assemblage.

Line 1: *solid-solution name*

solid-solution name--User-defined name of a solid solution.

Line 2: **-comp** *phase name, moles*

-comp--Identifier indicates a component of an ideal solid solution is defined. Component is part of the solid solution defined by the last Line 1 encountered. Optionally, **comp**, **component**, or **-c[omponent]**.

phase name--Name of the pure phase that is a component in the solid solution. A phase with this name must have been defined in a **PHASES** data block.

moles--Moles of the component in the solid solution.

Line 3: **-comp1** *phase name, moles*

-comp1--Identifier indicates the first component of a nonideal, binary solid solution is defined. The component is part of the solid solution defined by the last Line 1 encountered. Optionally, **comp1**.

phase name--Name of the pure phase that is component 1 of the nonideal solid solution. A phase with this name must have been defined in a **PHASES** data block.

moles--Moles of the component in the solid solution.

Line 4: **-comp2** *phase name, moles*

-comp2--Identifier indicates the second component of a nonideal, binary solid solution is defined. The component is part of the solid solution defined by the last Line 1 encountered. Optionally, **comp2**.

phase name--Name of the pure phase that is component 2 of the nonideal solid solution. A phase with this name must have been defined in a **PHASES** data block.

moles--Moles of the component in the solid solution.

Line 5: **-temp** *temperature in Celsius*

-temp--Temperature at which excess free-energy parameters are defined, in Celsius. Temperature, either **temp** (**tempc**) or **tempk**, is used if excess free-energy parameters are input with any of the following options: **-gugg_nondim**, **-activity_coefficients**, **-distribution_coefficients**, **-miscibility_gap**, **-spinodal_gap**, **-alyotropic_point**, or **-margules**. Optionally, **temp**, **tempc**, or **-t[empc]**. Default is 25°C.

Line 6: **-tempk** *temperature in Kelvin*

-tempk--Temperature at which excess free-energy parameters are defined, in Kelvin. Temperature, either **temp** (**tempc**) or **tempk**, is used if excess free-energy parameters are input with any of the following options: **-gugg_nondim**, **-activity_coefficients**, **-distribution_coefficients**, **-miscibility_gap**, **-spinodal_gap**, **-alyotropic_point**, or **-margules**. Optionally, **tempk**. Default is 298.15 K.

Line 7: **-Gugg_nondim** a_0, a_1

-Gugg_nondim--Nondimensional Guggenheim parameters are used to calculate dimensional Guggenheim parameters. Optionally, **gugg_nondimensional**, **parms**, **-g[ugg_nondimensional]**, or **-p[arms]**.

a_0--Guggenheim a_0 parameter, dimensionless. Default is 0.0.

a_1--Guggenheim a_1 parameter, dimensionless. Default is 0.0.

Line 8: **-Gugg_kJ** g_0, g_1

-Gugg_kJ--Guggenheim parameters with dimensions of kJ/mol define the excess free energy of the nonideal, binary solid solution. Optionally, **gugg_kJ**, or **-gugg_k[J]**.

g_0--Guggenheim a_0 parameter, kJ/mol. Default is 0.0.

g_1--Guggenheim a_1 parameter, kJ/mol. Default is 0.0.

Line 9: **-activity_coefficients** a_{comp_1}, a_{comp_2}, x_1, x_2

-activity_coefficients--Activity coefficients for components 1 and 2 are used to calculate dimensional Guggenheim parameters. Optionally, **activity_coefficients**, or **-a[ctivity_coefficients]**.

a_{comp_1} --Activity coefficient for component 1 in the solid solution. No default.

a_{comp_2} --Activity coefficient for component 2 in the solid solution. No default.

x_1--Mole fraction of component 2 for which a_{comp_1} applies. No default.

x_2--Mole fraction of component 2 for which a_{comp_2} applies. No default.

Line 10: **-distribution_coefficients** k_1, k_2, x_1, x_2

-distribution_coefficients--Two distribution coefficients are used to calculate dimensional Guggenheim parameters. Optionally, **distribution_coefficients**, or **-d[istribution_coefficients]**.

k_1 --Distribution coefficient of component 2 at mole fraction x_1 of component 2, expressed as $\frac{(\chi_2/\chi_1)}{(a_2/a_1)}$, where χ is the mole fraction in the solid and a is the aqueous activity. No default.

k_2 --Distribution coefficient of component 2 at mole fraction x_2 of component 2, expressed as above. No default.

x_1--Mole fraction of component 2 for which k_1 applies. No default.

x_2--Mole fraction of component 2 for which k_2 applies. No default.

Line 11: **-miscibility_gap** x_1, x_2

-miscibility_gap--The mole fractions of component 2 that delimit the miscibility gap are used to calculate dimensional Guggenheim parameters. Optionally, **miscibility_gap**, or **-m[iscibility_gap]**.

x_1--Mole fraction of component 2 at one end of the miscibility gap. No default.

x_2--Mole fraction of component 2 at the other end of the miscibility gap. No default.

Line 12: **-spinodal_gap** x_1, x_2

-spinodal_gap--The mole fractions of component 2 that delimit the spinodal gap are used to calculate dimensional Guggenheim parameters. Optionally, **spinodal_gap**, or **-s[pinodal_gap]**.

x_1--Mole fraction of component 2 at one end of the spinodal gap. No default.

x_2--Mole fraction of component 2 at the other end of the spinodal gap. No default.

Line 13: **-critical_point** x_{cp}, t_{cp}

-critical_point--The mole fraction of component 2 at the critical point and the critical temperature (Kelvin) are used to calculate dimensional Guggenheim parameters. Optionally, **critical_point**, or **-cr[itical_point]**.

x_{cp}--Mole fraction of component 2 at the critical point. No default.

t_{cp}--Critical temperature, in Kelvin. No default.

Line 14: **-alyotropic_point** x_{aly}, $\log_{10}(\Sigma\Pi)$

-alyotropic_point--The mole fraction of component 2 at the alyotropic point and the total solubility product at that point are used to calculate dimensional Guggenheim parameters. Optionally, **alyotropic_point**, or **-al[yotropic_point]**.

x_{aly}--Mole fraction of component 2 at the alyotropic point. No default.

$\log_{10}(\Sigma\Pi)$--Total solubility product at the alyotropic point, where $\Sigma\Pi = (a_1 + a_2)a_{\text{common ion}}$. No default.

Line 15: **-Thompson** wg_2, wg_1

-Thompson--Thompson and Waldbaum parameters wg_2 and wg_1 are used to calculate dimensional Guggenheim parameters. Optionally, **thompson**, or **-th[ompson]**.

wg_2--Thompson and Waldbaum parameter wg_2, kJ/mol. No default.

wg_1--Thompson and Waldbaum parameter wg_1, kJ/mol. No default.

Line 16: **-Margules** $alpha_2$, $alpha_3$

-Margules--Margules parameters $alpha_2$ and $alpha_3$ are used to calculate dimensional Guggenheim parameters. Optionally, **Margules**, or **-Ma[rgules]**.

$alpha_2$--Margules parameter $alpha_2$, dimensionless. No default.

$alpha_3$--Margules parameter $alpha_3$, dimensionless. No default.

Notes

Multiple solid solutions may be defined by multiple sets of lines 1, 2, 3, and 4. Line 2 may be repeated as necessary to define all the components of an ideal solid solution. Nonideal solid solution components must be defined with Lines 3, and 4. Calculations with solid solutions assume that the entire solid recrystallizes to be in equilibrium with the aqueous phase. This assumption is usually unrealistic because it is likely that only the outer layer of a solid would re-equilibrate with the solution, even given long periods of time. In most cases, the use of ideal solid solutions is also unrealistic because nonideal effects are nearly always present in solids.

Lines 7-16 provide alternative ways of defining the excess free energy of a nonideal, binary solid solution. Only one of these lines should be included in the definition of a single solid solution. The parameters in the example data block are taken from Glynn (1991) and Glynn (1990) for "nondefective" calcite (log K -8.48) and dolomite (expressed as $Ca_{0.5}Mg_{0.5}CO_3$, log K -8.545; note that a phase for dolomite with the given name, composition, and log K would have to be defined in a **PHASES** data block because it differs from the standard stoichiometry for dolomite in the databases). In the example data block, lines 7 through 16, except Line 14 (alyotropic point), define the same dimensional Guggenheim parameters. Internally, the program converts any one of these forms of input into dimensional Guggenheim parameters. When a batch-reaction or transport calculation is performed, the temperature of the calculation (as defined by mixing of solutions, **REACTION_TEMPERATURE** data block, or heat transport in **TRANSPORT** simulations) is used to convert the dimensional Guggenheim parameters to nondimensional Guggenheim parameters, which are then used in the calculation.

The identifiers **-gugg_nondim**, **-activity_coefficients**, **-distribution_coefficients**, **-miscibility_gap**, **-spinodal_gap**, **-alyotropic_point**, or **-margules** define parameters for a particular temperature which are converted to dimensional Guggenheim parameters using the default temperature of 25 C or the temperature specified in line 5 or 6. If more than one line 5 and (or) 6 is defined, the last definition will take precedence. If **-alyotropic_point** or **-distribution_coefficients** identifiers are used to define excess free-energy parameters, the dimensional Guggenheim parameter are dependent on (1) the values included with these two identifiers and (2) the equilibrium constants for the pure-phase components. The latter are defined by a **PHASES** data block in the input file or database file.

The parameters for excess free energy are dependent on which component is labeled "1" and which component is labeled "2". It is recommended that the component with the smaller value of log K be selected as component 1 and the component with the larger value of log K be selected as component 2. The excess free-energy parameters must be consistent with this numbering. A positive value of a_1 (nondimensional Guggenheim parameter) or g_1 (dimensional Guggenheim parameter) will result in skewing the excess free-energy function toward component 2 and, if a miscibility gap is present, it will not be symmetric about a mole fraction of 0.5, but instead will be shifted toward component 2. In the calcite-dolomite example, the positive value of a_1 (1.90) results in a miscibility gap extending almost to pure dolomite (mole fractions of miscibility gap are 0.0428 to 0.9991).

After a batch-reaction with a solid-solution assemblage has been simulated, it is possible to save the resulting solid-solution compositions with the **SAVE** keyword. If the new compositions are not saved, the solid-solution compositions will remain the same as they were before the batch reaction. After it has been defined or saved, the solid-solution assemblage may be used in subsequent simulations through the **USE** keyword. Solid-solution compositions are automatically saved following each shift in transport calculations.

Example problems

The keyword **SOLID_SOLUTIONS** is used in example problem 10.

Related keywords

PHASES, **SAVE solid_solution**, and **USE solid_solution**.

SOLUTION

This keyword data block is used to define the temperature and chemical composition of initial solutions. All input concentrations are converted internally to units of molality or, equivalently, moles of elements and element valence states and mass of water. Speciation calculations are performed on each solution and each solution is then available for subsequent batch-reaction, transport, or inverse-modeling calculations. Capabilities exist to adjust individual element concentrations to achieve charge balance or equilibrium with a pure phase.

Example data block

```
Line 0:  SOLUTION 25 Test solution number 25
Line 1:       temp       25.0
Line 2:       pH         7.0      charge
Line 3:       pe         4.5
Line 4:       redox      O(-2)/O(0)
Line 5:       units      ppm
Line 6:       density    1.02
Line 7a:      Ca         80.
Line 7b:      S(6)       96.      as SO4
Line 7c:      S(-2)      1.       as S
Line 7d:      N(5) N(3)  14.      as N
Line 7e:      O(0)       8.0
Line 7f:      C          61.0     as HCO3       CO2(g)      -3.5
Line 7g:      Fe         55.      ug/kgs as Fe S(6)/S(-2) Pyrite
Line 8a:      -isotope   13C      -12.    1.  # permil PDB
Line 8b:      -isotope   34S      15.     1.5 # permil CDT
Line 9:       -water     0.5      # kg
```

Explanation

Line 0: **SOLUTION** [*number*] [*description*]

SOLUTION is the keyword for the data block.

number--Positive number to designate the following solution composition. A range of numbers may also be given in the form *m-n*, where *m* and *n* are positive integers, *m* is less than *n*, and the two numbers are separated by a hyphen without intervening spaces. Default is 1.

description--Optional comment that describes the solution.

Line 1: **temp** *temperature*

temp--Indicates temperature is entered on this line. Optionally, **temperature**, or **-t[emperature]**.

temperature--Temperature in degrees Celsius. Default 25 °C.

Line 2: **pH** *pH* [(**charge** or *phase name* [*saturation index*])]

pH--Indicates pH is entered on this line. Optionally, **-ph**.

pH--pH value, negative log of the activity of hydrogen ion.

charge--Indicates pH is to be adjusted to achieve charge balance. If **charge** is specified for pH, it may not be specified for any other element.

phase name--pH will be adjusted to achieve specified saturation index with the specified phase.

saturation index--pH will be adjusted to achieve this saturation index for the specified phase. Default is 0.0.

If line 2 is not entered, the default pH is 7.0. Specifying both **charge** and a phase name is not allowed. Be sure that specifying a phase is reasonable; it may not be possible to adjust the pH to achieve the specified saturation index.

Line 3: **pe** *pe* [(**charge** or *phase name* [*saturation index*])]

pe--Indicates pe is entered on this line. Optionally, **-pe**.

pe--pe value, conventional negative log of the activity of the electron.

charge--(Not recommended) indicates pe is to be adjusted to achieve charge balance.

phase name--pe will be adjusted to achieve specified saturation index with the specified phase.

saturation index--pe will be adjusted to achieve this saturation index for the specified phase. Default is 0.0.

If line 3 is not entered, the default pe is 4.0. Specifying both **charge** and a phase name is not allowed. Adjusting pe for charge balance is not recommended. Care should also be used in adjusting pe to a fixed saturation index for a phase because frequently this is not possible.

Line 4: **redox** *redox couple*

redox--Indicates the definition of a redox couple that is used to calculate pe. This pe will be used for any redox element for which a pe is needed to determine the distribution of the element among its valence states. Optionally, **-r[edox]**.

redox couple--Redox couple which defines pe. A redox couple is specified by two valence states of an element separated by a "/". No spaces are allowed.

If line 4 is not entered, the input pe value will be the default. The use of **-redox** does not change the input pe. The example data block uses the dissolved oxygen concentration [defined by O(0) in line 7e] and the redox half-reaction for formation of $O_{2(aq)}$ from water (defined in the **SOLUTION_SPECIES** data block of the default databases) to calculate a default pe.

Line 5: **units** *concentration units*

units--Indicates default concentration units are entered on this line. Optionally, **-u[nits]**.

concentration units--Default concentration units. Three groups of concentration units are allowed, concentration (1) per liter ("/L"), (2) per kilogram solution ("/kgs"), or (3) per kilogram water ("/kgw"). All concentration units for a solution must be within the same group. Within a group, either grams or moles may be used, and prefixes milli (m) and micro (u) are acceptable. Parts per thousand, "ppt"; parts per million, "ppm"; and parts per billion, "ppb", are acceptable in the "per kilogram solution" group. Default is mmol/kgw (millimoles per kilogram water).

Line 6: **density** *density*

density--Indicates density is entered on this line. Optionally, **dens**, or **-d[ensity]**.

density--Density of the solution, kg/L (equals g/cm^3). Default is 1.0.

The density is used only if the input concentration units are "per liter".

Line 7: *element list, concentration,* [*units*], ([**as** *formula*] or [**gfw** *gfw*]), [*redox couple*], [(**charge** or *phase name* [*saturation index*])]

element list--An element name or a list of element valence states separated by white space (see line 7d). The element names and valence states must correspond to the items in the first column in **SOLUTION_MASTER_SPECIES**.

concentration--Concentration of element in solution or sum of concentrations of element valence states in solution.

units--Concentration unit for element (see line 7g). If units are not specified, the default units (**units** value if line 5 is present, or mmol/kgw if line 5 is absent) are assumed.

as *formula*--Indicates a chemical formula, *formula*, will be given from which a gram formula weight will be calculated. A gram formula weight is needed only when the input concentration is in mass units. The calculated gram formula weight is used to convert mass units into mole units for this element and this solution; it is not stored for further use. If a gram formula weight is not specified, the default is the gram formula weight defined in **SOLUTION_MASTER_SPECIES**. For alkalinity, the formula should give the gram equivalent weight. For alkalinity reported as calcium carbonate, the formula for the gram equivalent weight is $Ca_{0.5}(CO3)_{0.5}$; this is the default in the *phreeqc.dat* and *wateq4f.dat* database files distributed with this program.

gfw *gfw*--Indicates a gram formula weight, *gfw*, will be entered. A gram formula weight is needed only when the input concentration is in mass units. The specified gram formula weight is used to convert mass units into mole units only for this element and this solution; it is not stored for further use. If a gram formula weight is not specified, the default is the gram formula weight defined in **SOLUTION_MASTER_SPECIES**. For alkalinity, the gram equivalent weight should be entered. For alkalinity reported as calcium carbonate, the gram equivalent weight is approximately 50.04 g/eq.

redox couple--Redox couple to use for the element or element valence states in *element list*. Definition of a redox couple is appropriate only when the element being defined is redox active and either (1) the total amount of the element is specified (no parentheses in the element name) or (2) two or more valence-states are specified (a valence state is defined in parentheses following element name); definition of a redox couple is not needed for non-redox-active elements or for individual valence states of an element. Initial solution calculations do not require redox equilibrium among all redox couples of all redox elements. Specifying a *redox couple* will force selective redox equilibrium; the redox element being defined will be in equilibrium with the specified *redox couple*. A redox couple is specified by two valence states of an element separated by a "/". No spaces are allowed. The specified redox couple overrides the default pe or default redox couple and is used to calculate a pe by which the element is distributed among valence states. If no redox couple is entered, the default redox couple defined by line 4 will be used, or the pe if line 4 is not entered.

charge--Indicates the concentration of this element will be adjusted to achieve charge balance. The element must have ionic species. If **charge** is specified for one element, it may not be specified for pH or any other element. (Note that it is possible to have a greater charge imbalance than can be adjusted by removing all of the specified element, in which case the problem is unsolvable.)

phase name--The concentration of the element will be adjusted to achieve a specified saturation index for the given pure phase. Be sure that specifying equilibrium with the phase is reasonable; the ele-

ment should be a constituent in the phase. *Phase name* may not be used if **charge** has been specified for this element.

saturation index--The concentration of the element will be adjusted to achieve this saturation index for the given pure phase. Default is 0.0.

Line 8: **-isotope** *name, value, [uncertainty limit]*

-isotope--Indicates isotopic composition for an element or element valence state is entered on this line. Isotope data are used only in inverse modeling. Optionally, **isotope**, or **-i[sotope]**.

name--Name of the isotope. The name must begin with mass number followed by an element or element-valence-state name that is defined through **SOLUTION_MASTER_SPECIES**.

value--Isotopic composition of element or element valence state; units are usually a ratio, permil, or percent modern carbon.

uncertainty limit--The uncertainty limit to be used in inverse modeling. This value is optional in the **SOLUTION** data block and alternatively a default uncertainty limit may be used (see **INVERSE_MODELING**) or an uncertainty limit may be defined with the **-isotopes** identifier of the **INVERSE_MODELING** data block.

Line 9: **-water** *mass*

-water--Indicates mass of water is entered on this line. Molalities of solutes are calculated from input concentrations and the moles of solutes are determined by the mass of water in solution. Optionally, **water**, or **-w[ater]**.

mass--Mass of water in the solution (kg). Default is 1 kg.

Notes

The order in which the lines of **SOLUTION** input are entered is not important. Specifying both "**as**" and "**gfw**" within a single line is not allowed. Specifying both "**charge**" and a phase name within a single line is not allowed. Specifying the concentration of a valence state or an element concentration twice is not allowed. For example, specifying concentrations for both total Fe and Fe(+2) is not allowed, because ferrous iron is implicitly defined twice.

Alkalinity or total carbon or both may be specified in solution input. If both alkalinity and total carbon are specified, the pH is adjusted to attain the specified alkalinity. If the units of alkalinity are reported as calcium carbonate, the correct formula to use is "**as** Ca0.5(CO3)0.5", because the gram equivalent weight is 50.04, which corresponds to one half the formula $CaCO_3$. However, to avoid frequent errors, if "**as** CaCO3" is entered, the value of 50.04 will still be used as the equivalent weight.

All concentrations defined in the **SOLUTION** data block are converted into molality. The absolute number of moles is usually numerically equal to the molality because a kilogram of solvent water is assumed. It is possible to define a solution with a different mass of water by using the **-water** identifier. In that case, the moles of solutes are scaled to produce the molality as converted from the input data. A 1-mol/kgw solution of NaCl with "**-water** 0.5" has 0.5 moles of Na and Cl and 0.5 kilograms of water. Batch-reaction calculations may also cause the mass of water in a solution to deviate from 1 kilogram.

Isotope values are used only in conjunction with the **INVERSE_MODELING** data block. Uncertainty limits for isotopes in mole-balance modeling may be defined in three ways: default uncertainty limits may be used,

uncertainty limits may be defined in the **SOLUTION** data block, or uncertainty limits may be defined in the **INVERSE_MODELING** data block. Uncertainty limits defined in the **INVERSE_MODELING** data block take precedence over the **SOLUTION** data block, which in turn take precedence over the defaults given in table 5.

After a batch reaction has been simulated, it is possible to save the resulting solution composition with the **SAVE** keyword. If the new batch-reaction composition is not saved, the solution composition will remain the same as it was before the batch reaction. After it has been defined or saved, a solution may be used in subsequent simulations through the **USE** keyword. Solution compositions for a cell are automatically saved after each shift in transport calculations.

Example problems

The keyword **SOLUTION** is used in all example problems, 1 through 18.

Related keywords

INVERSE_MODELING, SAVE solution, SOLUTION_SPECIES, SOLUTION_MASTER_SPECIES, and **USE solution.**

SOLUTION_MASTER_SPECIES

This keyword is used to define the correspondence between element names and aqueous primary and secondary master species. The alkalinity contribution of the master species, the gram formula weight used to convert mass units, and the element gram formula weight also are defined in this data block. Normally, this data block is included in the database file and only additions and modifications are included in the input file.

Example data block

```
Line 0:  SOLUTION_MASTER_SPECIES
Line 1a:     H             H+        -1.0   1.008            1.008
Line 1b:     H(0)          H2        0.0    1.008
Line 1c:     S             SO4-2     0.0    SO4              32.06
Line 1d:     S(6)          SO4-2     0.0    SO4
Line 1e:     S(-2)         HS-       1.0    S
Line 1f:     Alkalinity    CO3-2     1.0    Ca0.5(CO3)0.5    50.04
```

Explanation

Line 0: **SOLUTION_MASTER_SPECIES**

Keyword for the data block. No other data are input on the keyword line.

Line 1: *element name, master species, alkalinity, (gram formula weight* or *formula), gram formula weight of element*

element name--An element name or an element name followed by a valence state in parentheses. The element name must begin with a capital letter, followed by zero or more lower case letters or underscores ("_").

master species--Formula for the master species, including its charge. If the *element name* does not contain a valence state in parentheses, the corresponding master species is a primary master species. If the element name does contain a valence state in parentheses, the master species is a secondary master species. The *master species* must be defined in the **SOLUTION_SPECIES** data block.

alkalinity--Alkalinity contribution of the master species. The alkalinity contribution of aqueous non-master species will be calculated from the alkalinities assigned to the master species.

gram formula weight--Default value used to convert input data in mass units to mole units for the element or element valence. For alkalinity, the gram equivalent weight is entered. Either *gram formula weight* or *formula* is required, but these items are mutually exclusive.

formula--Chemical formula used to calculate gram formula weight, which is used to convert input data from mass units to mole units for the element or element valence. For alkalinity, the formula for the gram equivalent weight is entered. Either *gram formula weight* or *formula* is required, but these items are mutually exclusive.

gram formula weight for element--This field is required for primary master species and must be the gram formula weight for the pure element, not for an aqueous species.

Notes

Line 1 must be repeated for each element and each element valence state to be used by the program. Each element must have a primary master species. If secondary master species are defined for an element, then the primary master species additionally must be defined as a secondary master species for one of the valence states. PHREEQC will reduce all chemical reaction equations to a form that contains only primary and secondary master species. Each primary master species must be defined by **SOLUTION_SPECIES** input to have an identity reaction with log *K* of 0.0. For example, the definition of the primary master species SO_4^{-2} in the **SOLUTION_SPECIES** data block of the database *phreeqc.dat* is SO4-2 = SO4-2, log *K* 0.0. Secondary master species that are not primary master species must be defined by **SOLUTION_SPECIES** input to have a reaction that contains electrons and the log *K* in general will not be 0.0. For example, the definition of the secondary master species HS^- in the **SOLUTION_SPECIES** data block of the database *phreeqc.dat* is SO4-2 + 9 H+ + 8 e- = HS- + 4 H2O, log *K* 33.65. The treatment of alkalinity is a special case and "Alkalinity" is defined as an additional element. In most cases, the definitions in **SOLUTION_MASTER_SPECIES** for alkalinity and carbon in the default database files should be used without modification.

The *gram formula weight* and *formula* are defined for convenience in converting units. For example, if data for nitrate are consistently reported in mg/L of nitrate as NO_3^-, then *gram formula weight* should be set to 62.0 or *formula* should be set to "NO3". Then it will not be necessary to use the **as** or **gfw** options in the **SOLUTION** or **SOLUTION_SPREAD** data block. If nitrate is reported as mg/L as N, then *gram formula weight* should be set to 14.0 or *formula* should be set to "N", as is the case in the default databases. These variables (*gram formula weight* and *formula*) are only used if the concentration units are in terms of mass; if the data are reported in moles, then the variables are not used. The value of *gram formula weight for element* is required for primary master species and its value is used to calculate the gram formula weight when a *formula* is given either in **SOLUTION_MASTER_SPECIES**, **SOLUTION**, or **SOLUTION_SPREAD** data block.

Example problems

The keyword **SOLUTION_MASTER_SPECIES** is used in example problems 1, 9 and 15. See also the listing of the default database file in Attachment B.

Related keywords

SOLUTION, **SOLUTION_SPREAD**, and **SOLUTION_SPECIES**.

SOLUTION_SPECIES

This keyword data block is used to define chemical reaction, log *K*, and activity-coefficient parameters for each aqueous species. Normally, this data block is included in the database file and only additions and modifications are included in the input file.

Example data block

```
Line 0:  SOLUTION_SPECIES
Line 1a: SO4-2 = SO4-2
Line 2a:       log_k      0.0
Line 5a:       -gamma     5.0      -0.04
Line 1b: SO4-2 + 9H+ + 8e- = HS- + 4H2O
Line 2b:       log_k      33.652
Line 3b:       delta_h    -40.14
Line 5b:       -gamma     3.5      0.0
Line 1c: H2O = OH- + H+
Line 2c:       log_k      -14.000
Line 3c:       delta_h    13.362   kcal
Line 4c:       -a_e -283.971 -0.05069842 13323.0  102.24447 -1119669.0
Line 5c:       -gamma     3.5      0.0
Line 1d: HS-   = S2-2 + H+
Line 2d:       log_k      -14.528
Line 3d:       delta_h    11.4
Line 6:        -no_check
Line 7d:       -mole_balance     S(-2)2
```

Explanation

Line 0: **SOLUTION_SPECIES**

Keyword for the data block. No other data are input on the keyword line.

Line 1: *Association reaction*

Association reaction for aqueous species. The defined species must be the first species to the right of the equal sign. The association reaction must precede any identifiers related to the aqueous species. The association reaction is an identity reaction for each primary master species.

Line 2: **log_k** *log K*

log_k--Identifier for log *K* at 25°C. Optionally, **-log_k**, **logk**, **-l[og_k]**, or **-l[ogk]**.

log K--Log *K* at 25°C for the reaction. *Log K* must be 0.0 for primary master species. Default is 0.0.

Line 3: **delta_h** *enthalpy, [units]*

delta_h--Identifier for enthalpy of reaction at 25°C. Optionally, **-delta_h**, **deltah**, **-d[elta_h]**, or **-d[eltah]**.

enthalpy--Enthalpy of reaction at 25°C for the reaction. Default is 0.0.

units--Default units are kilojoules per mole. Units may be calories, kilocalories, joules, or kilojoules per mole. Only the energy unit is needed (per mole is assumed) and abbreviations of these units are acceptable. Explicit definition of units for all enthalpy values is recommended. The enthalpy of reaction is used in the van't Hoff equation to determine the temperature dependence of the

equilibrium constant. Internally, all enthalpy calculations are performed with the units of kilojoules per mole.

Line 4: **-analytical_expression** A_1, A_2, A_3, A_4, A_5

-analytical_expression--Identifier for coefficients for an analytical expression for the temperature dependence of log K. Optionally, **analytical_expression**, **a_e**, **ae**, **-a[nalytical_expression]**, **-a[_e]**, **-a[e]**.

A_1, A_2, A_3, A_4, A_5--Five values defining log K as a function of temperature in the expression

$$\log_{10}K = A_1 + A_2T + \frac{A_3}{T} + A_4\log_{10}T + \frac{A_5}{T^2}, \text{ where } T \text{ is in Kelvin.}$$

Line 5: **-gamma** *Debye-Hückel a, Debye-Hückel b*

-gamma--Indicates activity-coefficient parameters are to be entered. If **-gamma** is not input for a species, for charged species the Davies equation is used to calculate the activity coefficient:

$\log\gamma = -Az^2\left(\frac{\sqrt{\mu}}{1+\sqrt{\mu}} - 0.3\mu\right)$; for uncharged species the following equation is used

$\log\gamma = 0.1\mu$. If **-gamma** is entered, then the equation from WATEQ (Truesdell and Jones, 1974) is used, $\log\gamma = \frac{-Az^2\sqrt{\mu}}{1+Ba^o\sqrt{\mu}} + b\mu$. In these equations, γ is the activity coefficient, μ is ionic strength, and A and B are constants at a given temperature. Optionally, **-g[amma]**.

Debye-Hückel a--Parameter a^o in the WATEQ activity-coefficient equation.

Debye-Hückel b--Parameter b in the WATEQ activity-coefficient equation.

Line 6: **-no_check**

-no_check--Indicates the reaction equation should not be checked for charge and elemental balance. The only exceptions might be polysulfide species which assume equilibrium with a solid phase; this assumption has the effect of removing solid sulfur from the mass-action equation. By default, all equations are checked. However, the identifier **-mole_balance** is needed to ensure that the proper number of atoms of each element are included in mole-balance equations (see **-mole_balance**). Optionally, **no_check**, or **-n[o_check]**.

Line 7: **-mole_balance** *formula*

-mole_balance--Indicates the stoichiometry of the species will be defined explicitly. Optionally, **mole_balance**, **mass_balance**, **mb**, **-m[ole_balance]**, **-mass_balance**, **-m[b]**.

formula--Chemical formula defining the stoichiometry of the species. Normally, both the stoichiometry and mass-action expression for the species are determined from the chemical equation that defines the species. Rarely, it may be necessary to define the stoichiometry of the species separately from the mass-action equation. The polysulfide species provide an example. These species are usually assumed to be in equilibrium with native sulfur. The activity of a pure solid is 1.0 and thus the term for native sulfur does not appear in the mass-action expression (Line 1d). The S_2^- species contains two atoms of sulfur, but the chemical equation indicates it is formed from species containing a total of one sulfur atom. The **-mole_balance** identifier is needed to give the correct stoichiometry. Note that unlike all other chemical formulas used in PHREEQC, the valence state of

the element can and should be included in the formula (Line 7d). The example indicates that the polysulfide species will be summed into the S(-2) mole-balance equation in any initial solution calculations for which total sulfide is defined.

Notes

Line 1 must be entered first in the definition of a species. Additional sets of lines (lines 1-7 as needed) may be added to define all of the aqueous species. A log *K* must be defined for each species with either **log_k** (line 2) or **-analytical_expression** (line 4); default is 0.0, but is not meaningful except for primary master species. In this example data block, the following types of aqueous species are defined: (a) a primary master species, SO_4^{-2}, the reaction is an identity reaction and log *K* is 0.0; (b) a secondary master species, HS^-, the reaction contains electrons; (c) an aqueous species that is not a master species, OH^-; and (d) an aqueous species for which the chemical equation does not balance, S_2^{-2}.

By default, equation checking for charge and elemental balance is in force for each equation that is processed. Checking can only be disabled by using **-no_check** for each equation that is to be excluded from the checking process.

Example problems

The keyword **SOLUTION_SPECIES** is used in example problem 1, 9 and 15. See also the listing of the default database file in Attachment B.

Related keywords

SOLUTION_MASTER_SPECIES.

SOLUTION_SPREAD

The **SOLUTION_SPREAD** data block is an alternative input format for **SOLUTION** that is compatible with the output of many spreadsheet programs. Input for **SOLUTION_SPREAD** is transposed from the input for **SOLUTION**, that is, the rows of input for **SOLUTION** become the columns of input for **SOLUTION_SPREAD**. The data are entered in columns which are tab-delimited ("\t" in the example data block).

Example data block

```
Line 0:  SOLUTION_S    # "\t" indicates the tab character
Line 1:  -temp         25
Line 2:  -ph           7.1
Line 3:  -pe           4
Line 4:  -redox        O(0)/O(-2)
Line 5:  -units        mmol/kgw
Line 6:  -density      1
Line 7:  -water        1.0
Line 8a: -isotope      34S        15.0   1.0
Line 8b: -isotope      13C       -12.0
Line 9:  -isotope_uncertainty   13C    1.0
Line 10: Number\t    13C\t uncertainty\t  pH\t  Ca\t  Na\t   Cl\t     Alkalinity\t  Description
Line 11:       \t       \t            \t    \t    \t    \t     \t mg/kgw as HCO3\t
Line 12a: 10-11\t  -10.2\t        0.05\t 6.9\t  23\t   6\t 10.5\t             61\t  soln 10-11
Line 12b:     1\t  -12.1\t         0.1\t    \t  17\t   6\t   9.\t             55\t  My well 1
Line 12c:     5\t  -14.1\t         0.2\t    \t  27\t   9\t  9.5\t             70\t  My well 5
```

Explanation

Line 0: **SOLUTION_SPREAD**

Keyword for the data block. No other data are input on the keyword line.

Line 1: **-temp** *temperature*

-temp--Identifier for temperature. The temperature will be used for all subsequent solutions in the data block if no column has the heading **temperature** (or **temp**) or if the entry for the **temperature** column is empty for a solution. Optionally, **temp**, **-t[emp]**, **temperature**, or **-t[emperature]**.

temperature--Temperature, degrees Celsius. Default is 25.0.

Line 2: **-pH** *pH*

-pH--Identifier for pH. The pH will be used for all subsequent solutions in the data block if no column has the heading **pH** or if the entry for the **pH** column is empty for a solution. Optionally, **ph**.

pH--pH value, negative log of the activity of hydrogen ion. Default is 7.0.

Line 3: **-pe** *pe*

-pe--Identifier for pe. The value *pe* will be used as default for all subsequent solutions in the data block if no column has the heading **pe** or if the entry for the **pe** column is empty for a solution. Optionally, **pe**.

pe--pe value, conventional negative log of the activity of the electron. Default is 4.0.

Line 4: **-redox** *redox couple*

-redox--Identifier for the redox couple to be used to calculate pe. This pe will be used for any redox element for which a pe is needed to determine the distribution of the element among its valence states. The redox couple will be used for all subsequent solutions in the data block if no column

has the heading **redox** or if the entry for the **redox** column is empty for a solution. If no redox couple is specified, the pe will be used. Optionally, **redox** or **-r[edox]**.

redox couple--Redox couple to use for pe calculations. A redox couple is specified by two valence states of an element separated by a "/". No spaces are allowed. Default is **pe**.

Line 5: **-units** *concentration units*

-units--Identifier for concentration units. The concentration units will be used for all subsequent solutions in the data block if no column has the heading **units** (or **unit**) or if the entry for the **units** column is empty for a solution. Optionally, **unit**, **units**, or **-u[nits]**.

concentration units--Default concentration units. Three groups of concentration units are allowed, concentration (1) per liter ("/L"), (2) per kilogram solution ("/kgs"), or (3) per kilogram water ("/kgw"). All concentration units for a solution must be within the same group. Within a group, either grams or moles may be used, and prefixes milli (m) and micro (u) are acceptable. Parts per thousand, "ppt"; parts per million, "ppm"; and parts per billion, "ppb", are acceptable in the "per kilogram solution" group. Default is mmol/kgw.

Line 6: **-density** *density*

-density--Identifier for solution density. The density will be used for all subsequent solutions in the data block if no column has the heading **density** (or **dens**) or if the entry for the **density** column is empty for a solution. Density is used only if concentration units are per liter. Optionally, **dens**, **density**, or **-d[ensity]**.

density--Density of solution, kg/L. Default is 1.0 kg/L.

Line 7: **-water** *mass*

-water--Identifier for mass of water. The mass of water will be used for all subsequent solutions in the data block if no column has the heading **water** or if the entry for the **water** column is empty for a solution. Molalities of solutes are calculated from input concentrations and the moles of solutes are determined by the mass of water in solution. Optionally, **water**, or **-w[ater]**.

mass--Mass of water in the solution (kg). Default is 1.0 kg.

Line 8: **-isotope** *name, value, [uncertainty_limit]*

-isotope--Identifier for default ratio for an isotope. The isotope ratio and uncertainty limit will be used for all subsequent solutions in the data block if no column has the same *name* as a heading or if the entry for that column is empty for a solution. Isotopes and isotope uncertainty limits can be used only in inverse modeling. Optionally, **isotope** or **-i[sotope]**.

name--Name of the isotope. The name must begin with mass number followed by an element name or element redox state that is defined through **SOLUTION_MASTER_SPECIES**.

value--Isotopic value, units are usually a ratio, permil, or percent modern carbon.

uncertainty limit--Uncertainty limit to be used in mole-balance modeling. This value is optional in the **SOLUTION** data block and alternatively the internally defined default uncertainty limit may be used or an uncertainty limit may be defined with the **-isotopes** identifier of the **INVERSE_MODELING** data block.

Line 9: **-isotope_uncertainty** *name, uncertainty_limit*

-isotope_uncertainty--Identifier for uncertainty limit in the ratio for an isotope. The uncertainty limit for the isotope ratio will be used as for all subsequent solutions in the data block if no column has the same *name* directly followed by a column headed **uncertainty** or if the entry for the **uncertainty** column is empty for a solution. Isotopes and isotope uncertainty limits are used only in inverse modeling. Optionally, **uncertainty**, **-unc[ertainty]**, **uncertainties**, **unc[ertainties]**, **isotope_uncertainty**, or **-isotope_[uncertainty]**.

name--Name of the isotope, beginning with mass number.

uncertainty_limit--Uncertainty limit for the isotope to be used in inverse modeling.

Line 10: *column headings*

column headings--Column headings are element names, element valence-state names (element chemical symbol followed by valence state in parentheses), isotope names (element chemical symbol preceded by the mass number), one of the identifiers in Lines 1-7 (without the hyphen), **number**, **description**, or **uncertainty**. Most often the headings are equivalent to the first data item of line 7 of the **SOLUTION** data block. A column heading "**number**" is used to specify solution numbers or range of solution numbers that are specified following the keyword in the **SOLUTION** data block. Similarly, a column heading "**description**" allows the entry of the descriptive information that is entered following the keyword and solution number in the **SOLUTION** data block. A column headed "**uncertainty**" may be entered adjacent to the right of any isotope column to define uncertainty limits for isotope data in inverse modeling. One and only one line of headings must be entered.

Line 11: [*subheadings*]

subheadings--Subheadings are used to specify element-specific units, redox couples, and concentration-determining phases. Anything entered following the second data item of line 7 of the **SOLUTION** data block can be entered on this line, including **as**, **gfw**, redox couple, or phase name and saturation index. Tabs, not spaces, must delimit the columns; data within a column must be space delimited. Subheadings are optional. At most one line of subheadings can be entered directly following the line of headings and it is identified as a line in which all fields begin with a character.

Line 12: *chemical data*

chemical data--Analytical data, one line for each solution. For most columns, the data are equivalent to the second data item of line 7 of the **SOLUTION** data block. Tabs, not spaces, must delimit the columns. Solution numbers or ranges of numbers are defined in a column with the heading **number**; default numbering begins sequentially from 1 or sequentially from the largest solution number that has been defined by any **SOLUTION**, **SOLUTION_SPREAD**, or **SAVE** data block in this or any previous simulation. Descriptive information can be entered in a column with the heading **description**. One Line 12 is needed for each solution.

Notes

SOLUTION_SPREAD is a complete equivalent to the **SOLUTION** data block that allows data entry in a tabular or spreadsheet format. In general, column headings are elements or element valence states and succeeding lines are the data values for each solution, with one solution defined on each line. Read the documentation for

SOLUTION for detailed descriptions of input capabilities to convert mass units to mole units, to change default redox calculations, and to adjust concentrations to obtain equilibrium with a specified phase. This information is entered as a subheading in **SOLUTION_SPREAD**. The identifiers of **SOLUTION** are included in **SOLUTION_SPREAD**, but in **SOLUTION_SPREAD**, the values defined for the identifiers apply to all subsequently defined solutions. Identifiers can precede or follow data lines (Line 12), and will apply to any subsequently defined solutions until the end of the data block or until the identifier is redefined. In the example data block, the pH of solutions 10-11 is defined to be 6.9 by an entry in the **pH** column; the pH for solutions 1 and 5 is the default defined by **-pH** identifier, 7.1. Empty entries in columns with headings that are not identifiers are interpreted as zero concentrations or missing values. If a column heading can not be interpreted as part of the solution input, warnings are printed and the data for that column are ignored.

Example problems

The keyword **SOLUTION_SPREAD** is used in example problem 16.

Related keywords

SOLUTION.

SURFACE

This keyword data block is used to define the amount and composition of each surface in a surface assemblage. The composition of a surface assemblage can be defined in two ways: (1) implicitly, by specifying that the surface assemblage is in equilibrium with a solution of fixed composition or (2) explicitly, by defining the amounts of the surfaces in their neutral form (for example, SurfbOH). A surface assemblage may have multiple surfaces and each surface may have multiple binding sites, which are identified by letters following an underscore.

Example data block 1

```
Line 0a: SURFACE 1 Surface in equilibrium with solution 10
Line 1a:        -equilibrate with solution 10
Line 2a:        Surfa_w   1.0     1000.     0.33
Line 2b:        Surfa_s   0.01
Line 2c:        Surfb     0.5     1000.     0.33
Line 0b: SURFACE 3 Sites related to pure phase and kinetic reactant
Line 1b:        -equilibrate with solution 10
Line 3a:        Surfc_wOH   Fe(OH)3(a)  equilibrium_phase 0.1     1e5
Line 3b:        Surfc_sOH   Fe(OH)3(a)  equilibrium_phase 0.001
Line 3b:        Surfd_sOH   Al(OH)3(a)  kinetic           0.001   2e4
Line 4:         -no_edl
Line 0c: SURFACE 5 Explicit calculation of diffuse layer composition
Line 1c:        -equilibrate with solution 10
Line 2d:        Surfe_w     0.5     1000.     0.33
Line 5:         -diffuse_layer    2e-8
Line 6:         -only_counter_ions
```

Explanation 1

Line 0: **SURFACE** [*number*] [*description*]

SURFACE is the keyword for the data block.

number--Positive number to designate the following surface assemblage and its composition. A range of numbers may also be given in the form *m-n*, where *m* and *n* are positive integers, *m* is less than *n*, and the two numbers are separated by a hyphen without intervening spaces. Default is 1.

description--Optional comment that describes the surface assemblage.

Line 1: **-equilibrate** *number*

-equilibrate--Indicates that the surface assemblage is defined to be in equilibrium with a given solution composition. Optionally, **equil, equilibrate**, or **-e[quilibrate]**.

number--Solution number with which the surface assemblage is to be in equilibrium. Any alphabetic characters following the identifier and preceding an integer ("with solution" in line 1a) are ignored.

Line 2: *surface binding-site name, sites, specific_area_per_gram, mass*

surface binding-site name--Name of a surface binding site.

sites--Total number of sites for this binding site, in moles.

specific_area_per_gram--Specific area of surface, in m^2/g. Default is 600 m^2/g.

mass--Mass of solid for calculation of surface area, in g; surface area is *mass* times *specific_area_per_gram*. Default is 0 g.

Line 3: *surface binding-site formula, name,* [(**equilibrium_phase** or **kinetic_reactant**)], *sites_per_mole, specific_area_per_mole*

surface binding-site formula--Formula of surface species including stoichiometry of surface site and other elements connected with a pure phase or kinetic reactant. The formula must be charge balanced and is normally the OH-form of the surface binding site. If no elements other than the surface site are included in the formula, then the surface site must be uncharged. If elements are included in the formula, then these elements must be present in the pure phase or kinetic reactant.

name--Name of the pure phase or kinetic reactant that has this kind of surface site. If *name* is the name of a phase, the moles of the phase in the **EQUILIBRIUM_PHASES** data block with the same number as this surface number (3, in the example data block) will be used to determine the number of moles of surface sites (moles of phase times *sites_per_mole* equals moles of surface sites). If *name* is the rate name for a kinetic reactant, the moles of the reactant in the **KINETICS** data block with the same number as this surface number (3, in the example data block) will be used to determine the number of surface sites (moles of kinetic reactant times *sites_per_mole* equals moles of surface sites). Note the stoichiometry of the phase or reactant must contain sufficient amounts of the elements in the surface complexes defined in Line 3. In the example data block, there must be at least 0.101 mol of oxygen and hydrogen per mole of Fe(OH)3(a).

equilibrium_phase or **kinetic_reactant**--If **equilibrium_phase** is used, the *name* on the line is a phase defined in an **EQUILIBRIUM_PHASES** data block. If **kinetic_reactant** is used, the name on the line is the rate name for a kinetic reactant defined in a **KINETICS** data block. Default is **equilibrium_phase**. Optionally, **e** or **k**, only the first letter is checked.

sites_per_mole--Moles of this surface sites per mole of phase or kinetic reactant, unitless (mol/mol).

specific_area_per_mole--Specific area of surface, in m^2/mol of equilibrium phase or kinetic reactant. Default is 0 m^2/mol.

Line 4: **-no_edl**

-no_edl--Indicates that no electrostatic terms will be used in the mass-action equations for surface species and no charge-balance equations for the surfaces will be used. The identifiers **-no_edl** and **-diffuse_layer** are mutually exclusive and apply to all surfaces in the surface assemblage. Optionally, **no_edl**, **-n[o_edl]**, **no_electrostatic**, **-n[o_electrostatic]**.

Line 5: **-diffuse_layer** [*thickness*]

-diffuse_layer--Indicates that the composition of the diffuse layer will be estimated, such that, the net surface charge plus the net charge in the diffuse layer will sum to zero. See notes following the example data block. The identifiers **-diffuse_layer** and **-no_edl** are mutually exclusive and apply to all surfaces in the surface assemblage. Optionally, **diffuse_layer** or **-d[iffuse_layer]**.

thickness--Thickness of the diffuse layer in meters. Default is 10^{-8} m (equals 100 Angstrom).

Line 6: **-only_counter_ions**

-only_counter_ions--Indicates that the surface charge will be counterbalanced in the diffuse layer with counter-ions only (the sign of charge of counter-ions is opposite to the surface charge). Charge balance by co-ion exclusion is neglected (co-ions have the same sign of charge as the surface). See notes following the example. The identifier **-only_counter_ions** only applies when the **-diffuse_layer** identifier is used and applies to all surfaces in the surface assemblage. Optionally, **only_counter_ions** or **-o[nly_counter_ions]**.

Notes 1

The default databases contain thermodynamic data for a surface named "Hfo" (Hydrous ferric oxide) that are derived from Dzombak and Morel (1990). Two sites are defined in the databases: a strong binding site, Hfo_s, and a weak binding site Hfo_w. Note that Dzombak and Morel (1990) used 0.2 mol weak sites and 0.005 mol strong sites per mol Fe, a surface area of 5.33e4 m^2/mol Fe, and a gram-formula weight of 89 g Hfo/mol Fe; to be consistent with their model, the relative number of strong and weak sites should remain constant as the total number of sites varies.

The order of lines 1, 2, 3, 4, and 5 is not important. Lines 1 and, optionally, 4, 5, or 6 should occur only once within the keyword data block. Lines 2 and 3 may be repeated to define the amounts of all binding sites for all surfaces.

Lines 1a, 1b, and 1c require the program to make three calculations to determine the composition of each of the surface assemblages, termed "initial surface-composition calculations". Before any batch-reaction or transport calculations, three initial surface-composition calculations will be performed to determine the composition of the surface assemblages that would exist in equilibrium with the specified solution (solution 10 for all three surface assemblages in this example data block). The composition of the solution will not change during these calculations. In contrast, during a batch-reaction calculation, when a surface assemblage (defined as in example data block 1 or example data block 2 of this section) is placed in contact with a solution with which it is not in equilibrium, both the surface composition and the solution composition will adjust to reach a new equilibrium.

SURFACE 1 has two surfaces, Surfa and Surfb. Surfa has two binding sites, Surfa_w and Surfa_s; the surface area and mass for Surfa must be defined in the input data for at least one of the two binding sites. Surfb has only one kind of binding site and the area and mass must be defined as part of the input for the single binding site.

SURFACE 3 has one surface, Surfc, which has two binding sites, Surfc_w and Surfc_s. The number of binding sites for these two kinds of sites is determined by the amount of $Fe(OH)_3(a)$ in **EQUILIBRIUM_PHASES** 3, where 3 is the same number as the surface number. If *m* represents the moles of $Fe(OH)_3(a)$ in **EQUILIBRIUM_PHASES** 3, then the number of sites of Surfc_w is $0.1m$ (mol) and of Surfc_s is $0.001m$ (mol). The surface area for Surfc is defined relative to the moles of $Fe(OH)_3(a)$, such that the surface area is $100{,}000m$ (m^2). During batch-reaction simulations the moles of $Fe(OH)_3(a)$ in **EQUILIBRIUM_PHASES** 3 may change, in which case the number of sites of Surfc will change as will the surface area associated with Surfc. Whenever Fe(OH)3(a) precipitates, the specified amounts of Surfc_wOH and Surfc_sOH are formed. These formulas are charge balanced and the OH groups are part of the formula for Fe(OH)3(a). The OH is not used in the initial surface-composition calculation, but is critical when amounts of Fe(OH)3(a) vary. Erroneous results will occur if the formula is not charge balanced; an error message will result if the elements in the surface complex (other than the surface site itself) are not contained in sufficient quantities in the equilibrium phase or kinetic formula.

The number of sites of Surfd in the example data block is determined by the amount of a kinetic reactant defined in **KINETICS** 3, where 3 is the same number as the surface number. Sites related to a kinetic reactant are exactly analogous to sites related to an equilibrium phase. The same restrictions apply--the formula must be charge balanced and the elements in the surface complex (other than the surface site itself) must be included in the formula of the reactant.

When **-diffuse_layer** is not used (default), to account for the charge that develops on the surface, an equal, but opposite, charge imbalance is attributed to the solution. Thus, charge imbalances accumulate in the solution and on the surface when surfaces and solutions are separated. This handling of charge imbalances for surfaces is physically incorrect. Consider the following, where a charge-balanced surface is brought together with a charge-balanced solution. Assume a positive charge develops at the surface. Now remove the surface from the solution. With the present formulation, a positive charge imbalance is associated with the surface, Z_s, and a negative charge imbalance, Z_{soln}, is associated with the solution. In reality, the charged surface plus the diffuse layer surrounding it would be electrically neutral and both should be removed when the surface is removed from solution. This would leave an electrically neutral solution. The default formulation is workable; its main defect is that the counter-ions that should be in the diffuse layer are retained in the solution. The model results are adequate, provided solutions and surfaces are not separated or the exact concentrations of aqueous counter-ions are not critical to the investigation.

The **-diffuse_layer** identifier is a switch that activates a different model to account for the accumulation of surface charge. When the **-diffuse_layer** identifier is used, the composition of the diffuse layer is calculated and an additional printout of the elemental composition of the diffuse layer is produced. The moles of each aqueous species in the diffuse layer are calculated according to the method of Borkovec and Westall (1983) and the assumption that the diffuse layer is a constant thickness (optional input with **-diffuse_layer**, default is 10^{-8} m). The variation of thickness of the diffuse layer with ionic strength is ignored. The net charge in the diffuse layer exactly balances the net surface charge. Conceptually, the results of using this alternative approach are correct--charge imbalances on the surface are balanced in the diffuse layer and the solution remains charge balanced. Great uncertainties exist in the true composition of the diffuse layer and the thickness of the diffuse layer. The ion complexation in the bulk solution is assumed to apply in the diffuse layer, which is unlikely because of changes in the dielectric constant of water near the charged surface. The thickness of the diffuse layer is purely an assumption that allows the volume of water in the diffuse layer to remain small relative to the solution volume. It is possible, especially for solutions of low ionic strength, for the calculated concentration of an element to be negative in the diffuse layer. In these cases, the assumed thickness of the diffuse layer is too small or the entire diffuse-layer approach is inappropriate. However, the identifier **-only_counter_ions** offers an option to let only the counter-ions increase in concentration in the diffuse layer, and to leave the co-ions at the same concentration in the diffuse layer as in the bulk solution. The counter-ions have a higher concentration in the diffuse layer than without this option, because co-ion exclusion is neglected. The calculation of the diffuse-layer composition involves a computer intensive integration and an additional set of iterations. The **-diffuse_layer** identifier causes calculations to be 5 to 10 times slower than calculations with the default approach.

A third alternative for modeling surface-complexation reactions, in addition to the default and **-diffuse_layer**, is to ignore the surface potential entirely. The **-no_edl** identifier eliminates the potential term from mass-action expressions for surface species, eliminates any charge-balance equations for surfaces, and eliminates

any charge-potential relationships. The charge on the surface is calculated and saved with the surface composition and an equal and opposite charge is stored with the aqueous phase. All of the cautions about separation of charge, mentioned in the previous paragraphs, apply to the calculation using **-no_edl**.

For transport calculations, it is much faster in terms of cpu time to use either the default (no explicit diffuse layer calculation) or **-no_edl**. However, **-diffuse_layer** can be used to test the sensitivity of the results to diffuse-layer effects. All solutions should be charge balanced for transport calculations.

Example data block 2

```
Line 0d:  SURFACE 1 Neutral surface composition
Line 7a:      Surf_wOH     0.3      660.      0.25
Line 7b:      Surf_sOH     0.003
Line 3a:      Surfc_sOH    Fe(OH)3(a)  equilibrium_phase 0.001
Line 3b:      Surfd_sOH    Al(OH)3(a)  kinetic           0.001
```

Explanation 2

Line 0d: **SURFACE** [*number*] [*description*]

Same as example data block 1.

Line 7: *surface binding-site formula, sites, specific_area_per_gram, mass*

surface binding-site formula--Formula of the surface binding site in its OH form, Surf_sOH and Surf_wOH in this example data block. It is important to include the OH in the formula or hydrogen and oxygen will be extracted from the solution during the reaction step, which will cause unexpected redox or pH reactions.

sites--Total number of sites for this binding site, in moles.

specific_area_per_gram--Specific area of surface, in m^2/g. Default is 600 m^2/g.

mass--Mass of solid for calculation of surface area, in g; surface area is *mass* times *specific_area_per_gram*. Default is 0 g.

Line 3: *surface binding-site formula, name,* [(**equilibrium_phase** or **kinetic_reactant**)], *sites_per_mole, specific_area_per_mole.* Same as example data block 1.

Notes 2

The difference between example data block 2 and example data block 1 is that no initial surface-composition calculation is performed in example data block 2; the initial states of the surfaces are defined to be in their OH form and not in equilibrium with any solution. Additional surfaces and binding sites can be defined by repeating lines 7 or 3. The **-diffuse_layer, -only_counter_ions**, or **-no_edl** identifier can also be included.

After a set of batch-reaction calculations has been simulated, it is possible to save the resulting surface composition with the **SAVE** keyword. If the new composition is not saved, the surface composition will remain the same as it was initially defined before the batch-reaction calculations. After it has been defined or saved, the surface composition may be used in subsequent simulations through the **USE** keyword. In **ADVECTION** and **TRANSPORT** simulations, the surface assemblages in the column are automatically updated after each shift.

Example problems

The keyword **SURFACE** is used in example problems 8 and 14.

Related keywords

ADVECTION, SURFACE_MASTER_SPECIES, SURFACE_SPECIES, SAVE surface, TRANSPORT, and **USE surface.**

SURFACE_MASTER_SPECIES

This keyword data block is used to define the correspondence between surface binding-site names and surface master species. Normally, this data block is included in the database file and only additions and modifications are included in the input file. The default databases contain master species for Hfo_s and Hfo_w, which represent the weak and strong binding sites of hydrous ferric oxides (Dzombak and Morel, 1990).

Example data block

```
Line 0:  SURFACE_MASTER_SPECIES
Line 1a:       Surf_s       Surf_sOH
Line 1b:       Surf_w       Surf_wOH
```

Explanation

Line 0: **SURFACE_MASTER_SPECIES**

Keyword for the data block. No other data are input on the keyword line.

Line 1: *surface binding-site name, surface master species*

surface binding-site name--Name of a surface binding site. It must begin with a capital letter, followed by zero or more lower case letters. Underscores ("_") plus one or more lower case letters are used to differentiate types of binding sites on a single surface. Multiple binding sites can be defined for each surface.

surface master species--Formula for the surface master species, usually the OH-form of the binding site.

Notes

In this example data block, a surface named "Surf" has a strong and a weak binding site. Association reactions must be defined with **SURFACE_SPECIES** for the master species associated with each binding site and for any additional surface complexation species. All reactions for the binding sites of a surface (Surf_s and Surf_w, in this example data block) must be written in terms of the surface master species (Surf_sOH and Surf_wOH in this example data block). Each surface master species must be defined by an identity reaction with log K of 0.0 in **SURFACE_SPECIES** input. The number of sites, in moles, for each binding site must be defined in the **SURFACE** data block. Information defining the surface area is also specified with the **SURFACE** data block. In setting up the equations for a simulation that includes multiple binding sites, one mole-balance equation is included for each binding site for each surface and one charge-balance equation is included for each surface (including all of its binding sites).

Example problems

The keyword **SURFACE_MASTER_SPECIES** is not used in the example problems. See the listing of the default database file in Attachment B for examples.

Related keywords

SURFACE and **SURFACE_SPECIES**.

SURFACE_SPECIES

This keyword data block is used to define a reaction and log *K* for each surface species, including surface master species. Normally, this data block is included in the database file and only additions and modifications are included in the input file. Surface species defined in Dzombak and Morel (1990) are defined in the default databases; the master species are Hfo_w and Hfo_s for the weak and strong binding sites of hydrous ferric oxide.

Example data block

```
Line 0:  SURFACE_SPECIES
Line 1a: Surf_sOH = Surf_sOH
Line 2a        log_k      0.0
Line 1b: Surf_sOH + H+ = Surf_sOH2+
Line 2b:       log_k      6.3
Line 1c: Surf_wOH = Surf_wOH
Line 2c        log_k      0.0
Line 1d: Surf_wOH + H+ = Surf_wOH2+
Line 2d:       log_k      4.3
```

Explanation

Line 0: **SURFACE_SPECIES**

Keyword for the data block. No other data are input on the keyword line.

Line 1: *Association reaction*

Association reaction for surface species. The defined species must be the first species to the right of the equal sign. The association reaction must precede all identifiers related to the surface species. Line 1a is a master-species identity reaction.

Line 2: **log_k** *log K*

log_k--Identifier for log *K* at 25°C. Optionally, **-log_k**, **logk**, **-l[og_k]**, or **-l[ogk]**.

log K--Log *K* at 25°C for the reaction. Log *K* for a master species is 0.0. Default is 0.0.

Notes

This example data block assumes that Surf_w and Surf_s are defined in a **SURFACE_MASTER_SPECIES** data block. Lines 1 and 2 may be repeated as necessary to define all of the surface reactions. An identity reaction is needed to define each master surface species, lines 1a and 1c in this example data block. The log *K* for the identity reaction must be 0.0, lines 2a and 2c in this example data block. This example data block assumes that Surf_w and Surf_s are defined in a **SURFACE_MASTER_SPECIES** data block.

An underscore plus one or more lowercase letters is used to define different binding sites for the same surface. In the example data block, association reactions for a strong and a weak binding site are defined for the surface named "Surf". Multiple surfaces may be defined simply by defining multiple master surface species (for example, Surfa, Surfb, and Surfc). Multiple binding sites can be defined for each surface by using an underscore followed by one or more lower case letters. Association reactions for each surface and binding site must be defined with **SURFACE_SPECIES** input.

Temperature dependence of log K can be defined with enthalpy of reaction (identifier **delta_h**) and the van't Hoff equation or with an analytical expression (**-analytical_expression**). See **SOLUTION_SPECIES** or **PHASES** for examples.

The identifier **-no_check** can be used to disable checking charge and elemental balances (see **SOLUTION_SPECIES**). The use of **-no_check** is not recommended. If **-no_check** is used, then the **-mole_balance** identifier is needed to ensure the correct stoichiometry for the surface species. In PHREEQC version 1, the **-no_check** option was included to permit the stoichiometry of a species to be defined separately from the mass-action equation. Specifically, the sorption of uranium on iron oxides as described by Waite and others (1994) provides an example, where they use different coefficients in the mass-action equation than the mole-balance equations. However, activity of a surface species is defined as mole fraction of sites occupied by the species in PHREEQC version 2 which is inconsistent with activity that is defined as molality by Waite and others (1994) and PHREEQC version 1. It is noted that formulas with coefficients of only 1 in the mass-action-equation will give identical results for PHREEQC version 1 and 2. The **-no_check** and **-mole_balance** identifiers have been retained in version 2, but its use should be restricted to special sorption formulas.

Example problems

The keyword **SURFACE_SPECIES** is used in example problems 8 and 14. See the listing of the default database file in Attachment B for additional examples.

Related keywords

PHASES, SOLUTION_SPECIES, SURFACE, and **SURFACE_MASTER_SPECIES.**

TITLE

This keyword data block is used to include a comment for a simulation in the output file. The comment will appear in the echo of the input data and it will appear at the beginning of the simulation calculations.

Example data block

```
Line 0:  TITLE The title may begin on this line,
Line 1a: or on this line.
Line 1b: It continues until a keyword is found at the beginning of a line
Line 1c: or until the end of the file.
```

Explanation

Line 0: **TITLE** *comment*

TITLE is the keyword for the data block. Optionally, **COMMENT**.

comment--The first line of a title (or comment) may begin on the same line as the keyword.

Line 1: *comment*

comment--The title (or comment) may continue on as many lines as necessary. Lines are read and saved as part of the title until a keyword begins a line or until the end of the input file.

Notes

Be careful not to begin a line of the title with a keyword because that signals the end of the **TITLE** data block. The **TITLE** data block is intended to document a simulation in the output file. If more than one title keyword is entered for a simulation, each will appear in the output file as part of the echo of the input data, but only the last will also appear at the beginning of the simulation calculations. The characters "#" and ";" have special meanings in PHREEQC input files; in the **TITLE** data block, the "#" will cause the remainder of the line to be excluded from *comment* and ";" will have the same effect as a line break at that character position. Lines that are entirely white space (tabs and spaces) and comments (characters following "#") are eliminated.

Example problems

The keyword **TITLE** is used in all examples, 1 through 18.

TRANSPORT

This keyword data block is used to simulate 1D transport processes that include advection and dispersion, diffusion, and diffusion into stagnant zones adjacent to the 1D flow system. All chemical processes modeled by PHREEQC, including kinetically controlled reactions, may be included in an advective-dispersive transport simulation. Purely advective transport plus reactions--without diffusion, dispersion, or stagnant zones--can be simulated with the **ADVECTION** data block.

Example data block

```
Line 0: TRANSPORT
Line 1:     -cells                  5
Line 2:     -shifts                 25
Line 3:     -time_step              3.15e7
Line 4:     -flow_direction         forward
Line 5:     -boundary_conditions    flux constant
Line 6:     -lengths                4*1.0 2.0
Line 7:     -dispersivities         4*0.1 0.2
Line 8:     -correct_disp           true
Line 9:     -diffusion_coefficient  1.0e-9
Line 10:    -stagnant               1  6.8e-6   0.3   0.1
Line 11:    -thermal_diffusion      3.0   0.5e-6
Line 12:    -initial_time           1000
Line 13:    -print_cells            1-3 5
Line 14:    -print_frequency        5
Line 15:    -punch_cells            2-5
Line 16:    -punch_frequency        5
Line 17:    -dump                   dump.file
Line 18:    -dump_frequency         10
Line 19:    -dump_restart           20
Line 20:    -warnings               false
```

Explanation

Line 0: **TRANSPORT**

TRANSPORT is the keyword for the data block. No other data are input on the keyword line.

Line 1: **-cells** *cells*

-cells--Indicates that the number of cells in the 1D column will be given. Optionally, **cells** or **-c[ells]**.

cells--Number of cells in a 1D column to be used in the advective-dispersive transport simulation. Default is 0.

Line 2: **-shifts** *shifts*

-shifts--Indicates that the number of shifts or diffusion periods in the advective-dispersive transport simulation will be given. Optionally, **shifts** or **-s[hifts]**.

shifts--For advective-dispersive transport, *shifts* is the number of advective shifts or time steps, which is the number of times the solution in each cell will be shifted to the next higher or lower numbered cell; the total time simulated is $shifts \times time\ step$. For purely diffusive transport, *shifts* is

the number of diffusion periods that are simulated; the total diffusion time is *shifts* × *time step*. Default is 1.

Line 3: **-time_step** *time step*

-time_step--Defines time step associated with each advective shift or diffusion period. The number of shifts or diffusion periods is given by **-shifts**. Optionally, **timest**, **-t[imest]**, **time_step**, or **-t[ime_step]**.

time step--Time, in seconds, associated with each shift or diffusion period. Default is 0.

Line 4: **-flow_direction (forward**, **back**, or **diffusion_only)**

-flow_direction--Defines direction of flow. Optionally, **direction**, **flow**, **flow_direction**, **-dir[ection]**, or **-f[low_direction]**.

forward, **back**, or **diffusion_only**--(1) **Forward**, advective flow direction is forward; optionally, **f[orward]**, (2) **Backward**, advective flow direction is backward; optionally **b[ackward]**, or (3) **Diffusion_only**, only diffusion occurs, there is no advective flow; optionally **d[iffusion_only]** or **n[o_flow]**. Default is **forward**.

Line 5: **-boundary_conditions** *first, last*

-boundary_conditions--Defines boundary conditions for the first and last cell. Optionally, **bc**, **bcond**, **-b[cond]**, **boundary_condition**, **-b[oundary_condition]**. Three types of boundary conditions are allowed at either end of the column (indicated by x_{end}):

constant--Concentration is constant $C(x_{end}, t) = C_0$, also known as first type or Dirichlet boundary condition. Optionally, **co[nstant]** or **1**.

closed--No flux at boundary, $\frac{\partial C(x_{end}, t)}{\partial x} = 0$, also known as second type or Neumann boundary condition. Optionally, **cl[osed]** or **2**.

flux--Flux boundary condition, $C(x_{end}, t) = C_0 + \frac{D_L \partial C(x_{end}, t)}{v \partial x}$, also known as third type or Cauchy boundary condition. Optionally, **f[lux]** or **3**.

first--Boundary condition at the first cell, **constant**, **closed**, or **flux**. Default is **flux**.

last--Boundary condition at the last cell, **constant**, **closed**, or **flux**. Default is **flux**.

Line 6: **-lengths** *list of lengths*

-lengths--Defines length of each cell for advective-dispersive transport simulations (m). Optionally, **length**, **lengths**, or **-l[engths]**.

list of lengths--Length of each cell (m). Any number of *lengths* up to the total number of cells (*cells*) may be entered. If *cells* is greater than the number of *lengths* entered, the final value read will be used for the remaining cells. Multiple lines may be used. Repeat factors can be used to input multiple data with the same value; in the example data block, 4*1.0 is interpreted as 4 values of 1.0. Default is 1.

Line 7: **-dispersivities** *list of dispersivities*

-dispersivities--Defines dispersivity of each cell for advective-dispersive transport simulations (m). Optionally, **disp**, **dispersivity**, **dispersivities**, **-dis[persivity]**, or **-dis[persivities]**.

list of dispersivities--Dispersivity assigned to each cell (m). Any number of *dispersivities* up to the total number of cells (*cells*) may be entered. If *cells* is greater than the number of *dispersivities*

entered, the final value read will be used for the remaining cells. Multiple lines may be used. Repeat factors can be used to input multiple data with the same value; in the example data block, 4*0.1 is interpreted as 4 values of 0.1. Default is 0.

Line 8: **-correct_disp** [(*True* or *False*)]

When *true*, dispersivity is multiplied with (1 + 1/*cells*) for column ends with flux boundary conditions. This correction is recommended when modeling effluent composition from column experiments. Optionally, **correct_disp** or **-co[rrect_disp]**. Default is **True**, value at beginning of run is **False**.

Line 9: **-diffusion_coefficient** *diffusion coefficient*

-diffusion_coefficient--Defines diffusion coefficient for all aqueous species (m^2/s). Optionally, **diffusion_coefficient**, **diffc**, **-dif[fusion_coefficient]**, or **-dif[fc]**.

diffusion coefficient--Diffusion coefficient. Default is 0.3e-9 m^2/s.

Line 10: **-stagnant** *stagnant_cells* [*exchange_factor* θ_m θ_{im}]

-stagnant--Defines the maximum number of stagnant (immobile) cells associated with each cell in which advection occurs (mobile cell). The immobile cells are usually defined to be a 1D column that is connected to the mobile cell; however, the connections among the immobile cells may be defined arbitrarily with **MIX** data blocks. The immobile cells associated with a mobile cell, *cell*, are numbered as follows: $n \times cells + 1 + cell$, where *cells* is number of mobile cells and $1 \le n \le stagnant\_cells$. Each immobile cell that is used must have a defined solution (**SOLUTION**, **SOLUTION_SPREAD**, or **SAVE** data block) and either a **MIX** data block must be defined or, for the first-order exchange model, the *exchange_factor* must be defined (only applicable if *stagnant_cells* equals 1). Mixing will be performed at each diffusion/dispersion time step. **EQUILIBRIUM_PHASES**, **EXCHANGE**, **GAS_PHASE**, **KINETICS**, **REACTION**, **REACTION_TEMPERATURE**, **SOLID_SOLUTIONS**, and **SURFACE** may be defined for an immobile cell. Thermal diffusion in excess of hydrodynamic diffusion can only be calculated for the first-order exchange model. Optionally, **stagnant** or **-st[agnant]**.

stagnant_cells--Number of stagnant (immobile) cells associated with each mobile cell. Default is 0.

exchange_factor--Factor describing exchange between mobile and immobile cells (s^{-1}). The *exchange_factor* is used only if *stagnant_cells* is 1 and all immobile cells have the same properties. WARNING: If *exchange_factor* is entered, all previously defined **MIX** structures will be deleted and **MIX** structures for the first order exchange model for a dual porosity medium will be created. Default is 0.

θ_m--Porosity in each mobile cell. The θ_m is used only if *stagnant_cells* is 1 and all immobile cells have the same properties. Default is 0.

θ_{im}--Porosity in each immobile cell. The θ_{im} is used only if *stagnant_cells* is 1 and all immobile cells have the same properties. Default is 0.

Line 11: **-thermal_diffusion** *temperature retardation factor, thermal diffusion coefficient*

-thermal_diffusion--Defines parameters for calculating the diffusive part of heat transport. Diffusive heat transport will be calculated as a separate process if the temperature in any of the solutions of the transport domain differs by more than 1°C, and when the *thermal diffusion coefficient* is larger than the effective (aqueous) *diffusion coefficient*. Otherwise, diffusive heat transport is calculated

as a part of aqueous diffusion. The *temperature retardation factor* is defined as the ratio of the heat capacity of the total aquifer over the heat capacity of water in the pores, and equals $R_T = 1 + \frac{(1-\theta)\,\rho_s k_s}{\theta\,\rho_w k_w}$, where θ is the water filled porosity, ρ is density (kg/m^3), k is specific heat (kJ°C^{-1}kg^{-1}), and subscripts w and s indicate water and solid, respectively. The thermal diffusion coefficient can be estimated using $\kappa_e = \frac{\kappa_t}{\theta \rho_w k_w}$, where κ_t is the heat conductivity of the aquifer, including pore water and solid (kJ°C^{-1}m^{-1}s^{-1}). The value of κ_e may be 100-1500 times larger than the aqueous diffusion coefficient, or about 1e-6 m^2/s. A temperature change during transport is reduced by the temperature retardation factor (-) to account for the heat capacity of the matrix. Optionally, **-th[ermal_diffusion]**.

retardation factor--Temperature retardation factor, unitless. Default is 2.0.

thermal diffusion coefficient--Thermal diffusion coefficient. Default is the aqueous diffusion coefficient.

Line 12: **-initial_time** *initial_time*

-initial_time--Identifier to set the time at the beginning of a transport simulation. The identifier sets the initial value of the variable controlled by **-time** in the **SELECTED_OUTPUT** data block. Optionally, **initial_time** or **-i[nitial_time]**.

initial_time--Time (seconds) at the beginning of the transport simulation. Default is the cumulative time including all preceding **ADVECTION** simulations for which **-time_step** has been defined and all preceding **TRANSPORT** simulations.

Line 13: **-print_cells** *list of cell numbers*

-print_cells--Identifier to select cells for which results will be written to the output file. Optionally, **print**, **print_cells**, or **-pr[int_cells]**. Note the hyphen is required to avoid a conflict with the keyword **PRINT**.

list of cell numbers--Printing to the output file will occur only for these cell numbers. The list of cell numbers may be continued on the succeeding line(s). A range of cell numbers may be included in the list in the form *m-n*, where *m* and *n* are positive integers, *m* is less than *n*, and the two numbers are separated by a hyphen without intervening spaces. Default 1-*cells*.

Line 14: **-print_frequency** *print_modulus*

-print_frequency--Identifier to select shifts for which results will be written to the output file. Optionally, **print_frequency**, **-print_f[requency]**, **output_frequency**, or **-o[utput_frequency]**.

print_modulus--Printing to the output file will occur after every *print_modulus* advection shifts or diffusion periods. Default is 1.

Line 15: **-punch_cells** *list of cell numbers*

-punch_cells--Identifier to select cells for which results will be written to the selected-output file. Optionally, **punch**, **punch_cells**, **-pu[nch_cells]**, **selected_cells**, or **-selected_c[ells]**.

list of cell numbers--Printing to the selected-output file will occur only for these cell numbers. The list of cell numbers may be continued on the succeeding line(s). A range of cell numbers may be included in the list in the form *m-n*, where *m* and *n* are positive integers, *m* is less than *n*, and the two numbers are separated by a hyphen without intervening spaces. Default 1-*cells*.

Line 16: **-punch_frequency** *punch_modulus*

-punch_frequency--Identifier to select shifts for which results will be written to the selected-output file. Optionally, **punch_frequency**, **-punch_f[requency]**, **selected_output_frequency**, **-selected_o[utput_frequency]**.

punch_modulus--Printing to the selected-output file will occur after every *punch_modulus* advection shifts or diffusion periods. Default is 1.

Line 17: **-dump** *dump file*

-dump--Identifier to write complete state of a advective-dispersive transport simulation after every *dump_modulus* advection shifts or diffusion periods. The file is formatted as an input file that can be used to restart calculations. Optionally, **dump** or **-du[mp]**.

dump file--Name of file to which complete state of advective-dispersive transport simulation will be written. Default is *phreeqc.dmp*.

Line 18: **-dump_frequency** *dump_modulus*

-dump_frequency--Complete state of a advective-dispersive transport simulation will be written to dump file after *dump_modulus* advection shifts or diffusion periods. Optionally, **dump_frequency** or **-dump_f[requency]**.

dump_modulus--Complete state will be printed after *dump_modulus* advection shifts or diffusion periods. Default is *shifts*/2 or 1, whichever is larger.

Line 19: **-dump_restart** *shift number*

-dump_restart--If an advective-dispersive transport simulation is restarted from a dump file, the starting shift number is given on this line. Optionally, **dump_restart** or **-dump_r[estart]**.

shift number--Starting shift number for the calculations, if restarting from a dump file. The shift number is written in the dump file by PHREEQC. It equals the shift number at which the dump file was created. Default is 1.

Line 20: **-warnings** [(*True* or *False*)]

-warnings--Identifier enables or disables printing of warning messages for transport calculations. In some cases, transport calculations could produce many warnings that are not errors. Once it is determined that the warnings are not due to erroneous input, disabling the warning messages can avoid generating large output files. Optionally, **warnings**, **warning**, or **-w[arnings]**.

[(*True* or *False*)]--If value is true, warning messages are printed to the screen and the output file; if value is false, warning messages are not printed to the screen or the output file. The value set with **-warnings** is retained in all subsequent transport simulations until changed. Default is **True**, value at beginning of run is **True**.

Notes

The advective-dispersive transport capabilities of PHREEQC are derived from a formulation of 1D, advective-dispersive transport presented by Appelo and Postma (1993). The 1D column is defined by a series of cells (number of cells is *cells*), each of which has the same pore volume. Lengths are defined for each cell and the time step (*time step*) gives the time necessary for a pore volume of water to move through each cell. Thus, the velocity of water in each cell is determined by the length of the cell divided by the time step. In the example data block, a column

of five cells (*cells*) is modeled and 5 pore volumes of filling solution are moved through the column (*shifts/cells* is 5). The total time of the simulation is 25 years (*shifts* × *time step*). The total length of the column is 6 m (four 1-m cells and one 2-m cell).

At each shift, advection is simulated by moving solution *cells-1* to cell *cells*, solution *cells-2* to cell *cells-1*, and so on, until solution 0 is moved to cell 1 (upwind scheme). With flux-type boundary conditions, the dispersion steps follow the advective shift. With Dirichlet boundary conditions, the dispersion step and the advective shift are alternated. After each advective shift and dispersion step, kinetic reactions and chemical equilibria are calculated. The moles of pure phases and the compositions of the exchange assemblage, surface assemblage, gas phase, solid-solution assemblage, and kinetic reactants in each cell are updated after each chemical equilibration.

The **-time_step** identifier defines the length of time associated with each advective shift or diffusion period. This time step may be subdivided into smaller dispersion time steps if necessary to calculate dispersion accurately. Each dispersion time step may be further subdivided to integrate the kinetic reactions (**KINETICS** data block). Kinetic reactions are likely to slow the calculations by up to a factor of six or more compared to pure equilibrium calculations.

The numerical scheme is for cell-centered concentrations, which has consequences for data interpretation. Thus, the composition in a boundary cell is a half-cell distance away from the column outlet and needs a half time step to arrive at (or from) the column end. The half time step must be added to the total residence time in the column when effluent from a column is simulated [use (TOTAL_TIME + *time step*/2) for time, see example 15, or (STEP_NO + 0.5)/*cells*) for pore volumes, see example 11]. The kinetics time for advective transport into the boundary cell is the advective time step divided by 2. Also, the cell-centered scheme does not account for dispersion in the border half-cell in case of a flux boundary condition. The identifier **-correct_disp** provides an option to correct the ignored dispersion, by increasing the dispersivity for all cells in the column by the appropriate amount. The correction will improve the comparison with analytical solutions for conservative elements when the number of cells is small.

It has been shown in the section "Transport in Dual Porosity Media" that a "dual porosity" model, in which part of the porosity allows advective flow and part of the porosity is accessible only by diffusion, can be developed with a first-order exchange model and with finite-differences, and both approaches can be defined in terms of a mixing among cells. With the **TRANSPORT** data block, one column of mobile cells is used to represent the part of the flow system in which advection occurs, and then additional immobile cells connected to the mobile cells are used to represent the stagnant zone that is accessible only by diffusion. The stagnant zone can be defined to be parallel or perpendicular to the column of mobile cells or to be a combination of the two by proper definition of mixing factors in **MIX** data blocks. A shortcut is available for the classical formulation of a dual porosity medium with a first-order rate of exchange. In this case, **-stagnant** is used to define one stagnant cell for each mobile cell (*stagnant_cells* = 1), an exchange factor (*exchange_factor*) for the exchange between immobile and mobile cells, and the porosities θ_m and θ_{im} for the mobile and immobile cells. WARNING: If this shortcut method is used to define the stagnant zone, then all previously defined **MIX** structures will be deleted and **MIX** structures for first order exchange in a dual porosity medium are set up.

Thermal diffusion can be modeled for a stagnant zone with first-order exchange between mobile and immobile cells. Thermal exchange is calculated after subtracting the part that is associated with hydrodynamic diffusion (see "Transport of Heat"). PHREEQC uses the value of the *diffusion coefficient* to find the correct heat

exchange factor, and the value entered with identifier **-diffusion_coefficient** should be the same as has been used in equation 125 to calculate the exchange factor α.

Most of the information for advective-dispersive transport calculations must be entered with other keyword data blocks. Advective-dispersive transport assumes that solutions with numbers 1 through *cells* have been defined using **SOLUTION**, **SOLUTION_SPREAD**, or **SAVE** data blocks. In addition the infilling solution must be defined. If **-flow direction** is **forward**, solution 0 is the infilling solution; if **-flow_direction** is **backward**, solution *cells* + *1* is the infilling solution, if **-flow direction** is **diffusion_only**, then infilling solutions at both column ends are optional. If stagnant zones are modeled, solution compositions for the stagnant-zone cells must be defined with **SOLUTION**, **SOLUTION_SPREAD**, or **SAVE** data blocks.

Pure-phase assemblages may be defined with **EQUILIBRIUM_PHASES** or **SAVE**, with the number of the assemblage corresponding to the cell number. Likewise, an exchange assemblage, a surface assemblage, a gas phase, and a solid-solution assemblage can be defined for each cell through **EXCHANGE**, **SURFACE**, **GAS_PHASE**, **SOLID_SOLUTIONS**, or **SAVE** keywords, with the identifying number corresponding to the cell number. Kinetically controlled reactions can be defined for each cell through the **KINETICS** data block. Note that ranges of numbers can be used to define multiple solutions, exchange assemblages, surface assemblages, gas phases, solid-solution assemblages, or kinetic reactions simultaneously and that **SAVE** allows definition of a range of numbers. Constant-rate reactions can be defined for mobile or immobile cells through **REACTION** data blocks, again with the identifying number of the **REACTION** data block corresponding to the cell number. **REACTION_TEMPERATURE** data blocks can be used to specify the initial temperatures of the cells in the transport simulation. Temperatures in the cells may change during the transport simulation depending on the temperature distribution and the temperature retardation factor defined by **-temp_retardation_factor.**

By default, the composition of the solution, pure-phase assemblage, exchange assemblage, surface assemblage, gas phase, solid-solution assemblage, and kinetic reactants are printed for each cell for each shift. Use of **-print_cells** and **-print_frequency** will limit the amount of data written to the output file. If **-print_cells** has been defined then only the specified cells will be written, otherwise, all cells will be written. The identifier **-print_frequency** will restrict writing to the output file to those shifts that are evenly divisible by *print_modulus*. In the example data block, results for cells 1, 2, 3, and 5 are written to the output file after each integer pore volume (5 shifts) has passed through the column. Data written to the output file can be further limited with the keyword **PRINT** (see **-reset false**).

If a **SELECTED_OUTPUT** data block has been defined (recommended), then selected data are written to the selected-output file. Use of **-punch_cells** and **-punch_frequency** in the **TRANSPORT** data block will limit the data that are written to the selected-output file. If **-punch_cells** has been defined then only the specified cells will be written, otherwise, all cells will be written. The identifier **-punch_frequency** will restrict writing to the selected-output file to those shifts that are evenly divisible by *punch_modulus*. In the example data block, results are written to the selected-output file for cells 2, 3, 4, and 5 after each integer pore volume (5 shifts) has passed through the column.

At the end of a advective-dispersive transport simulation, all the physical and chemical data (for example, compositions of solutions, equilibrium-phase assemblages, surfaces, exchangers, solid solutions, and kinetic reactants) are automatically saved and are identified by the cell number in which they reside. These data are available for subsequent simulations within a single run. Transient conditions can be simulated by including subsequent

TRANSPORT data blocks, which may define new chemical boundary and transport conditions. Only parameters that differ from the previous advective-dispersive transport simulation need to be redefined, such as new infilling solution (**SOLUTION** 0), a change from advection to diffusion only (**-flow_direction diffusion_only**), or a change in flow direction from forward to backward (**-flow_direction backward**). All parameters not specified in the new **TRANSPORT** data block remain the same as the previous advective-dispersive transport simulation. Normally, the diffusion coefficient, lengths of cells, dispersivities, and stagnant zone definitions would remain the same through all advective-dispersive transport simulations and thus need not be redefined.

For long advective-dispersive transport calculations, it may be desirable to save intermediate states in the calculation, either because of hardware failure or because of nonconvergence of the numerical method. The **-dump_frequency** identifier allows intermediate states to be saved at intervals during the calculation. The **-dump** identifier allows the definition of a file name in which to write these intermediate states. The dump file is formatted as an input file for PHREEQC, so calculations can be resumed from the point at which the dump file was made. The **-dump_restart** identifier allows a shift number to be specified from which to restart the calculations.

Example problems

The keyword **TRANSPORT** is used in example problems 11, 12, 13, and 15.

Related keywords

ADVECTION, EQUILIBRIUM_PHASES, EXCHANGE, GAS_PHASE, KINETICS, MIX, PRINT, REACTION, REACTION_TEMPERATURE, SAVE, SELECTED_OUTPUT, SOLID_SOLUTIONS, SOLUTION, and **SURFACE.**

USE

This keyword data block is used to specify explicitly which solution, exchange assemblage, pure-phase assemblage, solid-solution assemblage, and surface assemblage are to be used in the batch-reaction calculation of a simulation. **USE** can also specify previously defined kinetically controlled reactions (**KINETICS** data block), reaction parameters (**REACTION** data block), reaction-temperature parameters (**REACTION_TEMPERATURE** data block), and mixing parameters (**MIX** data block) to be used in a batch-reaction calculation.

Example data block

```
Line 0a: USE equilibrium_phases none
Line 0b: USE exchange 2
Line 0c: USE gas_phase 3
Line 0d: USE kinetics 1
Line 0e: USE mix 1
Line 0f: USE reaction 2
Line 0g: USE reaction_temperature 1
Line 0h: USE solid_solution 6
Line 0i: USE solution 1
Line 0j: USE surface 1
```

Explanation

Line 0: **USE** *keyword*, (*number* or **none**)

USE is the keyword for the data block.

keyword--One of ten keywords, **equilibrium_phases**, **exchange**, **gas_phase**, **kinetics**, **mix**, **reaction**, **reaction_temperature**, **solid_solutions**, **solution**, or **surface**.

number--Positive integer associated with previously defined composition or reaction parameters.

none--No data of the type of the specified keyword will be used in the batch-reaction calculation.

Notes

Batch-reactions are defined by allowing a solution or mixture of solutions to come to equilibrium with one or more of the following entities: an exchange assemblage, a pure-phase assemblage, a solid-solution assemblage, a surface assemblage, or a gas phase. In addition, kinetically controlled reactions, fixed-stoichiometry reactions, and reaction temperatures can be specified for batch-reaction calculations.

Entities can be defined implicitly: a solution or mixture (**SOLUTION** or **MIX** keywords) must be defined within the simulation, then the first of each kind of entity defined in the simulation will be used to define the reaction system. Thus, the first solution (or mixture) will be brought together with the first of each of the following entities that is defined in the simulation: exchange assemblage (**EXCHANGE**), gas phase (**GAS_PHASE**), pure-phase assemblage (**EQUILIBRIUM_PHASES**), solid-solution assemblage (**SOLID_SOLUTIONS**), surface assemblage (**SURFACE**); equilibrium among these entities will be calculated and maintained. Irreversible reactions may also be added implicitly to the system, and again, the first of the following entities that is defined in the simulation is added: kinetically controlled reaction (**KINETICS**), stoichiometric reaction (**REACTION**), and reaction temperature (**REACTION_TEMPERATURE**).

Entities to be included in the system can be defined explicitly with the **USE** keyword. Any combination of **USE** *keyword number* data blocks can be used to define a system. "**USE** *keyword* **none**" can be used to eliminate an entity that was implicitly defined to be in the system. For example, if only a solution and a surface are defined in a simulation and the surface is defined to be in equilibrium with the solution, then implicitly, an additional batch-reaction calculation will be made to equilibrate the solution with the surface. Though not incorrect, the batch-reaction calculation will produce the same compositions for the solution and surface as previously defined. By including "**USE solution none**", the batch-reaction calculation will be eliminated.

The composition of the solution, exchange assemblage, solid-solution assemblage, surface assemblage, pure-phase assemblage, or gas phase can be saved after a set of batch-reaction calculations with the **SAVE** keyword.

Example problems

The keyword **USE** is used in example problems 3, 6, 7, 8, 10, and 14.

Related keywords

EQUILIBRIUM_PHASES, EXCHANGE, GAS_PHASE, KINETICS, MIX, REACTION, REACTION_TEMPERATURE, SAVE, SOLID_SOLUTIONS, SOLUTION, and **SURFACE.**

USER_PRINT

This keyword data block is used to define Basic programs that print user-defined quantities to the output file. Any Basic "PRINT" statement will write to the output file.

Example data block

```
Line 0: USER_PRINT
Line 1:      -start
Basic:  10 REM convert to ppm
Basic:  20 PRINT "Sodium:    ", MOL("Na+")* 22.99 * 1000
Basic:  30 PRINT "Magnesium: ", MOL("Mg+2")* 24.3 * 1000
Basic:  40 pairs = MOL("NaCO3-") + MOL("MgCO3")
Basic:  50 PRINT "Pairs (mol/kgw): ", pairs
Basic:  60 REM print reaction increment
Basic:  70 PRINT "Rxn incr : ", RXN
Line 2:      -end
```

Explanation

Line 0: **USER_PRINT**

USER_PRINT is the keyword for the data block. No other data are input on the keyword line.

Line 1: **-start**

-start--Indicates the start of the Basic program. Optional.

Basic: *numbered Basic statement*

numbered Basic statement--A valid Basic language statement that must be numbered. The program should contain at least one "PRINT" statement. The statements are evaluated in numerical order. Statements and functions that are available through the Basic interpreter are listed in tables 8 and 9.

Line 2: **-end**

-end--Indicates the end of the Basic program. Optional. Note the hyphen is required to avoid a conflict with the keyword **END**.

Notes

USER_PRINT allows the user to write Basic programs to make calculations and print selected results as the program is running. Results of PRINT Basic statements are written directly to the output file after each calculation. More information on the Basic interpreter is available in the description of the **RATES** keyword. All of the functions defined in tables 8 and 9 are available in **USER_PRINT** Basic programs. Writing results of **USER_PRINT** can be enabled or suspended with the **-user_print** identifier in the **PRINT** data block. The **USER_PUNCH** data block is similar to **USER_PRINT**, except PUNCH Basic statements are used to write results to the selected-output file.

Example problems

The keyword **USER_PRINT** is used in example problem 6, 10, and 12.

Related keywords

PRINT, **RATES**, **SELECTED_OUTPUT**, and **USER_PUNCH**.

USER_PUNCH

This keyword data block is used to define Basic programs that print user-defined quantities to the selected-output file. Any Basic "PUNCH" statement will write to the selected-output file.

Example data block

```
Line 0: USER_PUNCH
Line 1:      -headings Na+ Mg+2 Pairs Rxn_increment
Line 2:      -start
Basic:  10 REM convert to ppm
Basic:  20 PUNCH MOL("Na+")* 22.99 * 1000
Basic:  30 PUNCH MOL("Mg+2")* 24.3 * 1000
Basic:  40 pairs = MOL("NaCO3-") + MOL("MgCO3")
Basic:  50 PUNCH pairs
Basic:  60 REM punch reaction increment
Basic:  70 PUNCH RXN
Line 3:      -end
```

Explanation

Line 0: **USER_PUNCH**

USER_PUNCH is the keyword for the data block. No other data are input on the keyword line.

Line 1: **-headings** *list of column headings*

-headings--Headings will appear on the first line of the selected-output file. Optionally, **heading**, **headings**, or **-h[eadings]**.

list of column headings--White-space-delimited (any combination of spaces and tabs) list of column headings.

Line 2: **-start**

-start--Indicates the start of the Basic program. Optional.

Basic: *numbered Basic statement*

numbered Basic statement--A valid Basic language statement that must be numbered. The program should contain at least one "PUNCH" statement. The statements are evaluated in numerical order. Statements and functions that are available through the Basic interpreter are listed in tables 8 and 9.

Line 2: **-end**

-end--Indicates the end of the Basic program. Optional. Note the hyphen is required to avoid a conflict with the keyword **END**.

Notes

USER_PUNCH allows the user to write a Basic program to make calculations and print selected results to the selected-output file as the program is running. Results of PUNCH Basic statements are written directly to the selected-output file after each calculation. The Basic program is useful for writing results in the correct form, so that they can be plotted directly. All of the functions defined in tables 8 and 9 are available in **USER_PRINT** Basic programs. More information on the Basic interpreter is available in the description of the **RATES** keyword.

USER_PUNCH has no effect unless a **SELECTED_OUTPUT** data block has been defined. Writing results of **USER_PUNCH** can be enabled or suspended with the **-user_punch** identifier in the **SELECTED_OUTPUT** data block. If the **-selected_output** identifier in the **PRINT** data block is false, then all selected output, including **USER_PUNCH**, is disabled.

Example problems

The keyword **USER_PUNCH** is used in example problems 6C, 9, 10, 11, 12, 13, 14, and 15.

Related keywords

PRINT, **RATES**, **SELECTED_OUTPUT**, and **USER_PRINT**.

SUMMARY OF DATA INPUT

ADVECTION

ADVECTION
-cells *cells*
-shifts *shifts*
-time_step *time step*
-initial_time *initial_time*
-print_cells *list of cell numbers*
-print_frequency *print_modulus*
-punch_cells *list of cell numbers*
-punch_frequency *punch_modulus*
-warnings [(*True* or *False*)]

END

EQUILIBRIUM_PHASES

EQUILIBRIUM_PHASES [*number*] [*description*]
phase name [*saturation index* [(*alternative formula* or *alternative phase*) [*amount*]]]

EXCHANGE

EXCHANGE [*number*] [*description*]
exchange formula, amount
exchange formula, name, [(**equilibrium_phase** or **kinetic_reactant**)], *exchange_per_mole*
-equilibrate *number*

EXCHANGE_MASTER_SPECIES

EXCHANGE_MASTER_SPECIES
exchange name, exchange master species

EXCHANGE_SPECIES

EXCHANGE_SPECIES
Association reaction
log_k *log K*
delta_h *enthalpy,* [*units*]
-analytical_expression A_1, A_2, A_3, A_4, A_5
-gamma *Debye-Hückel a, Debye-Hückel b*
-davies
-no_check

-mole_balance *formula*

GAS_PHASE

Fixed-pressure gas phase

GAS_PHASE [*number*] [*description*]
-fixed_pressure
-pressure *pressure*
-volume *volume*
-temperature *temp*
phase name, partial pressure

Fixed-volume gas phase: Define initial moles of components with partial pressures

GAS_PHASE [*number*] [*description*]
-fixed_volume
-volume *volume*
-temperature *temp*
phase name, partial pressure

Fixed-volume gas phase: Define initial moles of components by equilibrium with a solution

GAS_PHASE [*number*] [*description*]
-fixed_volume
-equilibrium *number*
-volume *volume*
phase name

INCREMENTAL_REACTIONS

INCREMENTAL_REACTIONS [(*True* or *False*)]

INVERSE_MODELING

INVERSE_MODELING [*number*] [*description*]
-solutions *list of solution numbers*
-uncertainty *list of uncertainty limits*
-phases
phase name [**force**] [**dissolve** or **precipitate**] [*list of isotope name, isotope ratio, isotope uncertainty limit*]
-balances
element or valence state name, [*list of uncertainty limits*]
-isotopes
isotope_name, list of uncertainty limits
-range [*maximum*]

-minimal
-tolerance *tol*
-force_solutions *list of* (*True* or *False*)
-uncertainty_water *moles*
-mineral_water [(*True* or *False*)]

KINETICS

Explicit definition of steps

KINETICS [*number*] [*description*]
rate name
-formula *list of formula,* [*stoichiometric coefficient*]
-m *moles*
-m0 *initial moles*
-parms *list of parameters*
-tol *tolerance*
-steps *list of time steps*
-step_divide *step_divide*
-runge_kutta (**1**, **2**, **3**, or **6**)

Equal-increment definition of steps

-steps *total time* [**in** *steps*]

KNOBS

KNOBS
-iterations *iterations*
-convergence_tolerance *convergence_tolerance*
-tolerance *tolerance*
-step_size *step_size*
-pe_step_size *pe_step_size*
-diagonal_scale [(*True* or *False*)]
-debug_diffuse_layer [(*True* or *False*)]
-debug_inverse [(*True* or *False*)]
-debug_model [(*True* or *False*)]
-debug_prep [(*True* or *False*)]
-debug_set [(*True* or *False*)]
-logfile [(*True* or *False*)]

MIX

MIX [*number*] [*description*]

solution number, mixing fraction

PHASES

PHASES

Phase name

Dissolution reaction

log_k *log K*

delta_h *enthalpy,* [*units*]

-analytical_expression A_1, A_2, A_3, A_4, A_5

-no_check

PRINT

PRINT

-reset [(*True* or *False*)]

-eh [(*True* or *False*)]

-equilibrium_phases [(*True* or *False*)]

-exchange [(*True* or *False*)]

-gas_phase [(*True* or *False*)]

-headings [(*True* or *False*)]

-inverse_modeling [(*True* or *False*)]

-kinetics [(*True* or *False*)]

-other [(*True* or *False*)]

-saturation_indices [(*True* or *False*)]

-solid_solutions [(*True* or *False*)]

-species [(*True* or *False*)]

-surface [(*True* or *False*)]

-totals [(*True* or *False*)]

-user_print [(*True* or *False*)]

-selected_output [(*True* or *False*)]

-status [(*True* or *False*)]

RATES

RATES

name of rate expression

-start

numbered Basic statements

-end

REACTION

Explicit definition of steps

REACTION [*number*] [*description*]
(*phase name* or *formula*), [*relative stoichiometry*]
list of reaction amounts, [*units*]

Equal increment definition of steps

reaction amount [*units*] [**in** *steps*]

REACTION_TEMPERATURE

Explicit definition of steps

REACTION_TEMPERATURE [*number*] [*description*]
list of temperatures

Equal increment definition of steps

$temp_1$, $temp_2$, **in** *steps*

SAVE

SAVE *keyword, number*

SELECTED_OUTPUT

SELECTED_OUTPUT
-file *file name*
-selected_out [(*True or False*)]
-user_punch [(*True* or *False*)]
-high_precision [(*True* or *False*)]
-reset [(*True* or *False*)]
-simulation [(*True* or *False*)]
-state [(*True* or *False*)]
-solution [(*True* or *False*)]
-distance [(*True* or *False*)]
-time [(*True* or *False*)]
-step [(*True* or *False*)]
-pH [(*True* or *False*)]
-pe [(*True* or *False*)]
-reaction [(*True* or *False*)]
-temperature [(*True* or *False*)]
-alkalinity [(*True* or *False*)]

-ionic_strength [(*True* or *False*)]
-water [(*True* or *False*)]
-charge_balance [(*True* or *False*)]
-percent_error [(*True* or *False*)]
-totals *element list*
-molalities *species list*
-activities *species list*
-equilibrium_phases *phase list*
-saturation_indices *phase list*
-gases *gas-component list*
-kinetic_reactants *reactant list*
-solid_solutions *component list*
-inverse_modeling [(*True* or *False*)]

SOLID_SOLUTIONS

SOLID_SOLUTIONS [*number*] [*description*]
solid-solution name
-comp *phase name, moles*
-comp1 *phase name, moles*
-comp2 *phase name, moles*
-temp *temperature in Celsius*
-tempk *temperature in Kelvin*
-Gugg_nondim *a0, a1*
-Gugg_kJ *a0, a1*
-activity_coefficients a_{comp_1}, a_{comp_2}, x_1, x_2
-distribution_coefficients k_1, k_2, x_1, x_2
-miscibility_gap x_1, x_2
-spinodal_gap x_1, x_2
-critical_point x_{cp}, t_{cp}
-alyotropic_point x_{aly}, $\log_{10}(\Sigma\Pi)$
-Thompson wg_2, wg_1
-Margules $alpha_2$, $alpha_3$

SOLUTION

SOLUTION [*number*] [*description*]
temp *temperature*
pH *pH* [(**charge** or *phase name* [*saturation index*])]
pe *pe* [(**charge** or *phase name* [*saturation index*])]

redox *redox couple*

units *concentration units*

density *density*

element list, concentration, [*units*], ([**as** *formula*] or [**gfw** *gfw*]), [*redox couple*], [(**charge** or *phase name* [*saturation index*])]

-isotope *name, value,* [*uncertainty limit*]

-water *mass*

SOLUTION_MASTER_SPECIES

SOLUTION_MASTER_SPECIES

element name, master species, alkalinity, (*gram formula weight* or *formula*), *gram formula weight of element*

SOLUTION_SPECIES

SOLUTION_SPECIES

Association reaction

log_k *log K*

delta_h *enthalpy,* [*units*]

-analytical_expression A_1, A_2, A_3, A_4, A_5

-gamma *Debye-Hückel a, Debye-Hückel b*

-no_check

-mole_balance *formula*

SOLUTION_SPREAD

SOLUTION_SPREAD

-temp *temperature*

-pH *pH*

-pe *pe*

-redox *redox couple*

-units *concentration units*

-density *density*

-water *mass*

-isotope *name, value,* [*uncertainty_limit*]

-isotope_uncertainty *name, uncertainty_limit*

column headings

[*subheadings*]

chemical data

SURFACE

Implicit definition of surface composition

SURFACE [*number*] [*description*]
-equilibrate *number*
surface binding-site name, sites, specific_area_per_gram, mass
surface binding-site formula, name, [(**equilibrium_phase** or **kinetic_reactant**)], *sites_per_mole, specific_area_per_mole*
-no_edl
-diffuse_layer [*thickness*]
-only_counter_ions

Explicit definition of surface composition

SURFACE [*number*] [*description*]
surface binding-site formula, sites, specific_area_per_gram, mass
surface binding-site formula, name, [(**equilibrium_phase** or **kinetic_reactant**)], *sites_per_mole, specific_area_per_mole*

SURFACE_MASTER_SPECIES

SURFACE_MASTER_SPECIES
surface binding-site name, surface master species

SURFACE_SPECIES

SURFACE_SPECIES
Association reaction
log_k *log K*
delta_h *enthalpy,* [*units*]
-analytical_expression A_1, A_2, A_3, A_4, A_5
-no_check
-mole_balance *formula*

TITLE

TITLE *comment*
comment

TRANSPORT

TRANSPORT
-cells *cells*
-shifts *shifts*

-time_step *time step*

-flow_direction (**forward**, **back**, or **diffusion_only**)

-boundary_conditions *first, last*

-lengths *list of lengths*

-dispersivities *list of dispersivities*

-correct_disp [(*True* or *False*)]

-diffusion_coefficient *diffusion coefficient*

-stagnant *stagnant_cells* [*exchange_factor* θ_m θ_{im}]

-thermal_diffusion *temperature retardation factor, thermal diffusion coefficient*

-initial_time *initial_time*

-print_cells *list of cell numbers*

-print_frequency *print_modulus*

-punch_cells *list of cell numbers*

-punch_frequency *punch_modulus*

-dump *dump file*

-dump_frequency *dump_modulus*

-dump_restart *shift number*

-warnings [(*True* or *False*)]

USE

USE *keyword*, (*number* or **none**)

USER_PRINT

USER_PRINT

-start

numbered Basic statements

-end

USER_PUNCH

USER_PUNCH

-headings *list of column headings*

-start

numbered Basic statements

-end

EXAMPLES

In this section of the report example calculations using PHREEQC are presented that demonstrate most of the capabilities of the program. Several of the examples are derived from examples in the PHREEQE manual (Parkhurst and others, 1980). The input files for all examples are included in tables, which should serve as templates for modeling other geochemical processes. Only selected output from each of the example runs is presented.

Example 1.--Speciation Calculation

This example calculates the distribution of aqueous species in seawater and the saturation state of seawater relative to a set of minerals. To demonstrate how to expand the model to new elements, uranium is added to the aqueous model defined by *phreeqc.dat*. [One of the database files included with the program distribution, *wateq4f.dat*, is derived from WATEQ4F (Ball and Nordstrom, 1991) and includes uranium.]

Table 10.--Seawater composition

[Concentration is in parts per million (ppm) unless specified otherwise]

Analysis	PHREEQC notation	Concentration
Calcium	Ca	412.3
Magnesium	Mg	1291.8
Sodium	Na	10768.0
Potassium	K	399.1
Iron	Fe	.002
Manganese	Mn	.0002
Silica, as SiO_2	Si	4.28
Chloride	Cl	19353.0
Alkalinity, as HCO_3^-	Alkalinity	141.682
Sulfate, as SO_4^{2-}	S(6)	2712.0
Nitrate. as NO_3^-	N(5)	.29
Ammonium, as NH_4^+	N(-3)	.03
Uranium	U	.0033
pH, standard units	pH	8.22
pe, unitless	pe	8.451
Temperature, °C	temperature	25.0
Density, kilograms per liter	density	1.023

The essential data needed for a speciation calculation are the temperature, pH, and concentrations of elements and (or) element valence states. These data for seawater are given in table 10. The input data set for this example calculation is shown in table 11. A comment about the calculations performed in this simulation is included with the **TITLE** keyword. The **SOLUTION** data block defines the composition of seawater. Note that

valence states are identified by the chemical symbol for the element followed by the valence in parentheses [S(6), N(5), N(-3), and O(0)].

The pe to be used for distributing redox elements and for calculating saturation indices is specified by the **redox** identifier. In this example, a pe is to be calculated from the O(-2)/O(0) redox couple, which corresponds to the dissolved oxygen/water couple, and this calculated pe will be used for all calculations that require a pe. If **redox** were not specified, the default would be the input pe. The default redox identifier can be overridden for any redox element, as demonstrated by the manganese input, where the input pe will be used to speciate manganese among its valence states, and the uranium input, where the nitrate/ammonium couple will be used to calculate a pe with which to speciate uranium among its valence states.

The default units are specified to be ppm in this data set (**units** identifier). This default can be overridden for any concentration, as demonstrated by the uranium concentration, which is specified to be ppb instead of ppm. Because ppm is a mass unit, not a mole unit, the program must use a gram formula weight to convert each concentration into molal units. The default gram formula weights for each master species are specified in the **SOLUTION_MASTER_SPECIES** input (the values for the default database *phreeqc.dat* are listed in table 4 and in Attachment B). If the data are reported relative to a gram formula weight different from the default, it is necessary to specify the appropriate gram formula weight in the input data set. This can be done with the **gfw** identifier, where the actual gram formula weight is input--the gram-formula weight by which to convert nitrate is specified to be 62.0 g/mol, or more simply with the **as** identifier, where the chemical formula for the reported units is input, as shown in the input for alkalinity and ammonium in this example. Note finally that the concentration of O(0), dissolved oxygen, is given an initial estimate of 1 ppm, but that its concentration will be adjusted until a log partial pressure of oxygen gas of -0.7 is achieved. [O2(g) is defined under **PHASES** input of the default database file (Attachment B).] When using phase equilibria to specify initial concentrations [like O(0) in this example], only one concentration is adjusted. For example, if gypsum were used to adjust the calcium concentration, the concentration of calcium would vary, but the concentration of sulfate would remain fixed.

Table 11.--Input data set for example 1

```
TITLE Example 1.--Add uranium and speciate seawater.
SOLUTION 1  SEAWATER FROM NORDSTROM ET AL. (1979)
        units   ppm
        pH      8.22
        pe      8.451
        density 1.023
        temp    25.0
        redox   O(0)/O(-2)
        Ca              412.3
        Mg              1291.8
        Na              10768.0
        K               399.1
        Fe              0.002
        Mn              0.0002  pe
        Si              4.28
        Cl              19353.0
        Alkalinity      141.682 as HCO3
        S(6)            2712.0
        N(5)            0.29    gfw   62.0
```

```
        N(-3)           0.03    as      NH4
        U               3.3     ppb     N(5)/N(-3)
        O(0)            1.0     O2(g)   -0.7
SOLUTION_MASTER_SPECIES
        U       U+4     0.0     238.0290        238.0290
        U(4)    U+4     0.0     238.0290
        U(5)    UO2+    0.0     238.0290
        U(6)    UO2+2   0.0     238.0290
SOLUTION_SPECIES
        #primary master species for U
        #is also secondary master species for U(4)
        U+4 = U+4
                log_k           0.0
        U+4 + 4 H2O = U(OH)4 + 4 H+
                log_k           -8.538
                delta_h         24.760 kcal
        U+4 + 5 H2O = U(OH)5- + 5 H+
                log_k           -13.147
                delta_h         27.580 kcal
        #secondary master species for U(5)
        U+4 + 2 H2O = UO2+ + 4 H+ + e-
                log_k           -6.432
                delta_h         31.130 kcal
        #secondary master species for U(6)
        U+4 + 2 H2O = UO2+2 + 4 H+ + 2 e-
                log_k           -9.217
                delta_h         34.430 kcal
        UO2+2 + H2O = UO2OH+ + H+
                log_k           -5.782
                delta_h         11.015 kcal
        2UO2+2 + 2H2O = (UO2)2(OH)2+2 + 2H+
                log_k           -5.626
                delta_h         -36.04 kcal
        3UO2+2 + 5H2O = (UO2)3(OH)5+ + 5H+
                log_k           -15.641
                delta_h         -44.27 kcal
        UO2+2 + CO3-2 = UO2CO3
                log_k           10.064
                delta_h         0.84 kcal
        UO2+2 + 2CO3-2 = UO2(CO3)2-2
                log_k           16.977
                delta_h         3.48 kcal
        UO2+2 + 3CO3-2 = UO2(CO3)3-4
                log_k           21.397
                delta_h         -8.78 kcal
PHASES
        Uraninite
        UO2 + 4 H+ = U+4 + 2 H2O
        log_k           -3.490
        delta_h         -18.630 kcal
END
```

Uranium is not included in *phreeqc.dat*, one of the database files that is distributed with the program. Thus, data to describe the thermodynamics and composition of aqueous uranium species must be included in the input data when using this database file. Two keyword data blocks are needed to define the uranium species, **SOLUTION_MASTER_SPECIES** and **SOLUTION_SPECIES**. By adding these two data blocks to the input data file, aqueous uranium species will be defined for the duration of the run. To add uranium permanently to the list of elements, these data blocks should be added to the database file. The data for uranium shown here are intended to be illustrative and are not a complete description of uranium speciation.

It is necessary to define a primary master species for uranium with **SOLUTION_MASTER_SPECIES** input. Because uranium is a redox-active element, it is also necessary to define a secondary master species for each valence state of uranium. The data block **SOLUTION_MASTER_SPECIES** (table 11) defines U^{+4} as the primary master species for uranium and also as the secondary master species for the +4 valence state. UO_2^{+} is the secondary master species for the +5 valence state, and UO_2^{+2} is the secondary master species for the +6 valence state. Equations defining these aqueous species plus any other complexes of uranium must be defined through **SOLUTION_SPECIES** input.

In the data block **SOLUTION_SPECIES** (table 11), the primary and secondary master species are noted with comments. A primary master species is always defined in the form of an identity reaction (U+4 = U+4). Secondary master species are the only aqueous species that contain electrons in their chemical reaction. Additional hydroxide and carbonate complexes are defined for the +4 and +6 valence states, but none for the +5 state.

Finally, a new phase, uraninite, is defined with **PHASES** input. This phase will be used in calculating saturation indices in speciation modeling, but could also be used, without redefinition, for batch-reaction, transport, or inverse calculations within the computer run.

Table 12.--Output for example 1

```
   Input file: ex1
  Output file: ex1.out
Database file: ../phreeqc.dat

------------------
Reading data base.
------------------

        SOLUTION_MASTER_SPECIES
        SOLUTION_SPECIES
        PHASES
        EXCHANGE_MASTER_SPECIES
        EXCHANGE_SPECIES
        SURFACE_MASTER_SPECIES
        SURFACE_SPECIES
        RATES
        END
-----------------------------------
Reading input data for simulation 1.
-----------------------------------

        TITLE Example 1.--Add uranium and speciate seawater.
        SOLUTION 1  SEAWATER FROM NORDSTROM ET AL. (1979)
                units   ppm
                pH      8.22
                pe      8.451
                density 1.023
                temp    25.0
                redox   O(0)/O(-2)
                Ca              412.3
                Mg              1291.8
                Na              10768.0
                K               399.1
                Fe              0.002
                Mn              0.0002  pe
                Si              4.28
                Cl              19353.0
                Alkalinity      141.682 as HCO3
                S(6)            2712.0
                N(5)            0.29    gfw   62.0
                N(-3)           0.03    as    NH4
                U               3.3     ppb   N(5)/N(-3)
                O(0)            1.0     O2(g) -0.7
```

```
        SOLUTION_MASTER_SPECIES
                U       U+4     0.0     238.0290     238.0290
                U(4)    U+4     0.0     238.0290
                U(5)    UO2+    0.0     238.0290
                U(6)    UO2+2   0.0     238.0290
        SOLUTION_SPECIES
                U+4 = U+4
                        log_k           0.0
                U+4 + 4 H2O = U(OH)4 + 4 H+
                        log_k           -8.538
                        delta_h         24.760 kcal
                U+4 + 5 H2O = U(OH)5- + 5 H+
                        log_k           -13.147
                        delta_h         27.580 kcal
                U+4 + 2 H2O = UO2+ + 4 H+ + e-
                        log_k           -6.432
                        delta_h         31.130 kcal
                U+4 + 2 H2O = UO2+2 + 4 H+ + 2 e-
                        log_k           -9.217
                        delta_h         34.430 kcal
                UO2+2 + H2O = UO2OH+ + H+
                        log_k           -5.782
                        delta_h         11.015 kcal
                2UO2+2 + 2H2O = (UO2)2(OH)2+2 + 2H+
                        log_k           -5.626
                        delta_h         -36.04 kcal
                3UO2+2 + 5H2O = (UO2)3(OH)5+ + 5H+
                        log_k           -15.641
                        delta_h         -44.27 kcal
                UO2+2 + CO3-2 = UO2CO3
                        log_k           10.064
                        delta_h         0.84 kcal
                UO2+2 + 2CO3-2 = UO2(CO3)2-2
                        log_k           16.977
                        delta_h         3.48 kcal
                UO2+2 + 3CO3-2 = UO2(CO3)3-4
                        log_k           21.397
                        delta_h         -8.78 kcal
        PHASES
                Uraninite
                UO2 + 4 H+ = U+4 + 2 H2O
                log_k           -3.490
                delta_h         -18.630 kcal
        END
-----
TITLE
-----

 Example 1.--Add uranium and speciate seawater.

-------------------------------------------
Beginning of initial solution calculations.
-------------------------------------------

Initial solution 1.     SEAWATER FROM NORDSTROM ET AL. (1979)

-----------------------------Solution composition------------------------------

        Elements            Molality       Moles

        Alkalinity        2.406e-03   2.406e-03
        Ca                1.066e-02   1.066e-02
        Cl                5.657e-01   5.657e-01
        Fe                3.711e-08   3.711e-08
        K                 1.058e-02   1.058e-02
        Mg                5.507e-02   5.507e-02
        Mn                3.773e-09   3.773e-09
        N(-3)             1.724e-06   1.724e-06
        N(5)              4.847e-06   4.847e-06
        Na                4.854e-01   4.854e-01
        O(0)              3.746e-04   3.746e-04  Equilibrium with O2(g)
        S(6)              2.926e-02   2.926e-02
        Si                7.382e-05   7.382e-05
        U                 1.437e-08   1.437e-08

----------------------------Description of solution----------------------------

                                       pH  =   8.220
                                       pe  =   8.451
                          Activity of water  =   0.981
                            Ionic strength  =   6.748e-01
                       Mass of water (kg)  =   1.000e+00
                  Total carbon (mol/kg)  =   2.180e-03
                     Total CO2 (mol/kg)  =   2.180e-03
                    Temperature (deg C)  =  25.000
                 Electrical balance (eq)  =   7.936e-04
 Percent error, 100*(Cat-|An|)/(Cat+|An|)  =   0.07
                                Iterations  =   7
                                   Total H  = 1.110147e+02
                                   Total O  = 5.563047e+01

---------------------------------Redox couples---------------------------------

        Redox couple             pe  Eh (volts)

        N(-3)/N(5)           4.6750       0.2766
        O(-2)/O(0)          12.3893       0.7329

--------------------------Distribution of species--------------------------
```

	Species		Molality	Activity	Log Molality	Log Activity	Log Gamma
	OH-		2.674e-06	1.629e-06	-5.573	-5.788	-0.215
	H+		7.981e-09	6.026e-09	-8.098	-8.220	-0.122
	H2O		5.551e+01	9.806e-01	-0.009	-0.009	0.000
C(4)		2.180e-03					
	HCO3-		1.514e-03	1.023e-03	-2.820	-2.990	-0.170
	MgHCO3+		2.195e-04	1.640e-04	-3.658	-3.785	-0.127
	NaHCO3		1.667e-04	1.948e-04	-3.778	-3.710	0.067
	MgCO3		8.913e-05	1.041e-04	-4.050	-3.982	0.067
	NaCO3-		6.718e-05	5.020e-05	-4.173	-4.299	-0.127
	CaHCO3+		4.597e-05	3.106e-05	-4.337	-4.508	-0.170
	CO3-2		3.821e-05	7.959e-06	-4.418	-5.099	-0.681
	CaCO3		2.725e-05	3.183e-05	-4.565	-4.497	0.067
	CO2		1.210e-05	1.413e-05	-4.917	-4.850	0.067
	UO2(CO3)3-4		1.255e-08	1.184e-10	-7.901	-9.927	-2.025
	UO2(CO3)2-2		1.814e-09	5.653e-10	-8.741	-9.248	-0.506
	MnCO3		2.696e-10	3.150e-10	-9.569	-9.502	0.067
	MnHCO3+		6.077e-11	4.541e-11	-10.216	-10.343	-0.127
	UO2CO3		7.429e-12	8.678e-12	-11.129	-11.062	0.067
	FeCO3		1.952e-20	2.281e-20	-19.709	-19.642	0.067
	FeHCO3+		1.635e-20	1.222e-20	-19.786	-19.913	-0.127
Ca		1.066e-02					
	Ca+2		9.504e-03	2.380e-03	-2.022	-2.623	-0.601
	CaSO4		1.083e-03	1.266e-03	-2.965	-2.898	0.067
	CaHCO3+		4.597e-05	3.106e-05	-4.337	-4.508	-0.170
	CaCO3		2.725e-05	3.183e-05	-4.565	-4.497	0.067
	CaOH+		8.604e-08	6.429e-08	-7.065	-7.192	-0.127
	CaHSO4+		5.979e-11	4.467e-11	-10.223	-10.350	-0.127
Cl		5.657e-01					
	Cl-		5.657e-01	3.528e-01	-0.247	-0.452	-0.205
	MnCl+		9.582e-10	7.160e-10	-9.019	-9.145	-0.127
	MnCl2		9.439e-11	1.103e-10	-10.025	-9.958	0.067
	MnCl3-		1.434e-11	1.071e-11	-10.844	-10.970	-0.127
	FeCl+2		9.557e-19	2.978e-19	-18.020	-18.526	-0.506
	FeCl2+		6.281e-19	4.693e-19	-18.202	-18.329	-0.127
	FeCl+		7.786e-20	5.817e-20	-19.109	-19.235	-0.127
	FeCl3		1.417e-20	1.656e-20	-19.849	-19.781	0.067
Fe(2)		6.909e-19					
	Fe+2		5.205e-19	1.195e-19	-18.284	-18.923	-0.639
	FeCl+		7.786e-20	5.817e-20	-19.109	-19.235	-0.127
	FeSO4		4.845e-20	5.660e-20	-19.315	-19.247	0.067
	FeCO3		1.952e-20	2.281e-20	-19.709	-19.642	0.067
	FeHCO3+		1.635e-20	1.222e-20	-19.786	-19.913	-0.127
	FeOH+		8.227e-21	6.147e-21	-20.085	-20.211	-0.127
	FeHSO4+		3.000e-27	2.242e-27	-26.523	-26.649	-0.127
Fe(3)		3.711e-08					
	Fe(OH)3		2.841e-08	3.318e-08	-7.547	-7.479	0.067
	Fe(OH)4-		6.591e-09	4.924e-09	-8.181	-8.308	-0.127
	Fe(OH)2+		2.118e-09	1.583e-09	-8.674	-8.801	-0.127
	FeOH+2		9.425e-14	2.937e-14	-13.026	-13.532	-0.506
	FeSO4+		1.093e-18	8.167e-19	-17.961	-18.088	-0.127
	FeCl+2		9.557e-19	2.978e-19	-18.020	-18.526	-0.506
	FeCl2+		6.281e-19	4.693e-19	-18.202	-18.329	-0.127
	Fe+3		3.509e-19	2.796e-20	-18.455	-19.554	-1.099
	Fe(SO4)2-		6.372e-20	4.761e-20	-19.196	-19.322	-0.127
	FeCl3		1.417e-20	1.656e-20	-19.849	-19.781	0.067
	Fe2(OH)2+4		2.462e-24	2.322e-26	-23.609	-25.634	-2.025
	FeHSO4+2		4.228e-26	1.318e-26	-25.374	-25.880	-0.506
	Fe3(OH)4+5		1.122e-29	7.679e-33	-28.950	-32.115	-3.165
H(0)		0.000e+00					
	H2		0.000e+00	0.000e+00	-44.436	-44.369	0.067
K		1.058e-02					
	K+		1.042e-02	6.495e-03	-1.982	-2.187	-0.205
	KSO4-		1.627e-04	1.216e-04	-3.789	-3.915	-0.127
	KOH		3.137e-09	3.665e-09	-8.503	-8.436	0.067
Mg		5.507e-02					
	Mg+2		4.742e-02	1.371e-02	-1.324	-1.863	-0.539
	MgSO4		7.330e-03	8.562e-03	-2.135	-2.067	0.067
	MgHCO3+		2.195e-04	1.640e-04	-3.658	-3.785	-0.127
	MgCO3		8.913e-05	1.041e-04	-4.050	-3.982	0.067
	MgOH+		1.084e-05	8.100e-06	-4.965	-5.092	-0.127
Mn(2)		3.773e-09					
	Mn+2		2.171e-09	4.982e-10	-8.663	-9.303	-0.639
	MnCl+		9.582e-10	7.160e-10	-9.019	-9.145	-0.127
	MnCO3		2.696e-10	3.150e-10	-9.569	-9.502	0.067
	MnSO4		2.021e-10	2.360e-10	-9.695	-9.627	0.067
	MnCl2		9.439e-11	1.103e-10	-10.025	-9.958	0.067
	MnHCO3+		6.077e-11	4.541e-11	-10.216	-10.343	-0.127
	MnCl3-		1.434e-11	1.071e-11	-10.844	-10.970	-0.127
	MnOH+		2.789e-12	2.084e-12	-11.555	-11.681	-0.127
	Mn(NO3)2		1.375e-20	1.606e-20	-19.862	-19.794	0.067
Mn(3)		5.993e-26					
	Mn+3		5.993e-26	4.349e-27	-25.222	-26.362	-1.139
N(-3)		1.724e-06					
	NH4+		1.609e-06	9.049e-07	-5.794	-6.043	-0.250
	NH3		7.326e-08	8.558e-08	-7.135	-7.068	0.067
	NH4SO4-		4.157e-08	3.106e-08	-7.381	-7.508	-0.127
N(5)		4.847e-06					
	NO3-		4.847e-06	2.846e-06	-5.315	-5.546	-0.231
	Mn(NO3)2		1.375e-20	1.606e-20	-19.862	-19.794	0.067
Na		4.854e-01					
	Na+		4.791e-01	3.387e-01	-0.320	-0.470	-0.151
	NaSO4-		6.053e-03	4.523e-03	-2.218	-2.345	-0.127
	NaHCO3		1.667e-04	1.948e-04	-3.778	-3.710	0.067
	NaCO3-		6.718e-05	5.020e-05	-4.173	-4.299	-0.127
	NaOH		3.117e-07	3.641e-07	-6.506	-6.439	0.067

```
O(0)              3.746e-04
        O2                1.873e-04   2.188e-04    -3.727    -3.660     0.067
S(6)              2.926e-02
        SO4-2             1.463e-02   2.664e-03    -1.835    -2.574    -0.740
        MgSO4             7.330e-03   8.562e-03    -2.135    -2.067     0.067
        NaSO4-            6.053e-03   4.523e-03    -2.218    -2.345    -0.127
        CaSO4             1.083e-03   1.266e-03    -2.965    -2.898     0.067
        KSO4-             1.627e-04   1.216e-04    -3.789    -3.915    -0.127
        NH4SO4-           4.157e-08   3.106e-08    -7.381    -7.508    -0.127
        HSO4-             2.089e-09   1.561e-09    -8.680    -8.807    -0.127
        MnSO4             2.021e-10   2.360e-10    -9.695    -9.627     0.067
        CaHSO4+           5.979e-11   4.467e-11   -10.223   -10.350    -0.127
        FeSO4+            1.093e-18   8.167e-19   -17.961   -18.088    -0.127
        Fe(SO4)2-         6.372e-20   4.761e-20   -19.196   -19.322    -0.127
        FeSO4             4.845e-20   5.660e-20   -19.315   -19.247     0.067
        FeHSO4+2          4.228e-26   1.318e-26   -25.374   -25.880    -0.506
        FeHSO4+           3.000e-27   2.242e-27   -26.523   -26.649    -0.127
Si                7.382e-05
        H4SiO4            7.110e-05   8.306e-05    -4.148    -4.081     0.067
        H3SiO4-           2.720e-06   2.032e-06    -5.565    -5.692    -0.127
        H2SiO4-2          7.362e-11   2.294e-11   -10.133   -10.639    -0.506
U(4)              1.034e-21
        U(OH)5-           1.034e-21   7.726e-22   -20.985   -21.112    -0.127
        U(OH)4            1.652e-25   1.930e-25   -24.782   -24.715     0.067
        U+4               0.000e+00   0.000e+00   -46.997   -49.022    -2.025
U(5)              1.622e-18
        UO2+              1.622e-18   1.212e-18   -17.790   -17.916    -0.127
U(6)              1.437e-08
        UO2(CO3)3-4       1.255e-08   1.184e-10    -7.901    -9.927    -2.025
        UO2(CO3)2-2       1.814e-09   5.653e-10    -8.741    -9.248    -0.506
        UO2CO3            7.429e-12   8.678e-12   -11.129   -11.062     0.067
        UO2OH+            3.385e-14   2.530e-14   -13.470   -13.597    -0.127
        UO2+2             3.019e-16   9.409e-17   -15.520   -16.026    -0.506
        (UO2)2(OH)2+2     1.780e-21   5.547e-22   -20.750   -21.256    -0.506
        (UO2)3(OH)5+      2.908e-23   2.173e-23   -22.536   -22.663    -0.127

------------------------------Saturation indices-------------------------------

        Phase               SI log IAP   log KT

        Anhydrite        -0.84   -5.20   -4.36  CaSO4
        Aragonite         0.61   -7.72   -8.34  CaCO3
        Calcite           0.76   -7.72   -8.48  CaCO3
        Chalcedony       -0.51   -4.06   -3.55  SiO2
        Chrysotile        3.36   35.56   32.20  Mg3Si2O5(OH)4
        CO2(g)           -3.38  -21.53  -18.15  CO2
        Dolomite          2.41  -14.68  -17.09  CaMg(CO3)2
        Fe(OH)3(a)        0.19   -3.42   -3.61  Fe(OH)3
        Goethite          6.09   -3.41   -9.50  FeOOH
        Gypsum           -0.63   -5.21   -4.58  CaSO4:2H2O
        H2(g)           -41.22    1.82   43.04  H2
        H2O(g)           -1.52   -0.01    1.51  H2O
        Halite           -2.50   -0.92    1.58  NaCl
        Hausmannite       1.57   19.56   17.99  Mn3O4
        Hematite         14.20   -6.81  -21.01  Fe2O3
        Jarosite-K       -7.52  -42.23  -34.71  KFe3(SO4)2(OH)6
        Manganite         2.39    6.21    3.82  MnOOH
        Melanterite     -19.35  -21.56   -2.21  FeSO4:7H2O
        NH3(g)           -8.84    2.18   11.01  NH3
        O2(g)            -0.70   -3.66   -2.96  O2
        Pyrochroite      -8.08    7.12   15.20  Mn(OH)2
        Pyrolusite        6.96    5.30   -1.66  MnO2
        Quartz           -0.08   -4.06   -3.98  SiO2
        Rhodochrosite    -3.27  -14.40  -11.13  MnCO3
        Sepiolite         1.16   16.92   15.76  Mg2Si3O7.5OH:3H2O
        Sepiolite(d)     -1.74   16.92   18.66  Mg2Si3O7.5OH:3H2O
        Siderite        -13.13  -24.02  -10.89  FeCO3
        SiO2(a)          -1.35   -4.06   -2.71  SiO2
        Talc              6.04   27.44   21.40  Mg3Si4O10(OH)2
        Uraninite       -12.67    4.39   17.06  UO2

------------------
End of simulation.
------------------
```

The output from the model (table 12) contains several blocks of information delineated by headings. First, the names of the input, output, and database files for the run are listed. Next, all keywords encountered in reading the database file are listed under the heading "Reading data base". Next, the input data, excluding comments and empty lines, is echoed under the heading "Reading input data for simulation 1". The simulation is defined by all input data up to and including the **END** keyword.

Any comment entered within the simulation with the **TITLE** keyword is printed next. The title is followed by the heading, "Beginning of initial solution calculations", below which are the results of the speciation calculation for seawater. The concentration data, converted to molality are given under the subheading "Solution composition". For initial solution calculations, the number of moles in solution is numerically equal to molality, because 1 kg of water is assumed. The **-water** identifier can be used to define a different mass of water for a

solution. During batch-reaction calculations, the mass of water may change and the moles in the aqueous phase will not exactly equal the molality of a constituent. Note that the molality of dissolved oxygen that produces a log partial pressure of -0.7 has been calculated and is annotated in the output.

After the subheading "Description of solution", some of the properties listed in the first block of output are equal to their input values and some are calculated. In this example, pH, pe, and temperature are equal to the input values. The ionic strength, total carbon (alkalinity was the input datum), total inorganic carbon ("Total CO2"), electrical balance, and percent error have been calculated by the model.

Under the subheading "Redox couples" the pe and Eh are printed for each redox couple for which data were available, in this case, ammonium/nitrate and water/dissolved oxygen.

Under the subheading "Distribution of species", the molalities, activities, and activity coefficients of all species of each element and element valence state are listed. The lists are alphabetical by element name and descending in terms of molality within each element or element valence state. Beside the name of each element or element valence state, the total molality is given.

Finally, under the subheading "Saturation indices", saturation indices for all minerals that are appropriate for the given analytical data are listed alphabetically by phase name near the end of the output. The saturation index is given in the column headed "SI", followed by the columns for the log of the ion activity product ("log IAP") and the log of the solubility constant ("log KT"). The chemical formulas for each of the phases is printed in the right-hand column. Note for example, that no aluminum bearing minerals are included because aluminum was not included in the analytical data. Also note that mackinawite (FeS) and other sulfide minerals are not included in the output because no analytical data were specified for S(-2). If a concentration for S [instead of S(6)] or S(-2) had been entered, then a concentration of S(-2) would have been calculated and a saturation index for mackinawite and other sulfide minerals would have been calculated.

Example 2.--Equilibration with Pure Phases

This example determines the solubility of the most stable phase, gypsum or anhydrite, over a range of temperatures. The input data set is given in table 13. Only the pH and temperature are used to define the pure water solution. Default units are millimolal, but no concentrations are specified. By default, pe is 4.0, the default redox calculation uses pe, and the density is 1.0 (not needed because no concentrations are "per liter"). All phases that are allowed to react to a specified saturation index during the batch-reaction calculation are listed in **EQUILIBRIUM_PHASES**, whether they are initially present or not. The input data include the name of the phase (previously defined through **PHASES** input in the database or input file), the specified saturation index, and the amount of the phase present, in moles. If a phase is not present initially, it is given 0.0 mol in the pure-phase assemblage. In this example, gypsum and anhydrite are allowed to react to equilibrium (saturation index equal to 0.0), and the initial phase assemblage has 1 mol of each mineral. Each mineral will react either to equilibrium or until it is exhausted in the assemblage. In most cases, 1 mol of a single phase is sufficient to reach equilibrium.

Table 13.--Input data set for example 2

```
TITLE Example 2.--Temperature dependence of solubility
                  of gypsum and anhydrite
SOLUTION 1 Pure water
        pH      7.0
        temp    25.0
```

```
EQUILIBRIUM_PHASES 1
        Gypsum              0.0      1.0
        Anhydrite           0.0      1.0
REACTION_TEMPERATURE 1
        25.0 75.0 in 51 steps
SELECTED_OUTPUT
        -file    ex2.sel
        -si      anhydrite  gypsum
END
```

A set of 51 temperatures is specified in the **REACTION_TEMPERATURE** data block. The input data specify that for every degree of temperature, beginning at 25°C and ending at 75°C, the phases defined by **EQUILIBRIUM_PHASES** (gypsum and anhydrite) will react to equilibrium, if possible, or until both phases are completely dissolved. Finally, **SELECTED_OUTPUT** is used to write the saturation indices for gypsum and anhydrite to the file *ex2.sel* after each calculation. This file was then used to generate figure 5.

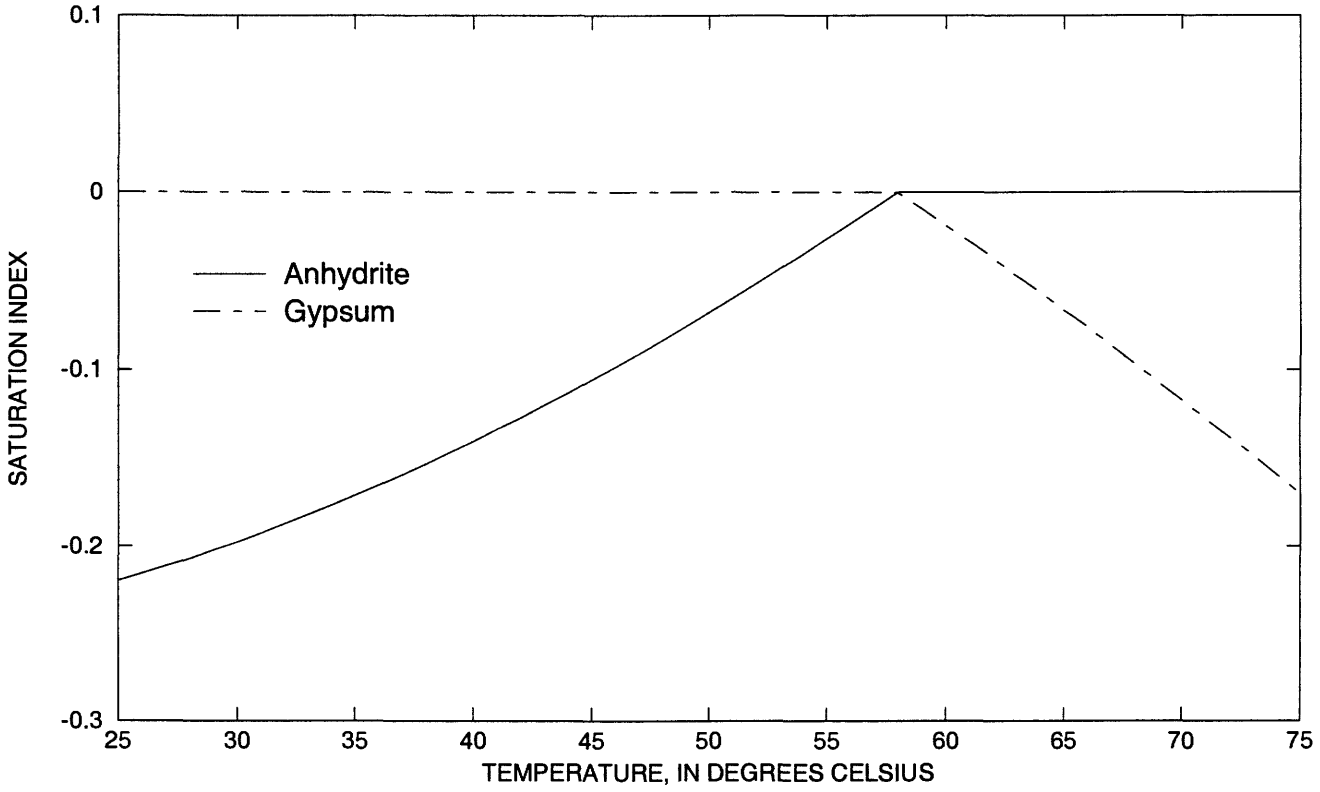

Figure 5.--Saturation indices of gypsum and anhydrite in solutions that have equilibrated with the more stable of the two phases over the temperature range 25 to 75° Celsius.

The results of the initial solution calculation and the first batch-reaction step are shown in table 14. The distribution of species for pure water is shown under the heading "Beginning of initial solution calculations". The equilibration of the system with the given amounts of gypsum and anhydrite at 25°C is the first batch-reaction step, which is displayed after the heading "Beginning of batch-reaction calculations". Immediately following this heading, the batch-reaction step is identified, followed by a list of the identity of the keyword data used in the calculation. In this example, the solution composition stored as number 1, the pure-phase assemblage stored as

number 1, and the reaction temperatures stored as number 1 are used in the calculation. Conceptually, the solution and the pure phases are put together in a beaker, which is regulated to 25°C, and allowed to react to system equilibrium.

Under the subheading "Phase assemblage", the saturation indices and amounts of each of the phases defined by **EQUILIBRIUM_PHASES** are listed. In the first batch-reaction step, the final phase assemblage contains no anhydrite, which is undersaturated with respect to the solution (saturation index equals -0.22), and 1.985 mol of gypsum, which is in equilibrium with the solution (saturation index equals 0.0). All of the anhydrite has dissolved and most of the calcium and sulfate have reprecipitated as gypsum. The "Solution composition" indicates that 15.64 mmol/kgw of calcium and sulfate remain in solution, which defines the solubility of gypsum in pure water. However, the total number of moles of each constituent in the aqueous phase is only 15.08 because the mass of water is only 0.9645 kg ("Description of solution"). In precipitating gypsum ($CaSO_4 \cdot 2H_2O$), water has been removed from solution. Thus, the mass of solvent water is not constant in batch-reaction calculations; reactions and waters of hydration in dissolving and precipitating phases may increase or decrease the mass of solvent water.

The saturation indices for all of the batch-reaction steps are plotted in figure 5. In each step, pure water was reacted with the phases at a different temperature (the reactions are not cumulative). The default database for PHREEQC indicates that gypsum is the stable phase (saturation index equals 0.0) at temperatures below about 57°C; above this temperature, anhydrite is calculated to be the stable phase.

Table 14.--Selected output for example 2

```
-------------------------------------------
Beginning of initial solution calculations.
-------------------------------------------

Initial solution 1.     Pure water

-----------------------------Solution composition------------------------------

        Elements           Molality       Moles

        Pure water

----------------------------Description of solution----------------------------

                                     pH  =   7.000
                                     pe  =   4.000
                        Activity of water  =   1.000
                           Ionic strength  =   1.001e-07
                       Mass of water (kg)  =   1.000e+00
                 Total alkalinity (eq/kg)  =   1.082e-10
                    Total carbon (mol/kg)  =   0.000e+00
                       Total CO2 (mol/kg)  =   0.000e+00
                      Temperature (deg C)  =  25.000
                   Electrical balance (eq)  =  -1.082e-10
 Percent error, 100*(Cat-|An|)/(Cat+|An|)  =  -0.05
                                Iterations  =   0
                                   Total H  = 1.110124e+02
                                   Total O  = 5.550622e+01

----------------------------Distribution of species----------------------------

                                                    Log         Log         Log
        Species            Molality    Activity  Molality  Activity     Gamma

        OH-               1.002e-07   1.001e-07    -6.999    -6.999    -0.000
        H+                1.001e-07   1.000e-07    -7.000    -7.000    -0.000
        H2O               5.551e+01   1.000e+00     0.000     0.000     0.000
H(0)              1.416e-25
        H2                7.079e-26   7.079e-26   -25.150   -25.150     0.000
O(0)              0.000e+00
        O2                0.000e+00   0.000e+00   -42.080   -42.080     0.000

-----------------------------Saturation indices-------------------------------

        Phase               SI log IAP  log KT

        H2(g)           -22.00  -22.00    0.00  H2
        H2O(g)           -1.51    0.00    1.51  H2O
        O2(g)           -39.12   44.00   83.12  O2

---------------------------------------
Beginning of batch-reaction calculations.
---------------------------------------
```

```
Reaction step 1.

Using solution 1.       Pure water
Using pure phase assemblage 1.
Using temperature 1.

-------------------------------Phase assemblage--------------------------------

                                                    Moles in assemblage
        Phase               SI log IAP  log KT    Initial      Final       Delta

        Anhydrite         -0.22   -4.58   -4.36  1.000e+00              -1.000e+00
        Gypsum             0.00   -4.58   -4.58  1.000e+00  1.985e+00   9.849e-01

-----------------------------Solution composition------------------------------

        Elements           Molality       Moles

        Ca                1.564e-02   1.508e-02
        S                 1.564e-02   1.508e-02

----------------------------Description of solution----------------------------

                                    pH  =   7.067      Charge balance
                                    pe  =  10.686      Adjusted to redox equilibrium
                         Activity of water  =   1.000
                            Ionic strength  =   4.178e-02
                        Mass of water (kg)  =   9.645e-01
                Total alkalinity (eq/kg)    =   1.122e-10
                   Total carbon (mol/kg)    =   0.000e+00
                      Total CO2 (mol/kg)    =   0.000e+00
                     Temperature (deg C)    =  25.000
                  Electrical balance (eq)   =  -1.082e-10
 Percent error, 100*(Cat-|An|)/(Cat+|An|)   =  -0.00
                                Iterations  =  19
                                   Total H  = 1.070728e+02
                                   Total O  = 5.359671e+01

----------------------------Distribution of species----------------------------

                                                    Log         Log         Log
        Species            Molality    Activity  Molality  Activity     Gamma

        OH-               1.417e-07   1.167e-07   -6.849    -6.933    -0.084
        H+                9.957e-08   8.575e-08   -7.002    -7.067    -0.065
        H2O               5.551e+01   9.996e-01   -0.000    -0.000     0.000
Ca              1.564e-02
        Ca+2              1.045e-02   5.176e-03   -1.981    -2.286    -0.305
        CaSO4             5.191e-03   5.242e-03   -2.285    -2.281     0.004
        CaOH+             1.204e-08   1.001e-08   -7.919    -7.999    -0.080
        CaHSO4+           3.166e-09   2.633e-09   -8.499    -8.580    -0.080
H(0)            4.383e-39
        H2                2.192e-39   2.213e-39  -38.659   -38.655     0.004
O(0)            1.685e-15
        O2                8.424e-16   8.505e-16  -15.074   -15.070     0.004
S(-2)           0.000e+00
        HS-               0.000e+00   0.000e+00 -117.646  -117.731    -0.084
        H2S               0.000e+00   0.000e+00 -117.860  -117.856     0.004
        S-2               0.000e+00   0.000e+00 -123.270  -123.582    -0.312
S(6)            1.564e-02
        SO4-2             1.045e-02   5.075e-03   -1.981    -2.295    -0.313
        CaSO4             5.191e-03   5.242e-03   -2.285    -2.281     0.004
        HSO4-             5.088e-08   4.231e-08   -7.293    -7.374    -0.080
        CaHSO4+           3.166e-09   2.633e-09   -8.499    -8.580    -0.080

-----------------------------Saturation indices-------------------------------

        Phase               SI log IAP  log KT

        Anhydrite         -0.22   -4.58   -4.36  CaSO4
        Gypsum             0.00   -4.58   -4.58  CaSO4:2H2O
        H2(g)            -35.51  -35.51    0.00  H2
        H2O(g)            -1.51   -0.00    1.51  H2O
        H2S(g)          -116.86 -158.45  -41.59  H2S
        O2(g)            -12.11   71.01   83.12  O2
        Sulfur           -87.23 -122.94  -35.71  S
```

Example 3.--Mixing

This example demonstrates the capabilities of PHREEQC to perform a series of geochemical simulations, with the final simulations relying on results from previous simulations within the same run. The example investigates diagenetic reactions that may occur in zones where seawater mixes with carbonate ground water. The example is divided into five simulations, labeled A through E in table 15. (A) Carbonate ground water is defined by equilibrating pure water with calcite at a P_{CO_2} of $10^{-2.0}$ atm. (B) Seawater is defined using the major-ion data given in table 10. (C) The two solutions are mixed together in the proportions 70 percent ground water and 30

percent seawater. (D) The mixture is equilibrated with calcite and dolomite. Finally, (E) the mixture is equilibrated with calcite only, to investigate the chemical evolution if dolomite precipitation is assumed to be negligible.

Table 15.--Input data set for example 3

```
TITLE Example 3, part A.--Calcite equilibrium at log Pco2 = -2.0 and 25C.
SOLUTION 1  Pure water
        pH      7.0
        temp    25.0
EQUILIBRIUM_PHASES
        CO2(g)          -2.0
        Calcite         0.0
SAVE solution 1
END
TITLE Example 3, part B.--Definition of seawater.
SOLUTION 2  Seawater
        units   ppm
        pH      8.22
        pe      8.451
        density 1.023
        temp    25.0
        Ca              412.3
        Mg              1291.8
        Na              10768.0
        K               399.1
        Si              4.28
        Cl              19353.0
        Alkalinity      141.682 as HCO3
        S(6)            2712.0
END
TITLE Example 3, part C.--Mix 70% ground water, 30% seawater.
MIX 1
        1       0.7
        2       0.3
SAVE solution   3
END
TITLE Example 3, part D.--Equilibrate mixture with calcite and dolomite.
EQUILIBRIUM_PHASES 1
        Calcite         0.0
        Dolomite        0.0
USE solution 3
END
TITLE Example 3, part E.--Equilibrate mixture with calcite only.
EQUILIBRIUM_PHASES 2
        Calcite         0.0
USE solution 3
END
```

The input for part A (table 15) consists of the definition of pure water with **SOLUTION** input, and the definition of a pure-phase assemblage with **EQUILIBRIUM_PHASES** input. In the definition of the phases, only a saturation index was given for each phase. Because it was not entered, the amount of each phase defaults to 10.0 mol, which is essentially an unlimited supply for most phases. The batch reaction is implicitly defined to be the

equilibration of the first solution defined in this simulation with the first pure-phase assemblage defined in the simulation. (Explicit definition of batch-reaction entities is done with the **USE** keyword.) The **SAVE** keyword instructs the program to save the batch-reaction solution composition from the final batch-reaction step as solution number 1. Thus, when the simulation begins, solution number 1 is pure water. After the batch-reaction calculations for the simulation are completed, the batch-reaction solution--water in equilibrium with calcite and CO_2 --is stored as solution 1.

Part B defines the composition of seawater, which is stored as solution number 2. Part C mixes ground water, (solution 1) with seawater (solution 2) in a closed system in which P_{CO_2} is calculated, not specified. The **MIX** keyword is used to define the mixing fractions (approximately mixing volumes) of each solution in the mixture. The **SAVE** keyword causes the mixture to be saved as solution number 3. The **MIX** keyword allows the mixing of an unlimited number of solutions in whatever fractions are specified. The fractions (volumes) need not sum to 1.0. If the fractions were 7.0 and 3.0 instead of 0.7 and 0.3, the number of moles of each element in solution 1 (including hydrogen and oxygen) would be multiplied by 7.0, the number of moles of each element in solution 2 would be multiplied by 3.0, and the resulting moles of elements would be added together. The mass of water in the mixture would be approximately 10 kg (7.0 from solution 1 and 3.0 from solution 2) instead of approximately 1 kg, if the fractions were 0.7 and 0.3. The concentrations in the mixture would be the same for either set of mixing fractions because the relative proportions of solution 1 and solution 2 are the same. However, during subsequent reactions it would take 10 times more mole transfer for mixing fractions 7.0 and 3.0 than that shown in table 16 because there would be 10 times more water in the system.

Part D equilibrates the mixture with calcite and dolomite. The **USE** keyword specifies that solution number 3, which is the mixture from part C, is to be the solution with which the phases will equilibrate. By defining the phase assemblage with "**EQUILIBRIUM_PHASES** 1", the phase assemblage replaces the previous assemblage number 1 that was defined in part A. Part E performs a similar calculation to part D, but uses phase assemblage 2, which does not contain dolomite as a reactant.

Table 16.--Selected results for example 3

[Simulation A generates carbonate ground water; B defines seawater; C performs mixing with no other mole transfer; D equilibrates the mixture with calcite and dolomite; and E equilibrates the mixture with calcite only. Mole transfer is relative to the moles in the phase assemblage; positive numbers indicate an increase in the amount of the phase present, that is, precipitation; negative numbers indicate a decrease in the amount of the phase present, or dissolution. Saturation index: "--" indicates saturation index calculation not possible because one of the constituent elements was not in solution. Mole transfer: "--" indicates no mole transfer of this mineral was allowed in the simulation]

Simulation	pH	log P_{CO_2}	Saturation index		Mole transfer, millimoles		
			Calcite	Dolomite	CO_2	Calcite	Dolomite
A	7.297	-2.00	0.00	--	-1.976	-1.646	--
B	8.220	-3.38	.76	2.41	--	--	--
C	7.351	-2.23	-.10	.52	--	--	--
D	7.056	-1.98	.00	.00	--	-15.71	7.935
E	7.442	-2.31	.00	.73	--	-.040	--

Selected results from the output for example 3 are presented in table 16. The ground water produced by part A is in equilibrium with calcite and has a log P_{CO_2} of -2.0, as specified by the input. The moles of CO_2 in the

phase assemblage decreased by about 2.0 mmol, which means that about 2.0 mmol dissolved into solution. Likewise, about 1.6 mmol of calcite dissolved. Part B defined seawater, which is calculated to have slightly greater than atmospheric carbon dioxide (-3.38 compared to about -3.5), and is supersaturated with calcite (saturation index 0.76) and dolomite (2.41). No mole transfers of minerals was allowed for part B. Part C performed the mixing and calculated the equilibrium distribution of species in the mixture, again with no mole transfers of the minerals allowed. The resulting log P_{CO_2} is -2.23, calcite is undersaturated, and dolomite is supersaturated. The saturation indices indicate that thermodynamically, dolomitization should occur, that is calcite should dissolve and dolomite should precipitate. Part D calculates the amounts of calcite and dolomite that should react. To produce equilibrium 15.71 mmol of calcite should dissolve and 7.935 mmol of dolomite should precipitate. Dolomitization is not observed to occur in present-day mixing zone environments, even though dolomite is the thermodynamically stable phase. The lack of significant dolomitization is due to the slow reaction kinetics of dolomite formation. Therefore, part E simulates what would happen if dolomite does not precipitate. If dolomite does not precipitate, only a very small amount of calcite dissolves (0.040 mmol) for this mixing ratio.

Example 4.--Evaporation and Homogeneous Redox Reactions

Evaporation is accomplished by removing water from the chemical system. Water can be removed by three methods: (1) water can be specified as an irreversible reactant with a negative reaction coefficient in the **REACTION** keyword input, (2) the solution can be mixed with pure water which is given a negative mixing fraction in **MIX**, or (3) "H2O" can be specified as the alternative reaction in **EQUILIBRIUM_PHASES** keyword input, in which case, water is removed or added to the aqueous phase to attain a specified saturation index for a pure phase. This example uses the first method; the **REACTION** data block is used to simulate concentration of rain water by approximately 20 fold by removing 95 percent of the water. The resulting solution contains only about 0.05 kg of water. In a subsequent simulation, the **MIX** keyword is used to generate a solution that has the same concentrations as the evaporated solution, but has a total of mass of water of approximately 1 kg.

The first simulation input data set (table 17) contains four keywords: (1) **TITLE** is used to specify a description of the simulation to be included in the output file, (2) **SOLUTION** is used to define the composition of rain water from central Oklahoma, (3) **REACTION** is used to specify the amount of water, in moles, to be removed from the aqueous phase, and (4) **SAVE** is used to store the result of the batch-reaction calculation as solution number 2.

Table 17.--Input data set for example 4

```
TITLE Example 4a.--Rain water evaporation
SOLUTION 1  Precipitation from Central Oklahoma
        units           mg/L
        pH              4.5   # estimated
        temp            25.0
        Ca              .384
        Mg              .043
        Na              .141
        K               .036
        Cl              .236
        C(4)            .1      CO2(g)  -3.5
        S(6)            1.3
        N(-3)           .208
```

```
        N(5)            .237
REACTION 1
        H2O     -1.0
        52.73 moles
SAVE solution 2
END
TITLE Example 4b.--Factor of 20 more solution
MIX
        2       20.
SAVE solution 3
END
```

All solutions defined by **SOLUTION** input are scaled to have exactly 1 kg (approximately 55.5 mol) of water, unless **-water** identifier is used. To concentrate the solution by 20 fold, it is necessary to remove approximately 52.8 mol of water (55.5 x 0.95).

The second simulation uses **MIX** to multiply by 20 the moles of all elements in the solution, including hydrogen and oxygen. This procedure effectively increases the total mass (or volume) of the aqueous phase, but maintains the same concentrations. For identification purposes, the solution that results from the **MIX** simulation is stored as solution 3 with the **SAVE** keyword. Solution 3 will have the same concentrations as solution 2 (from the previous simulation) but will have a mass of water of approximately 1 kg.

Selected results of the simulation are presented in table 18. The concentration factor of 20 is reasonable in terms of a water balance for the process of evapotranspiration in central Oklahoma (Parkhurst and others, 1996). The PHREEQC modeling assumes that evaporation and evapotranspiration have the same effect and that evapotranspiration has no effect on the ion ratios. These assumptions have not been verified and may not be correct. After evaporation, the simulated solution composition is still undersaturated with respect to calcite, dolomite, and gypsum. As expected, the mass of water decreases from 1 kg in rain water (solution 1) to approximately 0.05 kg in solution 2 after water was removed by the reaction. In general, the amount of water remaining after the reaction is approximate because water may be consumed or produced by homogeneous hydrolysis reactions, surface complexation reactions, and dissolution and precipitation of pure phases. The number of moles of chloride (μ mol) was unaffected by the removal of water; however, the concentration of chloride (μ mol/kgw) increased because the amount of water decreased. The second mixing simulation increased the mass of water and the moles of chloride by a factor of 20. Thus, the moles of chloride increased, but the chloride concentration is the same before (solution 2) and after (solution 3) the mixing simulation because the mass of water increased proportionately.

An important point about homogeneous redox reactions is illustrated in the results of these simulations (table 18). Batch-reaction calculations (and transport calculations) always produce aqueous equilibrium for each redox element. The rain water analysis contained data for both ammonium and nitrate, but none for dissolved nitrogen. The pe of the rain water has no effect on the distribution of species in the initial solution because the concentrations of individual redox states of all redox elements (C, N, and S) are specified. Although nitrate and ammonium should not coexist at thermodynamic equilibrium, the speciation calculation allows redox disequilibria and accepts the concentrations of the two redox states of nitrogen that are defined by the input data, regardless of thermodynamic equilibrium. During the batch-reaction (evaporation) step, redox equilibrium is attained for the aqueous phase, which causes ammonium to be oxidized and nitrate to be reduced, generating dissolved nitrogen [$N_{2(aq)}$, or N(0) in PHREEQC notation]. The first batch-reaction solution (solution 2) contains the equilibrium distribution of nitrogen, which consists of nitrate and dissolved nitrogen, but no ammonium (table 18). The

Table 18.--Selected results for example 4

[kg, kilogram. μ mol, micromole]

Constituent	Solution 1 Rain water	Solution 2 Concentrated 20 fold	Solution 3 Mixed with factor 20
Mass of water, kg	1.000	0.05002	1.000
Cl, μ mol	6.657	6.657	133.1
Cl, μ mol/kg water	6.657	133.1	133.1
Nitrate [N(5)], μ mol/kg water	16.9	160.1	160.1
Dissolved nitrogen [N(0)], μ mol/kg water	0	475.1	475.1
Ammonium [N(-3)], μ mol/kg water	14.8	0	0
Calcite saturation index	-9.21	-9.37	-9.37
Dolomite saturation index	-19.02	-19.35	-19.35
Gypsum saturation index	-5.35	-2.91	-2.91

oxidation of ammonium and reduction of nitrate occur in the batch-reaction calculation to produce redox equilibrium from the inherent redox disequilibrium in the definition of the rain water composition. Nitrogen redox reactions would have occurred in the simulation even if the **REACTION** keyword had specified that no water was to be removed. The only way to prevent complete equilibration of the nitrogen redox states would be to define the individual redox states as separate **SOLUTION_MASTER_SPECIES** and **SOLUTION_SPECIES**, for example by defining a new element in **SOLUTION_MASTER_SPECIES** called "Amm" and defining NH_3 and other N(-3) species in terms of Amm ($AmmH_3$, $AmmH_4^+$, and others). In this case, equilibrium would be attained among all species of N and all species of Amm, but no equilibria would exist between N and Amm species.

Example 5.--Irreversible Reactions

This example demonstrates the irreversible reaction capabilities of PHREEQC in modeling the oxidation of pyrite. Oxygen (O_2) and NaCl are added irreversibly to pure water in five varying amounts (0.0, 1.0, 5.0, 10.0, and 50.0 mmol); the relative proportion of O_2 to NaCl in the irreversible reaction is 1.0 to 0.5. Pyrite, calcite, and goethite are allowed to dissolve to equilibrium and carbon dioxide partial pressure is maintained at $10^{-3.5}$ (atmospheric partial pressure). In addition, gypsum is allowed to precipitate if it becomes supersaturated

Table 19.--Input data set for example 5

```
TITLE Example 5.--Add oxygen, equilibrate with pyrite, calcite, and goethite.
SOLUTION 1  PURE WATER
        pH      7.0
        temp    25.0
EQUILIBRIUM_PHASES 1
        Pyrite          0.0
        Goethite        0.0
        Calcite         0.0
        CO2(g)         -3.5
```

```
        Gypsum          0.0     0.0
REACTION 1
        O2      1.0
        NaCl    0.5
        0.0     0.001   0.005   0.01    0.05
SELECTED_OUTPUT
        -file   ex5.sel
        -total  Cl
        -si     Gypsum
        -equilibrium_phases  pyrite goethite calcite CO2(g) gypsum
END
```

Table 20.--Selected results for example 5

[Mole transfer is relative to the moles in the phase assemblage; positive numbers indicate an increase in the amount of the phase present, that is, precipitation; negative numbers indicate a decrease in the amount of the phase present, that is, dissolution]

Reactants added, millimoles		pH	pe	Mole transfer, millimoles					Saturation index of gypsum
O_2	NaCl			Pyrite	Goethite	Calcite	$CO_{2(g)}$	Gypsum	
0.0	0.0	8.28	-4.94	-0.000032	0.000011	-0.49	-0.49	0.0	-6.13
1.0	0.5	8.17	-4.29	-.27	.27	-.93	.14	.0	-2.02
5.0	2.5	7.98	-3.97	-1.33	1.33	-2.94	2.40	.0	-1.06
10.0	5.0	7.88	-3.82	-2.67	2.67	-5.56	5.11	0.0	-.65
50.0	25.0	7.72	-3.57	-13.33	13.33	-26.84	26.49	9.00	.0

Pure water is defined with **SOLUTION** input (table 19), and the pure-phase assemblage is defined with **EQUILIBRIUM_PHASES** input. By default, 10 mol of pyrite, goethite, calcite, and carbon dioxide are present in the pure-phase assemblage; gypsum is defined to have 0.0 mol in the pure-phase assemblage. Gypsum can only precipitate if it becomes supersaturated; it can not dissolve because no moles are initially present. The **REACTION** data block defines the irreversible reaction that is to be modeled. In this example, oxygen ("O2") will be added with a relative coefficient of 1.0 and NaCl will be added with a relative coefficient of 0.5. The steps of the reaction are defined to be 0.0, 0.001, 0.005, 0.01, and 0.05 mol. The reactants can be defined by a chemical formula, as in this case (O_2) or by a phase name that has been defined with **PHASES** input.Thus, the phase name "O2(g)" or "Halite" from the default database file, could have been used in place of "O2" or "NaCl" to achieve the same result. The number of moles of the element oxygen added to the aqueous phase in each reaction step is equal to the stoichiometric coefficient of oxygen in the formula "O2" (2.0) times the relative coefficient (1.0) times the moles of reaction defined by the reaction step (0.0, 0.001, 0.005, 0.01, or 0.05); the number of moles of chloride added at each step is the stoichiometric coefficient of chlorine in the formula "NaCl" (1.0) times the relative coefficient (0.5) times the moles in the reaction step. **SELECTED_OUTPUT** is used to write the total concentration of chloride, the saturation index of gypsum, and the total amounts and mole transfers of pyrite, goethite, calcite, carbon dioxide, and gypsum to the file *ex5.sel* after each equilibrium calculation.

The results for example 5 are summarized in table 20. When no oxygen and sodium chloride are added to the system, a small amount of calcite and carbon dioxide dissolves, and trace amounts of pyrite and goethite react;

the pH is 8.28, the pe is low (-4.94) because of equilibrium with pyrite, and gypsum is six orders of magnitude undersaturated (saturation index -6.13). As oxygen and sodium chloride are added, pyrite oxidizes and goethite, being relatively insoluble, precipitates. This reaction generates sulfuric acid, decreases the pH, slightly increases the pe, and causes calcite to dissolve and carbon dioxide to be released. At some point between 10 and 50 mmol of oxygen added, gypsum reaches saturation and begins to precipitate. When 50 mmol of oxygen and 25 mmol of sodium chloride have been added, a total of 9.00 mmol of gypsum has precipitated.

Example 6.--Reaction-Path Calculations

In this example, the precipitation of phases as a result of incongruent dissolution of K-feldspar (microcline) is investigated. Only a limited set of phases--K-feldspar, gibbsite, kaolinite, and K-mica (muscovite)--is considered in this example. The reaction path for this set of phases was originally addressed by Helgeson and others (1969). In this example, the thermodynamic data for the phases (table 21, **PHASES** keyword) are derived from Robie and others (1978) and are the same as test problem 5 in the PHREEQE manual (Parkhurst and others, 1980).

PHREEQC can be used to solve this problem in three ways: the individual intersections of the reaction path and the phase boundaries on a phase diagram can be calculated (example 6A), the reaction path can be calculated incrementally (6B), or the reaction path can be calculated as a kinetic process (6C). In the first approach, no knowledge of the amounts of reaction is needed, but a number of simulations are necessary to find the appropriate phase-boundary intersections. In the second approach, only one simulation is sufficient, but the appropriate amounts of reaction must be known beforehand. In the third approach, a kinetic rate expression is used to calculate the reaction path, using a step-size adjusting algorithm which takes care of phase boundary transitions by automatically decreasing the time interval when necessary. Only the total time to arrive at the point of K-feldspar equilibrium is required. All three approaches are demonstrated in this example. PHREEQC implicitly contains all the logic of a complete reaction-path program (for example Helgeson and others, 1970, Wolery, 1979, Wolery and others, 1990). Moreover, the capability to calculate directly the phase boundary intersections provides an efficient way to outline reaction paths on phase diagrams, and the option to add the reaction incrementally and automatically find the stable phase assemblage allows points on the reaction path between phase boundaries to be calculated easily and rapidly. The kinetic approach and the Basic interpreter that is embedded in PHREEQC can be used to save and print the arrival time and the aqueous composition at each phase transition.

Conceptually, the example considers the reactions that would occur if K-feldspar were placed in a beaker and allowed to react slowly. As K-feldspar dissolves, other phases may begin to precipitate. In this example, it is assumed that only gibbsite, kaolinite, or K-mica can form, and that these phases will precipitate reversibly if they reach saturation. Phases precipitated at the beginning of the reaction may redissolve as the reaction proceeds.

Table 21.--Input data set for example 6

```
TITLE Example 6A.--React to phase boundaries.
SOLUTION 1   PURE WATER
             pH     7.0 charge
           temp    25.0
PHASES
        Gibbsite
                Al(OH)3 + 3 H+ = Al+3 + 3 H2O
                log_k          8.049
                delta_h        -22.792 kcal
```

```
        Kaolinite
                Al2Si2O5(OH)4 + 6 H+ = H2O + 2 H4SiO4 + 2 Al+3
                log_k           5.708
                delta_h         -35.306 kcal
        K-mica
                KAl3Si3O10(OH)2 + 10 H+ = 3 Al+3 + 3 H4SiO4 + K+
                log_k           12.970
                delta_h         -59.377 kcal
        K-feldspar
                KAlSi3O8 + 4 H2O + 4 H+ = Al+3 + 3 H4SiO4 + K+
                log_k           0.875
                delta_h         -12.467 kcal
SELECTED_OUTPUT
        -file           ex6A-B.sel
        -activities     K+ H+ H4SiO4
        -si             Gibbsite Kaolinite K-mica K-feldspar
        -equilibrium    Gibbsite Kaolinite K-mica K-feldspar
END
TITLE Example 6A1.--Find amount of K-feldspar dissolved to
                    reach gibbsite saturation.
USE solution 1
EQUILIBRIUM_PHASES 1
        Gibbsite        0.0     KAlSi3O8        10.0
        Kaolinite       0.0     0.0
        K-mica          0.0     0.0
        K-feldspar      0.0     0.0
END
TITLE Example 6A2.--Find amount of K-feldspar dissolved to
                    reach kaolinite saturation.
USE solution 1
EQUILIBRIUM_PHASES 1
        Gibbsite        0.0     0.0
        Kaolinite       0.0     KAlSi3O8        10.0
        K-mica          0.0     0.0
        K-feldspar      0.0     0.0
END
TITLE Example 6A3.--Find amount of K-feldspar dissolved to
                    reach K-mica saturation.
USE solution 1
EQUILIBRIUM_PHASES 1
        Gibbsite        0.0     0.0
        Kaolinite       0.0     0.0
        K-mica          0.0     KAlSi3O8        10.0
        K-feldspar      0.0     0.0
END
TITLE Example 6A4.--Find amount of K-feldspar dissolved to
                    reach K-feldspar saturation.
USE solution 1
EQUILIBRIUM_PHASES 1
        Gibbsite        0.0     0.0
        Kaolinite       0.0     0.0
        K-mica          0.0     0.0
        K-feldspar      0.0     KAlSi3O8        10.0
END
```

```
TITLE Example 6A5.--Find point with kaolinite present,
                    but no gibbsite.
USE solution 1
EQUILIBRIUM_PHASES 1
        Gibbsite        0.0     KAlSi3O8        10.0
        Kaolinite       0.0     1.0
END
TITLE Example 6A6.--Find point with K-mica present,
                    but no kaolinite
USE solution 1
EQUILIBRIUM_PHASES 1
        Kaolinite       0.0     KAlSi3O8        10.0
        K-mica          0.0     1.0
END
TITLE Example 6B.--Path between phase boundaries.
USE solution 1
EQUILIBRIUM_PHASES 1
        Kaolinite       0.0     0.0
        Gibbsite        0.0     0.0
        K-mica          0.0     0.0
        K-feldspar      0.0     0.0
REACTION 1
        K-feldspar      1.0
        0.04 0.08 0.16 0.32 0.64 1.0 2.0 4.0
        8.0 16.0 32.0 64.0 100 200 umol
END
TITLE Example 6C.--kinetic calculation
SOLUTION 1
        -units mol/kgw
        Al          1.e-13
        K           1.e-13
        Si          3.e-13
EQUILIBRIUM_PHASES 1
        Gibbsite    0.0  0.0
        Kaolinite   0.0  0.0
        K-mica      0.0  0.0
KINETICS 1
K-feldspar
#       k0 * A/V = 1e-16 mol/cm2/s * (10% fsp, 0.1mm cubes) 136/cm = 136.e-13 mol/dm3/s
        -parms 1.36e-11
        -m0 2.16
        -m  1.94
        -step_divide 1e-6
        -steps    1e2 1e3 1e4 1e5 1e6 1e7 1e8
INCREMENTAL_REACTIONS true
RATES
K-feldspar
-start
  10  REM store the initial amount of K-feldspar
  20  IF EXISTS(1) = 0 THEN PUT(M, 1)
  30  REM calculate moles of reaction
  40  SR_kfld = SR("K-feldspar")
  50  moles = PARM(1) * (M/M0)^0.67 * (1 - SR_kfld) * TIME
  60  REM The following is for printout of phase transitions
```

```
  80  REM      Start Gibbsite
  90  if ABS(SI("Gibbsite")) > 1e-3 THEN GOTO 150
  100   i = 2
  110   GOSUB 1500
  150 REM      Start Gibbsite -> Kaolinite
  160 if ABS(SI("Kaolinite")) > 1e-3 THEN GOTO 200
  170   i = 3
  180   GOSUB 1500
  200 REM      End Gibbsite -> Kaolinite
  210 if ABS(SI("Kaolinite")) > 1e-3 OR EQUI("Gibbsite") > 0 THEN GOTO 250
  220   i = 4
  230   GOSUB 1500
  250 REM      Start Kaolinite -> K-mica
  260 if ABS(SI("K-mica")) > 1e-3 THEN GOTO 300
  270   i = 5
  280   GOSUB 1500
  300 REM      End Kaolinite -> K-mica
  310 if ABS(SI("K-mica")) > 1e-3 OR EQUI("Kaolinite") > 0 THEN GOTO 350
  320   i = 6
  330   GOSUB 1500
  350 REM      Start K-mica -> K-feldspar
  360 if ABS(SI("K-feldspar")) > 1e-3 THEN GOTO 1000
  370   i = 7
  380   GOSUB 1500
  1000 SAVE moles
  1010 END
  1500 REM subroutine to store data
  1510 if GET(i) >= M THEN RETURN
  1520     PUT(M, i)
  1530     PUT(TOTAL_TIME, i, 1)
  1540     PUT(LA("K+")-LA("H+"), i, 2)
  1550     PUT(LA("H4SiO4"), i, 3)
  1560 RETURN
-end
USER_PRINT
  10 DATA "A: Gibbsite               ", "B: Gibbsite  -> Kaolinite ", \
          "C: Gibbsite  -> Kaolinite ", "D: Kaolinite -> K-mica     ", \
          "E: Kaolinite -> K-mica    ", "F: K-mica    -> K-feldspar"
  20 PRINT "      Transition             Time   K-feldspar     LA(K/H)   LA(H4SiO4)"
  30 PRINT "                                                  reacted"
  40 PRINT "                                                  (moles)"
  50 FOR i = 2 TO 7
  60   READ s$
  70   IF EXISTS(i) THEN PRINT s$, GET(i,1), GET(1) - GET(i), GET(i,2), GET(i,3)
  80 NEXT i
SELECTED_OUTPUT
       -file ex6C.sel
       -reset false
USER_PUNCH
  -head pH+log[K]  log[H4SiO4]
  10 PUNCH LA("K+")-LA("H+") LA("H4SiO4")
END
```

The input data set (table 21) first defines pure water with **SOLUTION** input and the thermodynamic data for the phases with **PHASES** input. Some of the minerals are defined in the database file (*phreeqc.dat*), but inclusion in the input data set replaces any previous definitions for the duration of the run (the database file is not altered). **SELECTED_OUTPUT** is used to produce a file of all the data that appear in table 22 and that were used to construct figure 6. **SELECTED_OUTPUT** specifies that the log of the activities of potassium ion, hydrogen ion, and silicic acid; the saturation indices for gibbsite, kaolinite, K-mica, and K-feldspar; and the total amounts in the phase assemblage and mole transfers for gibbsite, kaolinite, K-mica, and K-feldspar will be written to the file *ex6A-B.sel* after each calculation. The definitions for **SELECTED_OUTPUT** remain in effect for all simulations in the run, until a new **SELECTED_OUTPUT** data block is read, or until writing to the file is suspended with the identifier **-selected_output** in the **PRINT** data block.

Table 22. --Selected results for example 6A

[Simulation refers to labels in the input data set for example 6A. Negative mole transfers indicate dissolution, positive mole transfers indicate precipitation. Point on graph refers to labeled points on figure 6]

Simulation	K-feldspar mole transfer micromoles	Log activity			Mole transfer, micromoles			Saturation index				Point on graph
		H^+	$\frac{K^+}{H^+}$	H_4SiO_4	Gibbsite	Kaolinite	K-mica	Gibbsite	Kaolinite	K-mica	K-feldspar	
6A1	-0.03	-7.01	-0.57	-7.10	0.00	0.00	0.00	0.0	-3.8	-10.7	-14.7	A
6A2	-2.18	-8.21	2.55	-5.20	1.78	.00	.00	.0	0	-1.9	-5.9	B
6A3	-20.02	-9.11	4.41	-4.47	.00	9.71	.00	-.7	.0	.0	-2.5	D
6A4	-190.9	-9.39	5.49	-3.55	.00	.00	63.61	-2.0	-.7	.0	.0	F
6A5	-3.02	-8.35	2.83	-5.20	.00	1.24	.00	.0	0	-1.6	-5.6	C
6A6	-32.68	-9.07	4.41	-4.25	.00	.00	10.78	-.9	.0	.0	-2.1	E

Simulation 6A1 allows K-feldspar to react until equilibrium with gibbsite is reached. This is set up in **EQUILIBRIUM_PHASES** input by specifying equilibrium for gibbsite (saturation index equals 0.0) and an alternative reaction to reach equilibrium, KAlSi3O8 (the formula for K-feldspar). A large amount of K-feldspar (10.0 mol) is present to assure equilibrium with gibbsite can be obtained. Kaolinite, K-mica, and K-feldspar are allowed to precipitate if they become saturated, but they can not dissolve because they were given zero initial moles in the phase assemblage. The amount of reaction that is calculated in this simulation is the dissolution of precisely enough K-feldspar to reach equilibrium with gibbsite, possibly including precipitation of one or more of the other minerals. No gibbsite will dissolve or precipitate; the alternative reactant (KAlSi3O8) will dissolve or precipitate in its place. Simulations 6A2-6A4 perform the same calculations for kaolinite, K-mica, and K-feldspar. At other temperatures or using other minerals, it may be that a target phase is undersaturated regardless of the amount of the alternative reaction that is added because the phase is unstable relative to other phases. In this case, the numerical method will find the amount of the alternative reaction that produces the maximum saturation index.

Selected results for simulations 6A1-6A4 are presented in table 22 and are plotted on figure 6 as points A, B, D, and F. The stability fields for the phases, which are based on the thermodynamic data, are outlined on the figure

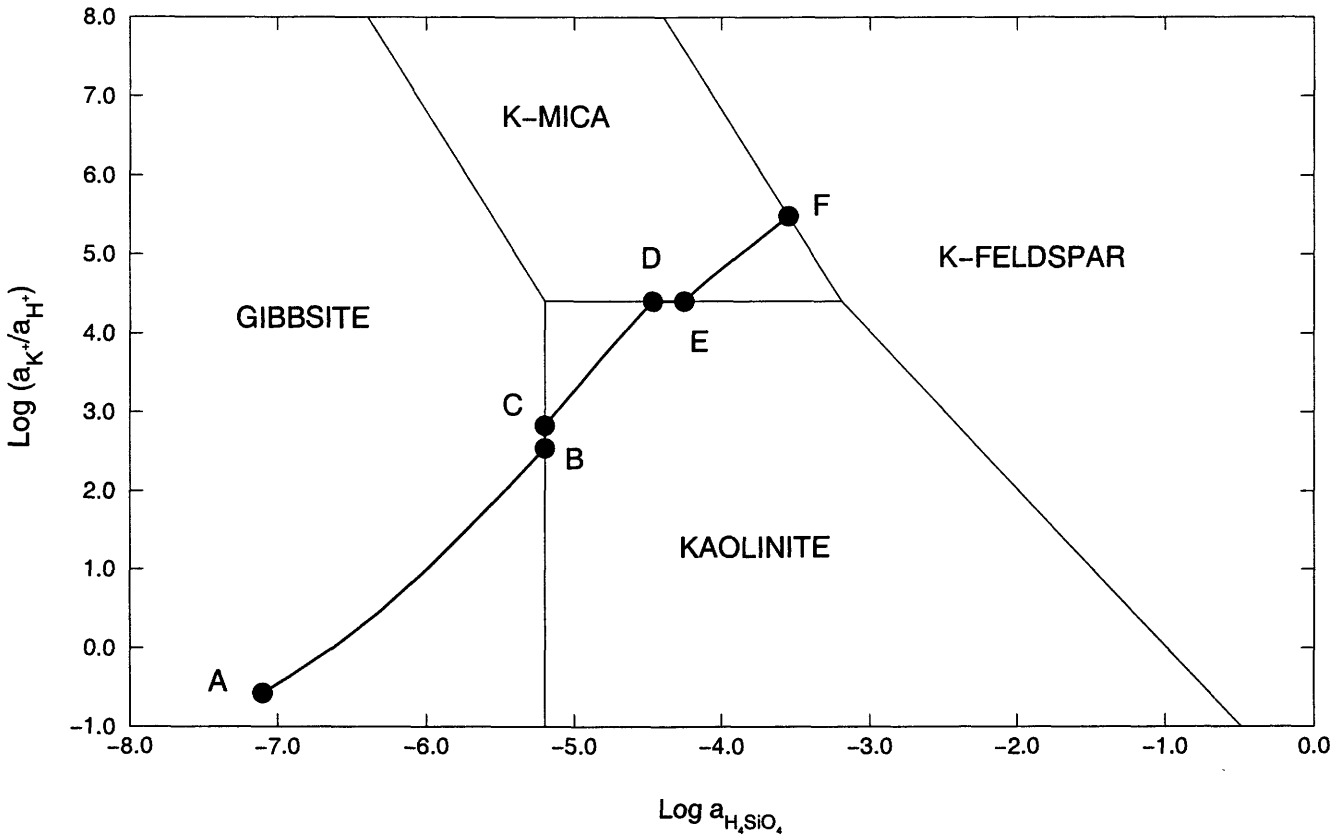

Figure 6.--Phase diagram for the dissolution of K-feldspar (microcline) in pure water at 25°C showing stable phase-boundary intersections (example 6A) and reaction paths across stability fields (example 6B). Diagram was constructed using thermodynamic data for gibbsite, kaolinite, K-mica (muscovite), and microcline from Robie and others (1978).

but are not calculated by the modeling in these simulations. At point B gibbsite starts to be transformed into kaolinite, a reaction which consumes Si. The reaction path must follow the gibbsite-kaolinite phase boundary to some intermediate point C until all gibbsite is converted, and then the path crosses the kaolinite field to point D. Similarly, there is a point E on the kaolinite-K-mica phase boundary, where the reaction path starts crossing the K-mica field to point F. Simulations 6A5 and 6A6 (table 21) solve for these two points. In simulation 6A5, point C is calculated by allowing K-feldspar to dissolve to a point where kaolinite is at saturation and is present in the phase assemblage, while gibbsite is at saturation, but not present in the phase assemblage. Likewise, simulation 6A6 solves for the point where K-mica is at saturation and present in the phase assemblage, while kaolinite is at saturation, but is not present in the phase assemblage. Assigning an initial amount of 1 mol to kaolinite in 6A5 and K-mica in 6A6 is arbitrary; the amount must only be sufficient to reach equilibrium with the mineral.

A simpler approach to determining the reaction path is to react K-feldspar incrementally, allowing the stable phase assemblage among gibbsite, kaolinite, K-mica, and K-feldspar to form at each point along the path. The only difficulty in this approach is to know the appropriate amounts of reaction to add. From points A and F in table 22, K-feldspar dissolution ranges from 0.03 to 190.9 mmol. In part 6B (table 21) a logarithmic range of reaction increments is used to define the path (solid line) across the phase diagram from its beginning at gibbsite equilibrium (point A) to equilibrium with K-feldspar (point F). However, the exact locations of points A through F will not be

determined with the arbitrary set of reaction increments that are used in part B. The reaction path calculated by part 6B is plotted on the phase diagram in figure 6 with points A through F from part 6A included in the set of points.

Finally, in the kinetic approach, kinetic dissolution of K-feldspar is followed for varying amounts of time, while the phases gibbsite, kaolinite, and K-mica are allowed to precipitate and redissolve as the kinetic reaction proceeds. **SOLUTION** 1 is defined to have a small amount of dissolved K-feldspar (1e-13 moles). The solution then contains all elements related to phases in **EQUILIBRIUM_PHASES**, which, although not required for the program to run successfully, eliminates some warning messages.

During the integration of the reaction rates a simple dissolution rate law was assumed based on transition-state theory:

$$R_{K-feldspar} = k_1 \frac{A}{V}\left(\frac{m}{m_0}\right)^{0.67}\left(1-\left(\frac{IAP}{K}\right)_{K-feldspar}\right) \tag{158}$$

with k_1 = 1e-16 mol/cm^2/s.

The **KINETICS** data block is used to enter specific data for the kinetic simulation. The stoichiometry of the kinetic reaction is the chemical formula of K-feldspar; by default the name of the rate is assumed to be a phase defined in **PHASES** data block and the formula of the phase is used as the stoichiometry of the reaction. It was assumed that the pristine soil contained 10 percent K-feldspar in the form of 0.1 mm cubes, and had ρ_b/θ = 6 g/cm^3, so that A/V = 136/cm. The value of $k_1\frac{A}{V}$ = 1.36e-11 mol/L/s is entered in the **KINETICS** data block with the identifier **-parms** (assuming that 1 kgw = 1 liter), and can be recalled as "PARM(1)" in the Basic rate definition in the **RATES** data block. It was assumed that the soil had already been weathered to some extent, and that only 90 percent of the initial K-feldspar was left [**-m0** 2.16 and **-m** 1.94, where **m0** indicates the initial mass (1 kg soil x 0.1 = 100 g / 278.3 g/mol = 0.359 mol/kg x 6 kg/L = 2.16 mol/L), and **m** the remaining mass (90 percent of 2.16 is 1.94 mol/L)]. The maximum amount of reaction for any time interval is restricted to 1e-6 moles (**-step_divide** 1e-6). Time steps (seconds) are defined with the identifier **-steps**. **INCREMENTAL_REACTIONS true** causes the total time simulated to be the sum of all of the time steps (1.111111e8 seconds); each new time step starts at end of the previous time step.

The rate for K-feldspar dissolution is defined in the form of Basic statements in the **RATES** data block. To demonstrate some of the features of the Basic interpreter, the Basic program also identifies and saves information at phase transitions, which is printed at the end of the run via **USER_PRINT**. The accuracy of locating a phase transition is determined by the user-definable accuracy of the integration. A small tolerance (**-tol**), a large **-step_divide** that is greater than 1 (initial time interval will be divided by this number), or a small **-step_divide** that is less than 1 (specifies maximum moles of reaction) will force smaller time intervals and more accurate identification of phase transitions. In this example, **-step_divide** is set to 1e-6, which limits the maximum amount of reaction for any time interval to be less than 1 micromole. Thus, the amount of reaction to reach a phase transition should be identified with an accuracy of one micromole. However, limiting the amount of reaction requires smaller time intervals during the integration and, consequently, more time intervals to complete the integration, which increases the CPU time of the run.

The functions of the different parts of the Basic program are described in table 23. The functions PUT, GET, and EXISTS are used to manipulate data in static, global storage. The subscripts used in the PUT statement identify a datum uniquely. EXISTS can be used to determine if a datum with a given set of subscripts has been stored, and GET is used to retrieve data that have been stored. Once a datum has been stored with PUT, it exists for the remainder

Table 23.--Description of Basic program for K-feldspar dissolution kinetics and identification of phase transitions

Line number	Function
20	Save initial amount of K-feldspar (1.94 moles)
40-50	Integrates K-feldspar dissolution rate over time interval given by TIME.
90-110	Identify greatest amount of K-feldspar present (least amount of reaction) at which gibbsite is saturated.
160-180	Identify greatest amount of K-feldspar present at which kaolinite is saturated.
200-230	Identify greatest amount of K-feldspar present at which kaolinite is saturated, but gibbsite is absent.
250-280	Identify greatest amount of K-feldspar present at which K-mica is saturated.
300-330	Identify greatest amount of K-feldspar present at which K-mica is saturated, but kaolinite is absent.
350-380	Identify greatest amount of K-feldspar present at which K-feldspar is saturated.
1000	Save integrated reaction.
1010	End of Basic program
1500-1560	Subroutine for saving values for phase transitions. If amount of K-feldspar is greater than current saved value for the index *i*, save amount of K-feldspar, cumulative time, log activity ratio of potassium ion divided by hydrogen ion, and log activity of silicic acid.

of the run, unless it is overwritten with another PUT statement with the same set of subscripts. Data stored with PUT can be retrieved by any Basic program, including those defined in **RATES**, **USER_PRINT**, and **USER_PUNCH**. In this example, data are stored by the **RATES** Basic program and the **USER_PRINT** Basic program retrieves the data and prints a summary of the phase transitions. While the **RATES** program is run many times during the kinetic integration of a time step (integration over many time intervals may be necessary for the required accuracy), the **USER_PRINT** program is run once at the end of each integration time step.

Table 24 gives the phase transitions encountered by the end of the last time step of example 6C. For each phase transition, the time at which the phase transition occurred, the total amount of K-feldspar that has reacted, and the coordinates of the transition on figure 6 are given. Although the values in table 24 are approximate, the amount of K-feldspar and the coordinates of the transition can be compared to table 22. As expected, the approximate mole transfers to reach the phase transitions are within 1 micromole of the values in table 22.

Table 24.--Phase transitions identified by the **RATES** Basic program and printed to the output file by the **USER_PRINT** Basic program in example 6C, which simulates the kinetic dissolution of K-feldspar

```
-----------------------------------User print-----------------------------------
       Transition                      Time   K-feldspar      LA(K/H)   LA(H4SiO4)
                                              reacted
                                              (moles)
A: Gibbsite                    1.10000E+03  1.40477E-07  3.57546E-01 -6.37631E+00
B: Gibbsite  -> Kaolinite      1.74339E+05  2.20636E-06  2.56086E+00 -5.19500E+00
C: Gibbsite  -> Kaolinite      2.39290E+05  3.02835E-06  2.83517E+00 -5.19434E+00
D: Kaolinite -> K-mica         1.58687E+06  2.00699E-05  4.40800E+00 -4.46585E+00
E: Kaolinite -> K-mica         2.59719E+06  3.27914E-05  4.41031E+00 -4.25093E+00
F: K-mica    -> K-feldspar     4.78404E+07  1.90721E-04  5.48792E+00 -3.55397E+00
```

The **SELECTED_OUTPUT** data block specifies that a new selected-output file will be used for this simulation, *ex6C.sel*, and all printing to the selected-output file is eliminated (**-reset** false). The **USER_PUNCH** data block causes two columns to be written to each line of the selected-output file, the log of the ratio of the activities of potassium ion to hydrogen ion and the log activity of silicic acid. The data will be written after each time step has been simulated (**-steps**, **KINETICS** data block). The following table shows the results written to *ex6C.sel*.

Table 25.--Results written to the selected-output file by the **USER_PUNCH** Basic program in example 6C, which simulates the kinetic dissolution of K-feldspar

["\t " indicates a tab character]

```
" pH+log[K]"\t   "log[H4SiO4"\t
-6.00016E+00\t   -1.25235E+01\t
-1.89754E+00\t   -8.42120E+00\t
-8.53163E-01\t   -7.37985E+00\t
 3.57546E-01\t   -6.37631E+00\t
 2.18230E+00\t   -5.38178E+00\t
 4.13600E+00\t   -4.60047E+00\t
 5.26137E+00\t   -3.71874E+00\t
 5.48836E+00\t   -3.55363E+00\t
```

Example 7.--Gas-Phase Calculations

This example demonstrates the capabilities of PHREEQC to model the evolution of gas compositions in equilibrium with an aqueous phase under the conditions of either fixed pressure or fixed volume of the gas phase. For a fixed-pressure gas phase, when the sum of the partial pressures of the component gases exceeds the specified pressure of the gas phase, a gas bubble forms. Once the bubble forms, the volume and composition of the gas bubble vary with extent of reactions. For a fixed-volume gas phase, the aqueous solution is in contact with a head space of fixed volume. The gas phase always exists in this head space and the pressure and composition of the gas phase vary with extent of reactions.

Gas-liquid reactions can be modeled in three ways with PHREEQC: (1) a gas can react to maintain a fixed partial pressure using **EQUILIBRIUM_PHASES** data block, (2) a fixed-pressure, multicomponent gas phase can be modeled using the **GAS_PHASE** data block with the **-fixed-pressure** identifier (default), or (3) a fixed-volume, multicomponent gas phase can be modeled using the **GAS_PHASE** data block with the **-fixed-volume** identifier. Conceptually, an infinite gas reservoir is assumed for the fixed-partial-pressure approach, as in the atmosphere or sometimes in the unsaturated zone. The partial pressure of the gas component in the reservoir does not change regardless of the extent of reactions. If the gas reservoir is finite and the pressure on the gas phase is constant, as in gas bubbles in estuarine and lake sediments, then a fixed-pressure gas phase is appropriate. If the gas reservoir is finite and the volume that the gas phase fills is constant, as in an experiment with a fixed head-space, then a fixed-volume gas phase is appropriate.

In this example, the **GAS_PHASE** data block is used to model the decomposition of organic matter in pure water under fixed-pressure and fixed-volume conditions, with the assumption that carbon, nitrogen, hydrogen, and oxygen are released in the stoichiometry $CH_2O(NH_3)_{.07}$ by the decomposition reaction. With no other electron

acceptors available in pure water, the pertinent microbiological decomposition reaction is methanogenesis. The carbon and nitrogen released by organic decomposition are assumed to react to redox and gas-solution equilibrium. Aqueous carbon species are defined in **SOLUTION_MASTER_SPECIES** or **SOLUTION_SPECIES** of the default databases for two valence states, carbon(+4) and carbon(-4) (methane); no intermediate valence states of carbon are defined. Aqueous nitrogen may occur in the +5, +3, 0, and -3 valence states. The gas components considered are carbon dioxide (CO_2), methane (CH_4), nitrogen (N_2), and ammonia (NH_3).

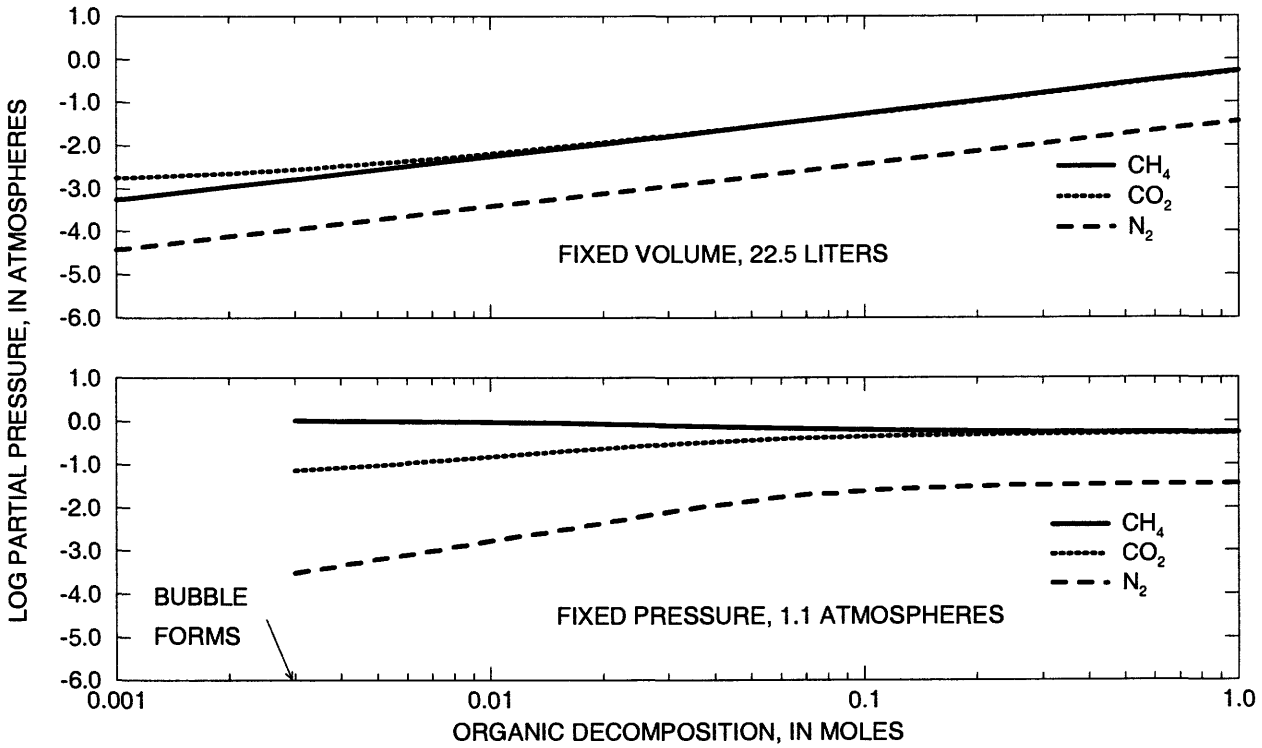

Figure 7.--Composition of the gas phase during decomposition of organic matter with a composition of $CH_2O(NH_3)_{0.07}$ in pure water under conditions of fixed volume and fixed pressure for the gas phase. [Partial pressure of ammonia gas is less than 10^{-7} atmospheres throughout (not shown).]

In the first simulation, the initial water is defined to be a ground water in equilibrium with calcite at a partial pressure of carbon dioxide of $10^{-1.5}$. Pure water is defined with the **SOLUTION** data block by using defaults for all values (pH = 7, pe = 4, temperature = 25°C); calcite and carbon dioxide are defined with **EQUILIBRIUM_PHASES**; and **SAVE** is used to save the equilibrated solution (table 26). **SELECTED_OUTPUT** is used to define a file (*ex7.sel*) to which specified data are written for each calculation. All default printing to the file is suspended with the identifier **-reset** false. The additional identifiers cause specific data items to be written to the selected-output file for each calculation in each simulation: **-simulation**, the simulation number; **-state**, the type of calculation (initial solution, reaction, transport, and others); **-reaction,** the amount of the reaction added for each calculation (as defined in the **REACTION** data block); **-si**, the saturation index of each mineral or log partial pressure of each gas that is specified; and **-gas**, the moles in the gas phase of each gas component that is specified.

In the second simulation, the organic decomposition reaction with a carbon to nitrogen ratio of approximately 15:1 is added irreversibly to the solution in increments ranging from 1 to 1000 mmol (**REACTION**

keyword). A gas phase, which initially has no moles of gas components present, is allowed to form if the sum of the partial pressures exceeds 1.1 atm; only CO_2, CH_4, N_2, and NH_3 are allowed in the gas phase (**GAS_PHASE** data block). In the third simulation, the same initial solution and reaction are used as in the second simulation. The gas phase initially has no moles of gas components present, but is defined to have a fixed volume of 22.5 L, which is the volume of the fixed-pressure gas phase after reaction of 1.0 mol of organic matter. After 1.0 mol of reaction, both the fixed-pressure and fixed-volume gas phases will have the same pressure, volume, and composition; at all other reaction increments the pressure, volume, and composition will differ between the two gas phases.

Table 26.--Input data set for example 7

```
TITLE Example 7.--Organic decomposition with fixed-pressure and
                  fixed-volume gas phases
SOLUTION 1
EQUILIBRIUM_PHASES 1
        Calcite
        CO2(g)  -1.5
SAVE solution 1
SELECTED_OUTPUT
        -reset false
        -file ex7.sel
        -simulation     true
        -state          true
        -reaction       true
        -si CO2(g) CH4(g) N2(g) NH3(g)
        -gas CO2(g) CH4(g) N2(g) NH3(g)
END
#  Simulation 2: Decomposition of organic matter, CH2O(NH3).07,
#  at fixed pressure of 1.1 atm
USE solution 1
GAS_PHASE 1 Fixed-pressure gas phase
        -fixed_pressure
        -pressure       1.1
        CO2(g)          0.0
        CH4(g)          0.0
        N2(g)           0.0
        NH3(g)          0.0
REACTION 1
        CH2O(NH3)0.07     1.0
        1. 2. 3. 4. 8. 16. 32 64. 125. 250. 500. 1000. mmol
END
#  Simulation 3: Decomposition of organic matter, CH2O(NH3).07,
#  at fixed volume of 22.5 L
USE solution 1
USE reaction 1
GAS_PHASE 1 Fixed volume gas phase
        -fixed_volume
        -volume         22.5
        CO2(g)          0.0
        CH4(g)          0.0
        N2(g)           0.0
        NH3(g)          0.0
END
```

For the fixed-pressure gas phase, a bubble forms when nearly 3 mmol of reaction have been added (fig. 7). Initially the gas is more than 90 percent CH_4 and less than 10 percent CO_2, with only minor amounts of N_2 and NH_3 (NH_3 partial pressures were less than 10^{-7} atm throughout the batch-reaction calculation and are not plotted). The volume of gas produced ranges from less than 1 mL at 3 mmol of reaction to 22.5 L after 1 mol of reaction. After 1 mol of reaction is added, nearly all of the carbon and nitrogen is in the gas phase.

For the fixed-volume gas phase, the gas phase exists for all amounts of organic decomposition (fig. 7). Initially the gas is primarily CO_2 with significant amounts of CH_4 and a small amount of N_2. As the gas composition evolves, CO_2 and CH_4 partial pressures become nearly equal. The N_2 partial pressure is always an order of magnitude smaller than CO_2 and CH_4; NH_3 partial pressures are always small (not shown). The partial pressures of the fixed-volume gas phases are smaller than the fixed pressure gas phase up to 1.0 mol of reaction. If the reaction continued beyond 1.0 mol, the pressure of the fixed-volume gas phase would continue to increase and would be greater than the pressure in the fixed-pressure gas phase, which remains constant. Conversely, the volume of the fixed-pressure gas phase is less than the volume of the fixed-volume gas phase until 1.0 mol of reaction. If the reaction continued beyond 1.0 mol, the volume of the fixed-pressure gas phase would exceed the volume of the fixed-volume gas phase.

Example 8.--Surface Complexation

PHREEQC contains three surface-complexation models: (1) By default, the generalized two-layer model is used with no explicit calculation of the diffuse-layer composition. (2) Alternatively, an electrostatic double layer model with explicit calculation of the diffuse-layer composition may be used (**-diffuse_layer**). (3) Finally, a non-electrostatic model may be selected (**-no_edl**). The electrostatic model is the generalized two-layer model described in Dzombak and Morel (1990) with the following modifications: (1) surfaces may have more than two types of binding sites, (2) surface precipitation is not included, and (3) optionally, an alternative formulation for the charge-potential relationship, modified from Borkovec and Westall (1983), that explicitly calculates the composition of the diffuse layer can be employed (**-diffuse_layer**). The non-electrostatic model does not consider the effects of the development of surface charge on the formation of surface complexes, with the result that surface complexes are treated mathematically very much like aqueous complexes without activity coefficient terms.

The following example of the generalized two-layer model is taken from Dzombak and Morel (1990, chapter 8) with no explicit calculation of the diffuse-layer composition. Zinc sorption on hydrous ferric oxide is simulated assuming two types of sites, weak and strong, are available on the oxide surface. Protons and zinc ions compete for the two types of binding sites, and equilibrium is described by mass-action equations. Activities of the surface species depend on the potential at the surface, which is due to the development of surface charge. The example considers the variation in sorption of zinc on hydrous ferric oxides as a function of pH for low zinc concentration (10^{-7} m) and high zinc concentration (10^{-4} m) in 0.1 m sodium nitrate electrolyte.

Three keyword data blocks are required to define surface-complexation data for a simulation: **SURFACE_MASTER_SPECIES**, **SURFACE_SPECIES**, and **SURFACE**. The **SURFACE_MASTER_SPECIES** data block in the default database files defines a binding site named "Hfo" (hydrous ferric oxides) with two binding sites. The name of a binding site is composed of a name for the surface, "Hfo", optionally followed by an underscore and a lowercase binding site designation, "Hfo_w" and "Hfo_s" for "weak" and "strong" in the database files. The underscore notation is necessary only if two or more binding sites

exist for a single surface. The notation allows a mole-balance equation to be derived for each of the binding sites (Hfo_w and Hfo_s, in this example) and a single charge-potential or charge-balance equation for the surface (Hfo, in this example). Thus, the charge that develops on each binding site will enter into a single charge-potential or charge-balance equation for the surface.

Table 27.--Input data set for example 8

```
TITLE Example 8.--Sorption of zinc on hydrous
                  iron oxides.
SURFACE_SPECIES
        Hfo_sOH  + H+ = Hfo_sOH2+
        log_k  7.18

        Hfo_sOH = Hfo_sO- + H+
        log_k  -8.82

        Hfo_sOH + Zn+2 = Hfo_sOZn+ + H+
        log_k  0.66

        Hfo_wOH  + H+ = Hfo_wOH2+
        log_k  7.18

        Hfo_wOH = Hfo_wO- + H+
        log_k  -8.82

        Hfo_wOH + Zn+2 = Hfo_wOZn+ + H+
        log_k  -2.32
SURFACE 1
        Hfo_sOH         5e-6    600.    0.09
        Hfo_wOH         2e-4
SOLUTION 1
        -units  mmol/kgw
        pH      8.0
        Zn      0.0001
        Na      100.    charge
        N(5)    100.
SOLUTION 2
        -units  mmol/kgw
        pH      8.0
        Zn      0.1
        Na      100.    charge
        N(5)    100.
USE solution none
#
# Model definitions
#
PHASES
        Fix_H+
        H+ = H+
        log_k  0.0
END
#
#   Zn = 1e-7
```

```
#
SELECTED_OUTPUT
        -file ex8.sel
        -molalities     Zn+2    Hfo_wOZn+       Hfo_sOZn+
USE solution 1
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+   -5.0   NaOH    10.0
END
USE solution 1
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+   -5.25  NaOH    10.0
END
USE solution 1
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+   -5.5   NaOH    10.0
END
USE solution 1
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+   -5.75  NaOH    10.0
END
USE solution 1
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+   -6.0   NaOH    10.0
END
USE solution 1
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+   -6.25  NaOH    10.0
END
USE solution 1
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+   -6.5   NaOH    10.0
END
USE solution 1
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+   -6.75  NaOH    10.0
END
USE solution 1
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+   -7.0   NaOH    10.0
END
USE solution 1
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+   -7.25  NaOH    10.0
END
```

```
USE solution 1
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+    -7.5    NaOH     10.0
END
USE solution 1
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+    -7.75   NaOH     10.0
END
USE solution 1
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+    -8.0    NaOH     10.0
END
#
#    Zn = 1e-4
#
USE solution 2
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+    -5.0    NaOH     10.0
END
USE solution 2
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+    -5.25   NaOH     10.0
END
USE solution 2
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+    -5.5    NaOH     10.0
END
USE solution 2
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+    -5.75   NaOH     10.0
END
USE solution 2
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+    -6.0    NaOH     10.0
END
USE solution 2
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+    -6.25   NaOH     10.0
END
USE solution 2
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+    -6.5    NaOH     10.0
END
USE solution 2
```

```
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+    -6.75  NaOH     10.0
END
USE solution 2
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+    -7.0   NaOH     10.0
END
USE solution 2
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+    -7.25  NaOH     10.0
END
USE solution 2
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+    -7.5   NaOH     10.0
END
USE solution 2
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+    -7.75  NaOH     10.0
END
USE solution 2
USE surface 1
EQUILIBRIUM_PHASES 1
        Fix_H+    -8.0   NaOH     10.0
END
```

Surface-complexation reactions derived from the summary of Dzombak and Morel (1990) are defined by the **SURFACE_SPECIES** in the default database files for PHREEQC. However, the intrinsic stability constants used in this example of Dzombak and Morel (1990, chapter 8) differ from their summary values, and are therefore specified explicitly with a **SURFACE_SPECIES** data block in the input file (table 27). The mass-action equations taken from Dzombak and Morel (1990, p. 259) are given in the input data set (table 27). Note the activity coefficient or potential term is not included as part of the mass-action expression; the potential term is added internally by the program.

The composition and other characteristics of an assemblage of surfaces is defined with the **SURFACE** data block. The composition of multiple surfaces, each with multiple binding sites, may be defined within this data block. For each surface, the moles of each type of site, the initial composition of the surface, and the surface area must be defined. The composition of the surfaces will vary with the extent of reactions. The number of binding sites and surface areas, may remain fixed or may vary if the surface is related to the moles of an equilibrium phase or a kinetic reaction. In this example, one surface (Hfo) with two binding sites (Hfo_w and Hfo_s) is defined and the number of binding sites and surface area are fixed. The number of moles of strong binding sites, Hfo_s, is 5×10^{-6} sites and the number of moles of weak binding sites, Hfo_w, is 2×10^{-4}. Initially, all surface sites are in the uncharged, protonated form. The surface area for the entire surface, Hfo, must be defined with two numbers, the area per mass of surface material (600 m^2/g, in this example) and the total mass of surface material (0.09 g, in this example). The use of these two numbers to define surface area is traditional, but only the product of these numbers

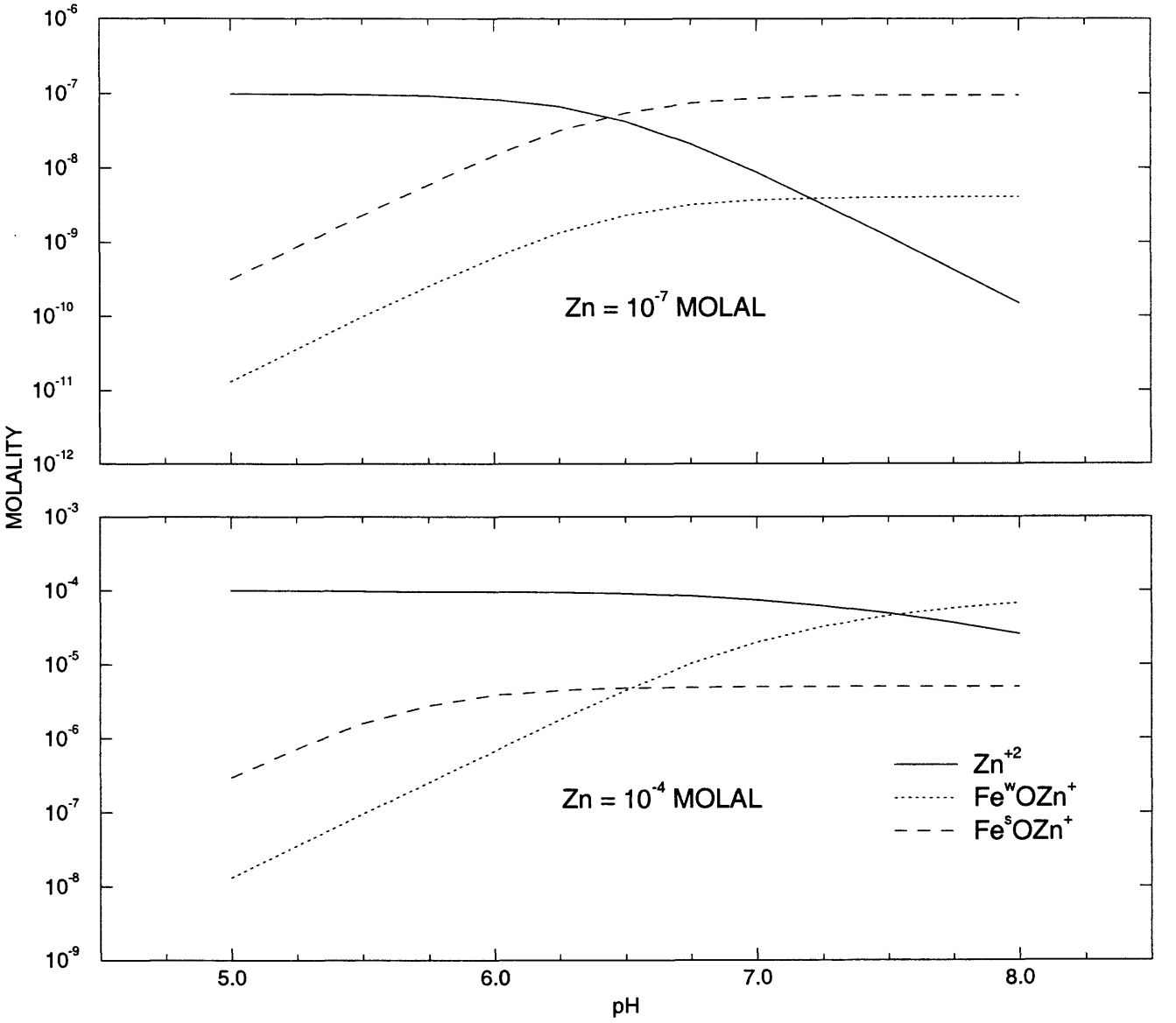

Figure 8.--Distribution of zinc among the aqueous phase and strong and weak surface sites of hydrous iron oxide as a function of pH for total zinc concentrations of 10^{-7} and 10^{-4} molal.

is used in the model to obtain the surface area; the individual numbers are not used separately. Surface area may be entered with the data for any of the binding sites for a surface; in this example, the surface area is entered with Hfo_s.

To complete the definition of the initial conditions for the simulations, two sodium nitrate solutions are defined with differing concentrations of zinc (**SOLUTION** 1 and 2 data blocks). A pseudo-phase, "Fix_H+" is defined with the **PHASES** data block. This phase is not real, but is used in each of the batch-reaction simulations to adjust pH to fixed values. Finally, the line "**USE** surface none" eliminates an implicitly defined batch-reaction calculation for the first simulation. By default, if a **SOLUTION** and **SURFACE** data block are defined in a simulation, then the first solution defined in the simulation (**SOLUTION** 1 in this example) and the first surface defined in the simulation are put together (possibly with other assemblages and a gas phase) and allowed to equilibrate. The **USE** keyword with

"solution none" removes the solution from the system for the batch-reaction calculation, with the effect that no batch-reaction calculation is performed. (A batch reaction is implicitly defined whenever a solution or mixture is defined in the simulation and any one of the keyword data blocks, **EXCHANGE**, **EQUILIBRIUM_PHASES**, **GAS_PHASE**, **KINETICS**, **REACTION**, **REACTION_TEMPERATURE**, **SOLID_SOLUTIONS**, or **SURFACE**, also is defined in the same simulation.)

The remaining simulations in the input data set equilibrate the surface assemblage with either solution 1 or solution 2 for pH values that range from 5 to 8. It would be possible to use the **REACTION** data block to add varying amounts of NaOH to a solution in a single simulation, but the reaction increments would not produce evenly spaced pH values and the size of the reaction increments is not known beforehand. In the example, a different approach is taken that produces evenly spaced pH values with no previous knowledge of the amount of NaOH required, but many simulations are needed to produce all of the desired pH values. Each of the simulations uses the phase "Fix_H+" in an **EQUILIBRIUM_PHASES** data block with varying saturation indices to adjust pH. The reaction NaOH is added or removed from each solution to produce a specified saturation index which, by the definition of the reaction for "Fix_H+" is numerically equal to the log of the hydrogen activity, or negative pH. Note that, although it is possible to attain the desired pH in all of these simulations, a pH that is sufficiently low will cause the program to fail because a very low pH can not be reached even by removing all of the sodium in solution.

The results of the simulation are plotted in figure 8 and are consistent with the results shown in Dzombak and Morel (1990, figure 8.9). Zinc is more strongly sorbed at high pH values than at low pH values. In addition, at low concentrations of zinc, the strong binding sites outcompete the weak binding sites for zinc over the entire pH range, and at high pH most of the zinc resides at the strong binding sites. At larger zinc concentrations, the strong binding sites predominate only at low pH. Because all the strong binding sites become filled at higher pH, most of the zinc resides at the more numerous weak binding sites at high pH and large zinc concentrations.

Example 9.--Kinetic Oxidation of Dissolved Ferrous Iron with Oxygen

Kinetic rate expressions can be defined in a completely general way in PHREEQC using Basic statements in the **RATES** data block. The rate expressions can be used in batch-reaction or transport calculations by using the **KINETICS** data block. For transport calculations (**ADVECTION** or **TRANSPORT**), kinetic reactions can be defined cell by cell by the number range following the **KINETICS** keyword (**KINETICS** *m-n*). The rate expressions are integrated with an embedded 4th- and 5th-order Runge-Kutta-Fehlberg algorithm. Equilibrium is calculated before a kinetic calculation is initiated and again when a kinetic reaction increment is added. Equilibrium is calculated for all solution-species, and for all exchange, equilibrium-phase, solid-solutions, surface assemblages and gas phases that have been defined. A check is performed to ensure that the difference between the 4th- and 5th-order estimates of the integrated rate over a time interval does not vary by more than a user-defined tolerance. If the tolerance is not satisfied, then the integration over the time interval is automatically restarted with a smaller time interval.

Kinetic reactions between solids and the aqueous phase can be calculated without any modification of the database. PHREEQC can also calculate kinetic reactions among aqueous species that are normally assumed to be in equilibrium, but this requires that the database be redefined. Aqueous species that react kinetically must be defined essentially as separate elements with **SOLUTION_MASTER_SPECIES**. This example illustrates the procedure

for decoupling two valence states of an element (iron) and shows how PHREEQC can be used to calculate the kinetic oxidation of Fe^{2+} to Fe^{3+} in water.

The rate of oxidation of Fe^{2+} by O_2 in water is given by (Singer and Stumm, 1970):

$$\frac{dm_{Fe^{2+}}}{dt} = -\left(2.91e\text{-}9 + 1.33e12\ a_{OH^-}^2 P_{O_2}\right) m_{Fe^{2+}}, \quad (159)$$

where t is time in seconds, a_{OH^-} is the activity of the hydroxyl ion, $m_{Fe^{2+}}$ is the total molality of ferrous iron in solution, and P_{O_2} is the oxygen partial pressure (atm).

The time for complete oxidation of ferrous iron is a matter of minutes in an aerated solution when pH is above 7.0. However, Fe^{3+} forms solute complexes with OH^- and it may also precipitate as iron oxyhydroxides, so that pH decreases during oxidation. Because the rate has quadratic dependence on the activity of OH^-, the oxidation rate rapidly diminishes as pH decreases. The rate equation is highly non-linear in an unbuffered solution and must be integrated numerically. This example models a reaction vessel with 10 mmol NaCl / kgw and 0.1 mmol $FeCl_2$ / kgw at pH = 7.0 through which air is bubbled; the change in solution composition over time is calculated.

The calculation requires the uncoupling of equilibrium among the Fe(2) and Fe(3) species. Two new "elements" are defined in **SOLUTION_MASTER_SPECIES**--"Fe_di", which corresponds to Fe(2), and "Fe_tri", which corresponds to Fe(3). The master species for these elements are defined to be Fe_di+2 and Fe_tri+3, and all solution species, phases, exchange species, and surface species must be rewritten using these new elements and master species. A few of the transcriptions are shown in table 28, which gives the partial input file for this example.

Table 28.--Partial input data set for example 9

```
TITLE Example 9.--Kinetically controlled oxidation of ferrous
                  iron. Decoupled valence states of iron.
SOLUTION_MASTER_SPECIES
Fe_di              Fe_di+2    0.0     Fe_di              55.847
Fe_tri             Fe_tri+3   0.0     Fe_tri             55.847
SOLUTION_SPECIES
Fe_di+2 = Fe_di+2
        log_k   0.0
Fe_tri+3 = Fe_tri+3
        log_k   0.0
#
# Fe+2 species
#
Fe_di+2 + H2O = Fe_diOH+ + H+
        log_k   -9.5
        delta_h 13.20   kcal
#
#... and also other Fe+2 species
#
#
# Fe+3 species
#
Fe_tri+3 + H2O = Fe_triOH+2 + H+
        log_k   -2.19
        delta_h 10.4    kcal
#
#... and also other Fe+3 species
```

```
#
PHASES
Goethite
        Fe_triOOH + 3 H+ = Fe_tri+3 + 2 H2O
        log_k   -1.0
END
SOLUTION 1
        pH  7.0
        pe 10.0  O2(g) -0.67
        Fe_di  0.1
        Na  10.
        Cl  10.  charge
EQUILIBRIUM_PHASES 1
        O2(g)           -0.67
RATES
Fe_di_ox
-start
10  Fe_di = TOT("Fe_di")
20  if (Fe_di <= 0) then goto 200
30  p_o2 = 10^(SI("O2(g)"))
40  moles = (2.91e-9 + 1.33e12 * (ACT("OH-"))^2 * p_o2) * Fe_di * TIME
200 SAVE moles
-end
KINETICS 1
Fe_di_ox
        -formula  Fe_di  -1.0  Fe_tri  1.0
       -steps 100 400 3100 10800 21600 5.04e4 8.64e4 1.728e5 1.728e5 1.728e5 1.728e5
INCREMENTAL_REACTIONS true
SELECTED_OUTPUT
        -file ex9.sel
        -reset false
USER_PUNCH
-headings Days  Fe(2)  Fe(3)  pH  si_goethite
10 PUNCH SIM_TIME/3600/24 TOT("Fe_di")*1e6, TOT("Fe_tri")*1e6, -LA("H+"),
SI("Goethite")
END
```

The **SOLUTION** data block defines a sodium chloride solution that has 0.1 mmol/kgw ferrous iron (Fe_di) and is in equilibrium with atmospheric oxygen. The **EQUILIBRIUM_PHASES** phases data block specifies that all batch-reaction solutions will also be in equilibrium with atmospheric oxygen; thus, there is a continuous supply of oxygen for oxidation of ferrous iron.

In the **RATES** data block, the rate expression is designated with the name "Fe_di_ox", and defined according to equation 159. Note the use of the special Basic function "TOT" to obtain the total concentration (molality) of ferrous iron (line 10), "SI" to obtain the saturation index, or, in the case of a gas, the log of the gas partial pressure (oxygen, line 30), and "ACT" to obtain the activity of OH^- (line 40). Line 40 defines the moles of reaction. Notice also that the variable *moles* is calculated by multiplying the rate times the current time interval (TIME) and that the rate definition ends with a SAVE statement. The SAVE and TIME statements must be included in a rate definition; they specify the moles that reacted over the time (sub-)interval.

In the **KINETICS** data block, the rate expression named "Fe_di_ox" is invoked and parameters are defined. When the rate name in the **KINETICS** data block is identical to a mineral name that is defined under **PHASES**,

the stoichiometry of that mineral will be used in the reaction. However, no mineral is associated with the rate name of this example, and the identifier **-formula** must be used to specify the reaction stoichiometry. The reaction involves loss of Fe_di [equivalent to Fe(2)] from solution as indicated by the stoichiometric coefficients of -1.0. The loss is balanced by a gain in solution of Fe_tri [equivalent to Fe(3)] with a stoichiometric coefficient of +1.0. Note that the formula only contains the elements for which the mass changes in the system. Thus, the overall kinetic reaction of the example is $Fe^{2+} + H^{+} + 0.25O_2 = Fe^{3+} + 0.5H_2O$, but the reaction of protons and oxygen to form water does not change the total mass of hydrogen or oxygen in the system. Hydrogen and oxygen are therefore not included in the formula. In the example, oxygen is replenished by equilibrium with atmospheric O2(g), and a mole transfer of oxygen does occur from the phase O2(g) in **EQUILIBRIUM_PHASES** into the solution. In a system closed to oxygen, the dissolved oxygen would be partly consumed.

The identifier **-steps** in the **KINETICS** data block gives the time step(s) over which the kinetic reactions must be integrated. When **INCREMENTAL_REACTIONS true** is used, each time step increments the total time to be simulated and the results from the previous time step are used as the starting point for the current time step.

The **SELECTED_OUTPUT** data block specifies the file name of the selected-output file and eliminates all default printing to that file (**-reset** false). The only output to the selected-output file in this example is defined with the **USER_PUNCH** data block. The Basic program in **USER_PUNCH** specifies that the following be printed after each kinetic time step (**-steps** defines 11 kinetic time steps): the cumulative time of the simulation, in days; the total ferrous and ferric iron, in μmol/kgw; the pH; and the saturation index of goethite.

When the input file is run, two warning messages are generated during the integration. If the integration time interval is too large, it is possible that the initial estimates of kinetic reaction increments produce negative solution concentrations. When this happens, the program prints a warning message, decreases the size of the time interval, and restarts the integration. The messages are warnings, not errors, and the program successfully completes the calculation. It is possible to eliminate the warning messages by reducing the initial integration interval. No warning messages are printed if the identifier **-step_divide** 100 is used (**KINETICS**), which divides the initial (overall) time step by 100. Likewise, no warning messages are printed if the identifier **-step_divide** 1e-7 is used, which causes the reaction increment to be less than 1e-7 mol. The former approach, with **-step_divide** 100, is usually preferable because, although initial reaction increments are compelled to be small, later on in the integration large reaction increments are possible. Using **-step_divide** 1e-7 forces reaction increments to remain small throughout the entire integration, and in this example, the run time is about 5 times longer than using **-step_divide** 100, and about 10 times longer than when **-step_divide** is not used at all.

Figure 9 shows the concentration of total Fe(2), total Fe(3), and pH in the reaction vessel over the 10 days of the simulation. It can be seen that the pH rapidly decreases at the beginning of the reaction. The slope of Fe(2) against time is initially very steep, but lessens as the reaction progresses, which is consistent with equation 159. When the experiment is performed in reality in an unbuffered solution, it is noted that the pH initially rises. This rise in pH is consistent with slowly forming hydroxy-complexes of Fe(3). Because the oxidation reaction by itself consumes protons, the pH would initially rise if the hydroxy-complexes that lower the pH form slowly. Such kinetic formation of aqueous complexes could also be included in PHREEQC simulations, but it would require that the hydroxy-complexes of Fe(3) also be defined using a separate **SOLUTION_MASTER_SPECIES** and that a rate expression be defined for the kinetic formation of the complexes.

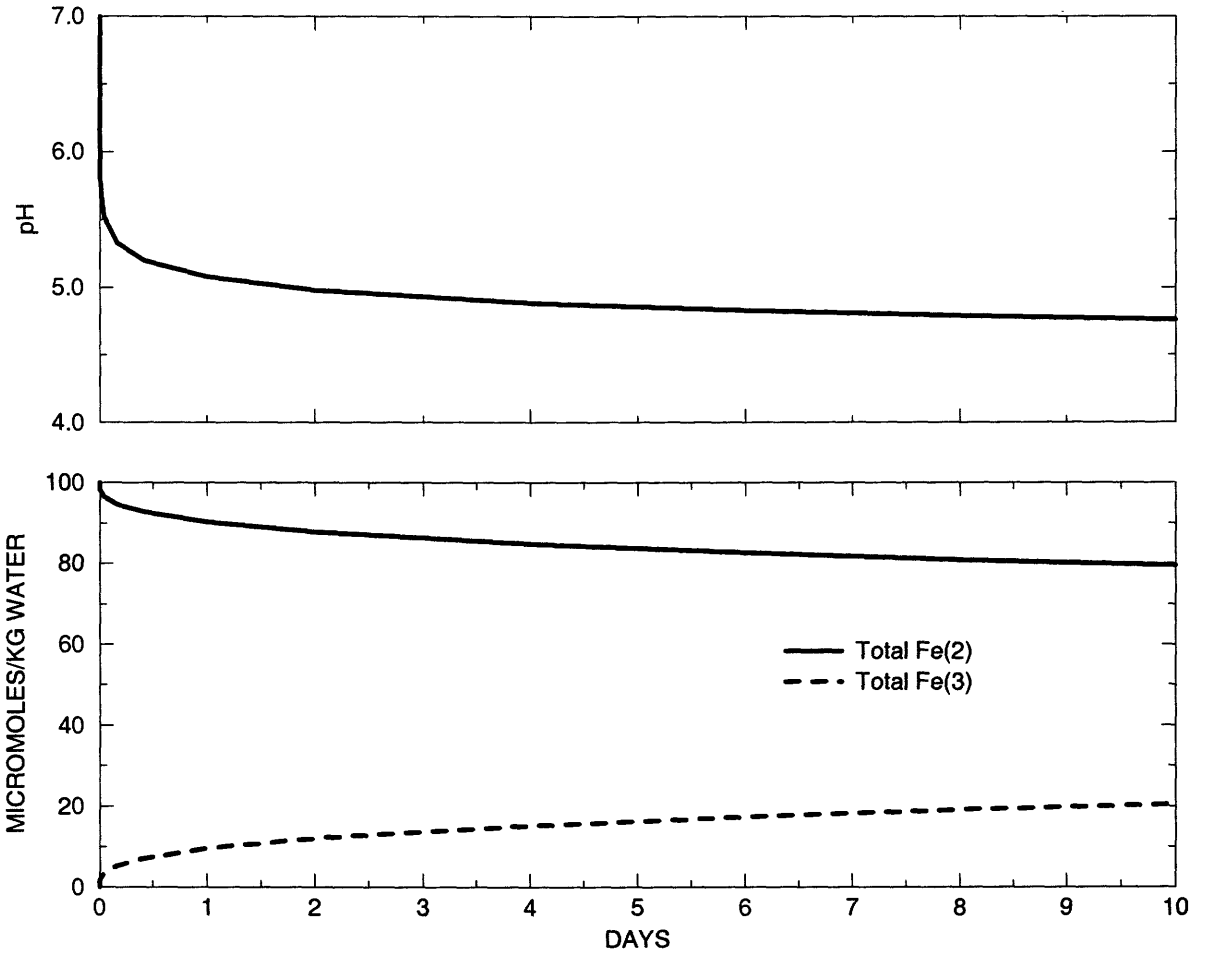

Figure 9.--Concentration of total Fe(2), total Fe(3), and pH as dissolved ferrous iron [Fe(2)] is kinetically oxidized to ferric iron [Fe(3)] by oxygen.

Example 10.--Aragonite-Strontianite Solid Solution

PHREEQC has the capability to model ideal, multicomponent or nonideal, binary solid solutions. For ideal solid solutions, the activity of each end-member solid is equal to its mole fraction. For nonideal solid solutions, the activity of each end member is the product of the mole fraction and an activity coefficient, which is determined from the mole fraction and Guggenheim excess free-energy parameters. The following example considers an aragonite ($CaCO_3$)-strontianite ($SrCO_3$) solid solution and demonstrates how the composition of the solid solution and aqueous phase change as strontium carbonate is added to an initially pure calcium carbonate system.

The example is derived from a diagram presented in Glynn and Parkhurst (1992). The equilibrium constants at 25 °C, $K_{SrCO_3} = 10^{-9.271}$ and $K_{CaCO_3} = 10^{-8.336}$, and the Guggenheim parameters, $a_0 = 3.43$ and $a_1 = -1.82$, are derived from Plummer and Busenberg (1987). The input data set is shown in table 29. The **PHASES** data block defines the log *K*'s for aragonite and strontianite and overrides any data for these minerals that might be present in the database file. The **SOLID_SOLUTIONS** data block defines the unitless Guggenheim excess free-energy parameters and the initial composition of the solid solution, which is zero moles of aragonite

and strontianite. Initial solution 1 is defined to be a calcium bicarbonate solution. The solution is then equilibrated with aragonite at nearly 1 atm partial pressure of carbon dioxide and saved as the new composition of solution 1.

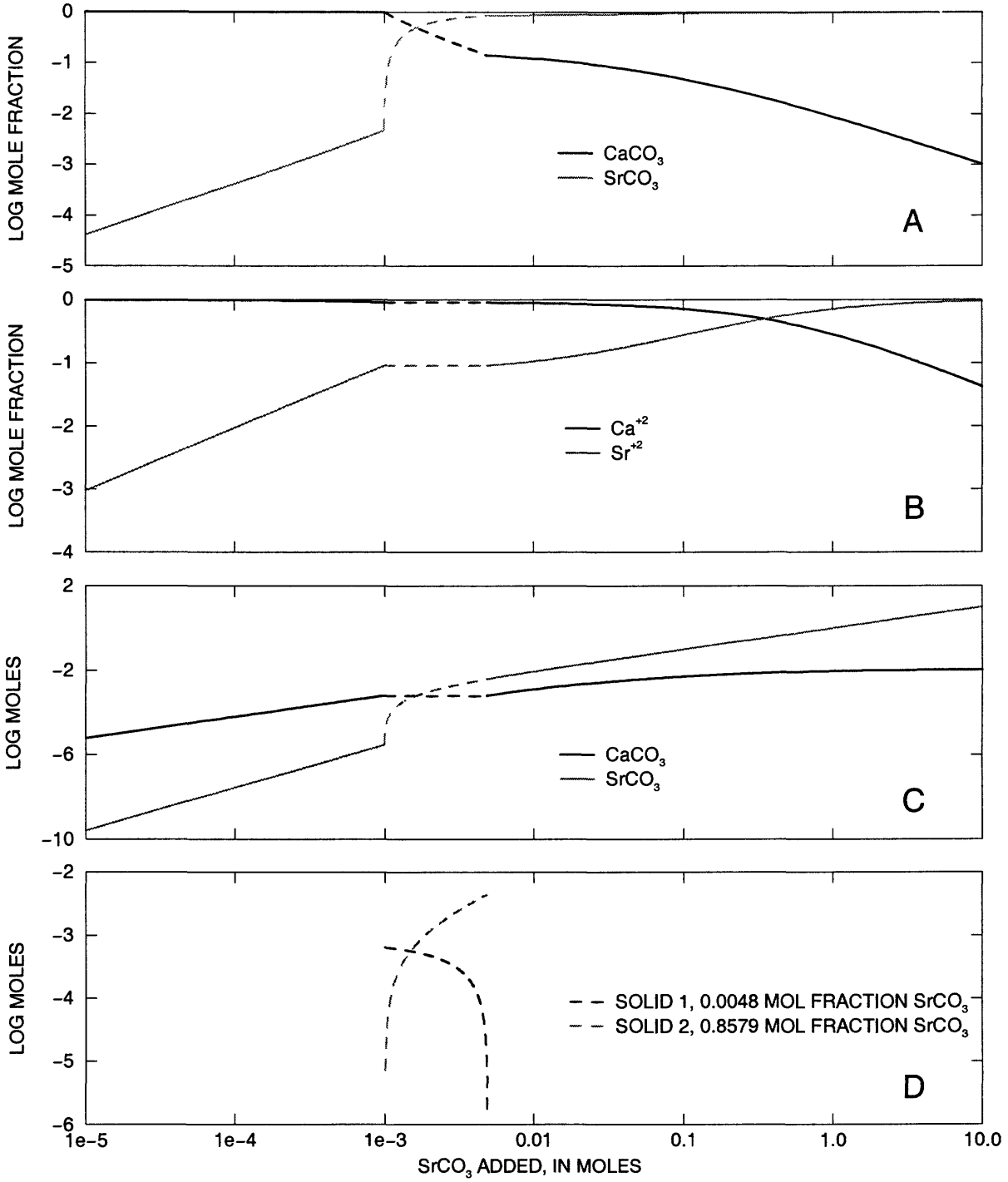

Figure 10.--(A) Mole fraction of strontianite and aragonite in solid solution, (B) mole fraction of calcium and strontium in aqueous phase, (C) moles of strontianite and aragonite in solid solution, and (D) moles of miscibility-gap end members in solid solution, as a function of the amount of strontium carbonate added. Dashed lines indicate compositions within the miscibility gap.

In the next simulation, solution 1 is brought together with the solid solution (**USE** keywords) and 5 millimoles of strontium carbonate are added in 500 steps (**REACTION** data block). The **PRINT** keyword data block excludes all default printing to the output file and includes only the printing defined in the **USER_PRINT** data block. The **USER_PRINT** data block specifies that the following information about the solid solution be printed to the output file after each reaction step: the simulation number, reaction-step number, amount of strontium carbonate added, $\log(\sum \Pi)$, mole fractions of strontianite and aragonite, aqueous mole fractions of calcium and strontium, and the composition of the two solids that exist within the miscibility gap. The **SELECTED_OUTPUT** data block defines the selected-output file to be *ex10.sel*, cancels any default printing to the selected-output file (**-reset** false), and requests that the amount of reaction added at each step (as defined in the **REACTION** data block) be written to the selected-output file (**-reaction** true). The **USER_PUNCH** data block prints additional columns of information to the selected-output file, including all of the information needed to make figure 10. Two additional simulations add successively larger amounts of strontium carbonate to the system up to a total addition of 10 moles.

Table 29.--Input data set for example 10

```
TITLE Example 10.--Solid solution of strontianite and aragonite.
PHASES
        Strontianite
                SrCO3 = CO3-2 + Sr+2
                log_k           -9.271
        Aragonite
                CaCO3 = CO3-2 + Ca+2
                log_k           -8.336
END
SOLID_SOLUTIONS 1
        Ca(x)Sr(1-x)CO3
                -comp1   Aragonite       0
                -comp2   Strontianite    0
                -Gugg_nondim   3.43    -1.82
END
SOLUTION 1
        -units mmol/kgw
        pH 5.93 charge
        Ca      3.932
        C       7.864
EQUILIBRIUM_PHASES 1
        CO2(g) -0.01265 10
        Aragonite
SAVE solution 1
END
#  Total of 0.00001 to 0.005 moles of SrCO3 added
USE solution 1
USE solid_solution 1
REACTION 1
        SrCO3   1.0
        .005 in 500 steps
PRINT
        -reset false
        -user_print true
```

```
USER_PRINT
-start
  10 sum = (S_S("Strontianite") + S_S("Aragonite"))
  20 if sum = 0 THEN GOTO 110
  30 xb = S_S("Strontianite")/sum
  40 xc = S_S("Aragonite")/sum
  50 PRINT "Simulation number:     ", SIM_NO
  60 PRINT "Reaction step number: ", STEP_NO
  70 PRINT "SrCO3 added:           ", RXN
  80 PRINT "Log Sigma pi:        ", LOG10 (ACT("CO3-2") * (ACT("Ca+2") + ACT("Sr+2")))
  90 PRINT "XAragonite:            ", xc
 100 PRINT "XStrontianite:         ", xb
 110 PRINT "XCa:                   ", TOT("Ca")/(TOT("Ca") + TOT("Sr"))
 120 PRINT "XSr:                   ", TOT("Sr")/(TOT("Ca") + TOT("Sr"))
 130 PRINT "Misc 1:                ", MISC1("Ca(x)Sr(1-x)CO3")
 140 PRINT "Misc 2:                ", MISC2("Ca(x)Sr(1-x)CO3")
-end
SELECTED_OUTPUT
        -file ex10.sel
        -reset false
        -reaction true
USER_PUNCH
-head   lg_SigmaPi X_Arag X_Stront X_Ca_aq X_Sr_aq mol_Misc1 mol_Misc2 mol_Arag
mol_Stront
-start
  10 sum = (S_S("Strontianite") + S_S("Aragonite"))
  20 if sum = 0 THEN GOTO 60
  30 xb = S_S("Strontianite")/(S_S("Strontianite") + S_S("Aragonite"))
  40 xc = S_S("Aragonite")/(S_S("Strontianite") + S_S("Aragonite"))
  50 REM Sigma Pi
  60 PUNCH LOG10(ACT("CO3-2") * (ACT("Ca+2") + ACT("Sr+2")))
  70 PUNCH xc                                  # Mole fraction aragonite
  80 PUNCH xb                                  # Mole fraction strontianite
  90 PUNCH TOT("Ca")/(TOT("Ca") + TOT("Sr"))   # Mole aqueous calcium
 100 PUNCH TOT("Sr")/(TOT("Ca") + TOT("Sr")) # Mole aqueous strontium
 110 x1 = MISC1("Ca(x)Sr(1-x)CO3")
 120 x2 = MISC2("Ca(x)Sr(1-x)CO3")
 130 if (xb < x1 OR xb > x2) THEN GOTO 250
 140    nc = S_S("Aragonite")
 150    nb = S_S("Strontianite")
 160    mol2 = ((x1 - 1)/x1)*nb + nc
 170    mol2 = mol2 / ( ((x1 -1)/x1)*x2 + (1 - x2))
 180    mol1 = (nb - mol2*x2)/x1
 190    REM                                   # Moles of misc. end members if in gap
 200    PUNCH mol1
 210    PUNCH mol2
 220    GOTO 300
 250    REM                                 # Moles of misc. end members if not in gap
 260    PUNCH 1e-10
 270    PUNCH 1e-10
 300 PUNCH S_S("Aragonite")                   # Moles aragonite
 310 PUNCH S_S("Strontianite")                # Moles Strontianite
-end
END
```

```
#
#  Total of 0.001 to 0.1 moles of SrCO3 added
#
USE solution 1
USE solid_solution 1
REACTION 1
        SrCO3    1.0
        .1 in 100 steps
END
#
#  Total of 0.1 to 10 moles of SrCO3 added
#
USE solution 1
USE solid_solution 1
REACTION 1
        SrCO3    1.0
        10.0 in 100 steps
END
```

The excess free-energy parameters describe a nonideal solid solution that has a miscibility gap. For compositions that fall within the miscibility gap, the activities of calcium and strontium within the aqueous phase remain fixed and are in equilibrium with solids of two compositions, one solid with a strontium mole fraction of 0.0048 and one solid with a strontium mole fraction of 0.8579. For the simulations in the example, each incremental addition of strontium carbonate increases the mole fraction of strontium carbonate in the solid until about 0.001 mol of strontium carbonate have been added (fig. 10A). That point is the beginning of the miscibility gap (fig. 10) and the composition of the solid is 0.0048 strontium mole fraction. The next increments of strontium carbonate (up to 0.005 mol strontium carbonate added) produce constant mole fractions of calcium and strontium in the solution (fig. 10B) and equilibrium with both the miscibility-gap end members. However, the amounts of calcium carbonate and strontium carbonate in the solid phases (fig. 10C) and the amounts of each of the miscibility gap end members (fig. 10D) vary with the amount of strontium carbonate added. Finally, the end of the miscibility gap is reached after about 0.005 mol of strontium carbonate have been added. At this point, the solution is in equilibrium with a single solid with a strontium mole fraction of 0.8579. Addition of more strontium carbonate increases the mole fractions of strontium in the aqueous phase and in the solid solution until both mole fractions are nearly 1.0 after the addition of 10 mol of strontium carbonate.

Example 11.--Transport and Cation Exchange

The following example of advective transport in the presence of a cation exchanger is derived from a sample calculation for the program PHREEQM (Appelo and Postma, 1993, example 10.13, p. 431-434). The chemical composition of the effluent from a column containing a cation exchanger is simulated. Initially, the column contains a sodium-potassium-nitrate solution in equilibrium with the cation exchanger. The column is then flushed with three pore volumes of calcium chloride solution. Calcium, potassium, and sodium react to equilibrium with the exchanger at all times. The problem is run two ways--by using the **ADVECTION** data block, which models only advection, and by using the **TRANSPORT** data block, which simulates advection and dispersive mixing.

The input data set is listed in table 30. The column has 40 cells to be consistent with one of the runs described by Appelo and Postma (1993). This requires that 40 solutions, numbered 1 through 40, be defined; the number of the solution corresponds to the number of the cell in a column. In this example, all cells contain the same solution,

but this is not necessary. Solutions could be defined differently for each cell and could be defined by reactions in the current or preceding simulations (using the **SAVE** keyword). The definition of a solution for each cell is mandatory, but the definition of an exchanger for each cell is optional. The number of the exchanger corresponds to the number of the cell in a column, and if an exchanger is defined for a cell number, it is used in the calculations for that cell. In this example, an identical exchanger composition is prescribed for all cells.

The solution filling each of the 40 cells of the column is defined with the **SOLUTION** 1-40 data block. The infilling solution for the column must be defined as **SOLUTION** 0, and it is a calcium chloride solution. The amount and composition of the exchanger in each of the 40 cells is defined by the **EXCHANGE** 1-40 data block. The number of exchange sites in each cell is 1.1 mmol, and the initial composition of the exchanger is calculated such that it is in equilibrium with solution 1. Note that the initial exchange composition is calculated assuming that the composition of solution 1 is fixed; the composition of solution 1 is not changed during the initial exchange-composition calculation.

The **ADVECTION** data block need only include the number of cells and the number of shifts for the simulation. The calculation only accounts for numbers of pore volumes that flow through the cells, no explicit definition of time or distance is used. The identifiers **-punch_cells** and **-punch_frequency** specify that data will be written to the selected-output file for cell 40 at each shift. The identifiers **-print_cells** and **-print_frequency** indicate that data will be written to the output file for cell 40 every 20 shifts.

The **SELECTED_OUTPUT** data block specifies that the shift (or advection step number) and the total dissolved concentrations of sodium, chloride, potassium, and calcium will be written to the file *ex11adv.sel*. Pore volumes can be calculated from the shift number; one shift moves a solution to the next cell, and the last solution out of the column. PHREEQC calculates cell-centered concentrations, so that the concentrations in the last cell arrive a *half* shift later at the column end. In this example, one shift represents 1/40 of the column pore volume. The number of pore volumes (*PV*) that have been flushed from the column is therefore *PV* = (number of shifts + 0.5) / 40. The number of pore volumes is calculated and printed to the selected-output file using the **USER_PUNCH** data block.

Following the advection calculation (**ADVECTION**), the initial conditions are reset for the advection and dispersion calculation (**TRANSPORT**) with a second set of **SOLUTION** and **EXCHANGE** data blocks. **SOLUTION** 0 is unchanged by the **ADVECTION** simulation and need not be redefined. The **TRANSPORT** data block includes a much more explicit description of the transport process than the **ADVECTION** data block. The length of each cell (**-length**), the boundary conditions at the column ends (**-boundary_cond**), the direction of flow (**-flow_direction**), the dispersivity (**-dispersivity**), and the diffusion coefficient (**-diffc**) can all be specified. The identifier **-correct_disp** should be set to true when modeling outflow from a column with flux boundary conditions. The identifiers **-punch**, **-punch_frequency**, **-print**, and **-print_frequency** serve the same function as in the **ADVECTION** data block. The second **SELECTED_OUTPUT** data block specifies that the transport step (shift) number and total dissolved concentrations of sodium, chloride, potassium, and calcium will be written to the file *ex11trn.sel*. The **USER_PUNCH** data block from the advection simulation is still in effect and the pore volume at each transport step is calculated and written to the selected-output file.

Table 30.--Input data set for example 11

```
TITLE Example 11.--Transport and ion exchange.
SOLUTION 0  CaCl2
        units           mmol/kgw
```

```
        temp            25.0
        pH              7.0     charge
        pe              12.5    O2(g)   -0.68
        Ca              0.6
        Cl              1.2
SOLUTION 1-40  Initial solution for column
        units           mmol/kgw
        temp            25.0
        pH              7.0     charge
        pe              12.5    O2(g)   -0.68
        Na              1.0
        K               0.2
        N(5)            1.2
EXCHANGE 1-40
        equilibrate 1
        X               0.0011
ADVECTION
        -cells          40
        -shifts         120
        -punch_cells    40
        -punch_frequency 1
        -print_cells    40
        -print_frequency 20
SELECTED_OUTPUT
        -file           ex11adv.sel
        -reset          false
        -step
        -totals         Na Cl K Ca
USER_PUNCH
  -heading  Pore_vol
  10 PUNCH (STEP_NO + .5) / 40.
END
SOLUTION 1-40  Initial solution for column
        units           mmol/kgw
        temp            25.0
        pH              7.0     charge
        pe              12.5    O2(g)   -0.68
        Na              1.0
        K               0.2
        N(5)            1.2
EXCHANGE 1-40
        equilibrate 1
        X               0.0011
TRANSPORT
        -cells          40
        -length         0.002
        -shifts         120
        -time_step      720.0
        -flow_direction forward
        -boundary_cond  flux    flux
        -diffc          0.0e-9
        -dispersivity   0.002
        -correct_disp   true
        -punch          40
```

```
        -punch_frequency 1
        -print           40
        -print_frequency 20
SELECTED_OUTPUT
        -file            ex11trn.sel
        -reset           false
        -step
        -totals          Na Cl K Ca
END
```

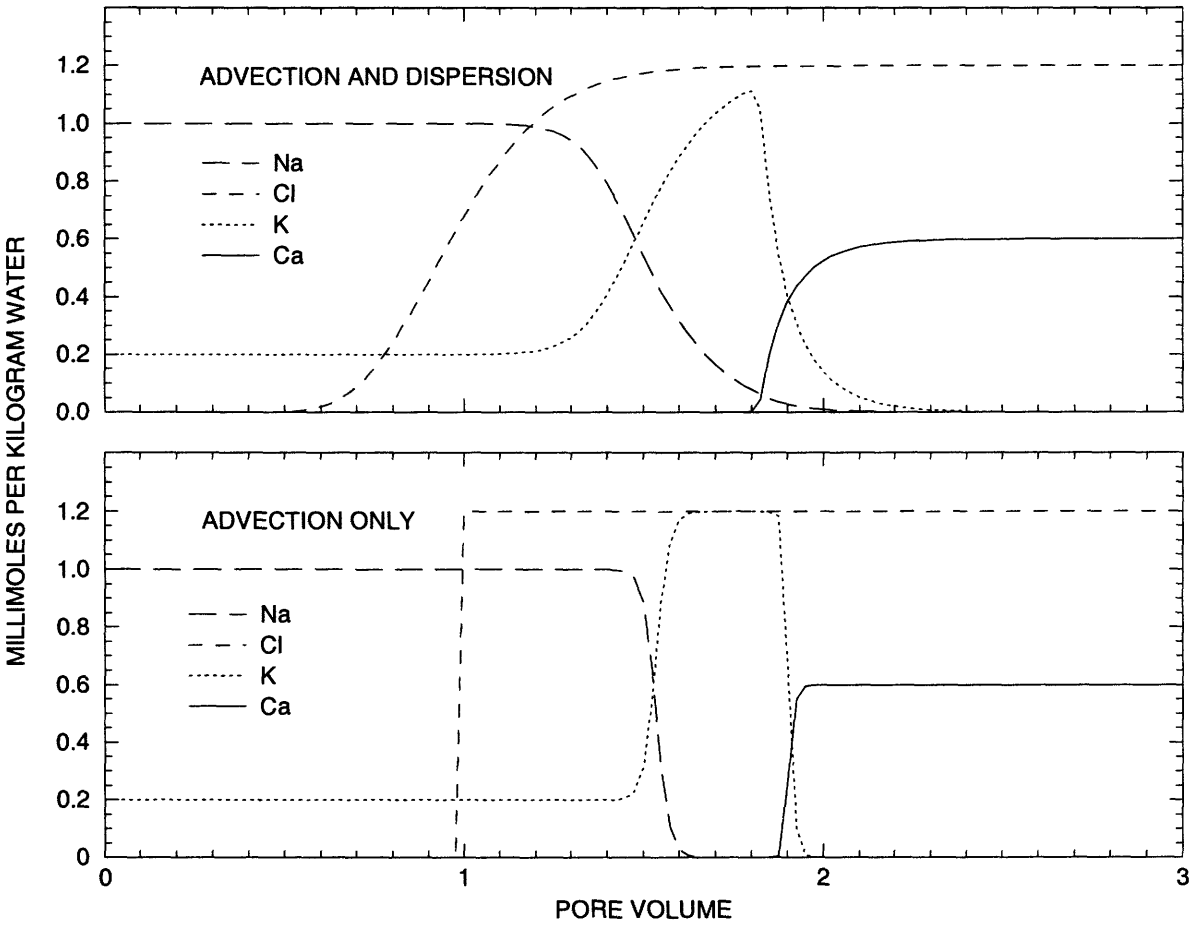

Figure 11.--Results of transport simulation of the replacement of sodium and potassium on a cation exchanger by infilling calcium chloride solution. Lines display concentrations at the outlet of the column as calculated with PHREEQC with advection only (**ADVECTION** keyword) and with advection and dispersion (**TRANSPORT** keyword).

The results for example 11 using **ADVECTION** and **TRANSPORT** keywords are shown by the curves in figure 11. The concentrations in cell 40, the end cell, are plotted against pore volumes. The main features of the calculations are the same between the two transport simulations. Chloride is a conservative solute and arrives in the effluent at about one pore volume. The sodium initially present in the column exchanges with the incoming calcium and is eluted as long as the exchanger contains sodium. The midpoint of the breakthrough curve for sodium occurs at about 1.5 pore volumes. Because potassium exchanges more strongly than sodium (larger log K in the exchange

reaction), potassium is released after sodium. Finally, when all of the potassium has been released, the concentration of calcium increases to a steady-state value equal to the concentration in the infilling solution.

The concentration changes of sodium and potassium in the effluent form a chromatographic pattern, which often can be calculated by simple means (Appelo, 1994b). The number of pore volumes needed for the arrival of the sodium-decrease front can be calculated with the formula $P_v = 1 + V^s$, where $V^s = \Delta q/\Delta c$, Δq indicates the change in sorbed concentration (mol/kgw), and Δc the change in solute concentration over the front. The sodium concentration in the solution that initially fills the column is 1.0 mmol/kgw and the initial sorbed concentration of sodium is 0.55; the concentration of sodium in the infilling solution is zero, which must eventually result in 0 sorbed sodium. Thus, $(V^s)_1 = \Delta q/\Delta c = (0.55 - 0)/(1 - 0) = 0.55$ and $P_v = 1.55$, which indicates that the midpoint of the sodium front should arrive at the end of the column after 1.55 pore volumes.

Next, potassium is displaced from the exchanger. The concentration in solution increases to 1.2 mmol/kgw to balance the Cl^- concentration, and then falls to 0 when the exchanger is exhausted. When potassium is the only cation in solution, it will also be the only cation on the exchanger. For potassium, $(V^s)_2 = \Delta q/\Delta c = (1.1 - 0)/(1.2 - 0) = 0.917$ and $P_v = 1.917$ pore volumes. It can be seen that the front locations for $(V^s)_1$ and $(V^s)_2$ are closely matched by the midpoints of the concentration changes shown in figure 11.

The differences between the two simulations are due entirely to the inclusion of dispersion in the **TRANSPORT** calculation. The breakthrough curve for chloride in the **TRANSPORT** calculation coincides with an analytical solution to the advection dispersion equation for a conservative solute (Appelo and Postma, 1993, p. 433). Without dispersion, the **ADVECTION** calculation produces a square-wave breakthrough curve for chloride. The characteristic smearing effects of dispersion are absent in the fronts calculated for the other elements as well, although some curvature exists due to the effects of the exchange reactions. The peak potassium concentration is larger in the **ADVECTION** calculation because the effects of dispersion are neglected.

Example 12.--Advective and Diffusive Flux of Heat and Solutes

The following example demonstrates the capability of PHREEQC to calculate transient transport of heat and solutes in a column or along a 1D flowline. A column is initially filled with a dilute KCl solution at 25°C in equilibrium with a cation exchanger. A KNO_3 solution then advects into the column and establishes a new temperature of 0°C. Subsequently, a sodium chloride solution at 24°C is allowed to diffuse from both ends of the column, assuming no heat is lost through the column walls. At one end a constant boundary condition is imposed and at the other end the final cell is filled with the sodium chloride solution and a closed boundary condition is prescribed. For the column end with a constant boundary condition, an analytical solution is compared with PHREEQC results, both for retardation $R = 1.0$ (Cl^-) and $R = 3.0$ (Na^+ and temperature). Finally, the second-order accuracy of the numerical method is verified by increasing the number of cells by a factor of three and demonstrating a decrease in the error of the numerical solution by approximately an order of magnitude relative to the analytical solution.

Table 31.--Input data set for example 12

```
TITLE Example 12.--Advective and diffusive transport of heat and solutes.
      Constant boundary condition at one end, closed at other.
      The problem is designed so that temperature should equal Na-conc
      (in mmol/kgw) after diffusion.
EXCHANGE_SPECIES
```

```
  Na+ + X- = NaX
    log_k        0.0
    -gamma       4.0      0.075
  H+ + X- = HX
    log_k        -99.
    -gamma       9.0      0.0
  K+ + X- = KX
    log_k        0.0
    -gamma        3.5     0.015
SOLUTION 0   24.0 mM KNO3
    units mol/kgw
    temp  0                  # Incoming solution 0C
    pH    7.0
    pe    12.0   O2(g) -0.67
    K     24.e-3
    N(5) 24.e-3
SOLUTION 1-20   0.001 mM KCl
    units mol/kgw
    temp 25    # Column is at 25C
    pH   7.0
    pe  12.0   O2(g) -0.67
    K    1e-6
    Cl   1e-6
EXCHANGE 1-20
    KX    0.048
TRANSPORT                    # Make column temperature 0C, displace Cl
   -cells   20
   -shifts  19
   -flow_d  forward
   -bcon    flux  flux
   -length  1.0
   -disp    0.0             # No dispersion
   -diffc   0.0             # No diffusion
   -thermal_diffusion   1.0 # No retardation for heat
PRINT
   -reset   false
END
SOLUTION 0   Fixed temp 24C, and NaCl conc (first type boundary cond) at inlet
    units  mol/kgw
    temp 24
    pH  7.0
    pe  12.0   O2(g) -0.67
    Na  24.e-3
    Cl  24.e-3
SOLUTION 20  Same as soln 0 in cell 20 at closed column end (second type boundary cond)
    units  mol/kgw
    temp 24
    pH  7.0
    pe  12.0   O2(g) -0.67
    Na  24.e-3
    Cl  24.e-3
EXCHANGE 20
    NaX    0.048
TRANSPORT                    # Diffuse 24C, NaCl solution from column end
```

```
   -shifts 1
   -flow_d diffusion
   -bcon   constant  closed
   -thermal_diffusion  3.0  # heat is retarded equal to Na
   -diffc  0.3e-9           # m^2/s
   -timest 1.0e+10          # 317 years, 19 substeps will be used
SELECTED_OUTPUT
   -file    ex12.sel
   -high_precision  true
   -reset   false
   -dist    true
   -temp    true
USER_PUNCH
   -head Na_mmol K_mmol Cl_mmol
   10 PUNCH TOT("Na")*1000, TOT("K")*1000, TOT("Cl")*1000
END
```

The input data set for example 12 is shown in table 31. The **EXCHANGE_SPECIES** data block is used (1) to make the exchange constant for KX equal to NaX (log_k = 0.0), (2) to effectively remove the possibility of hydrogen ion exchange, and (3) to set activity coefficients for exchange species equal to their aqueous counterparts (**-gamma** identifier), so that the exchange between Na^+ and K^+ is linear and the retardation is constant. The infilling solution for transport, **SOLUTION** 0, is defined with temperature 0°C and 24 mmol/kgw KNO_3. To stabilize the pe, the concentration of oxygen is defined to be that which is in equilibrium with atmospheric oxygen partial pressure.The column is discretized in 20 cells, which are filled initially with a 1 μ mol/kgw KCl solution (**SOLUTION** 1-20). Each cell has 48 mmol of exchange sites, which are defined to contain only potassium by the data block **EXCHANGE** 1-20.

The **TRANSPORT** data block defines cell lengths of 1 m (**-length** 1), no dispersion (**-disp** 0), no diffusion (**-diffc** 0), and no retardation for temperature (**-thermal_diffusion** 1). **SOLUTION** 0 is shifted 19 times into the column (**-shifts** 19, **-flow_d** forward), and arrives in cell 19 at the last shift of the advective-dispersive transport simulation. The boundary conditions at the column ends are of flux type (**-bcon** flux flux). In this initial advective-dispersive transport simulation, no exchange occurs (the solution contains only K^+ as cation, and the exchange sites are completely filled with potassium), and the concentrations and temperatures in the first 19 cells are brought to 24 mmol/kgw KNO_3 and 0°C. This result could be more easily achieved with **SOLUTION** data blocks directly (as is done in Attachment C), but the simulation demonstrates how transient boundary and flow conditions can be represented. The dissolved and solid composition and temperature of each cell of the column is automatically saved after each shift in the advective-dispersive transport simulation. The keyword **PRINT** is used to exclude all printing to the output file (**-reset** false).

In the next advective-dispersive transport simulation, diffusion is calculated from the column ends, beginning with the column composition and temperatures that exist following the first advective-dispersive transport calculation, except that a NaCl solution is now defined as solution 0, which diffuses into the top of the column, and as solution 20, which diffuses from the bottom of the column. The new **SOLUTION** 0 is defined with a temperature of 24°C and with 24 mmol/kgw NaCl. The last cell (**SOLUTION** 20) is also defined to have this solution composition and temperature. The exchanger in cell 20 is defined to be in equilibrium with the new solution composition in cell 20 (**EXCHANGE** 20).

The **TRANSPORT** data block defines one diffusive transport period (**-shifts** 1; **-flow_d** diffusion). The boundary condition at the first cell is constant concentration, and at the last cell the column is closed (**-bcon** constant closed). The effective diffusion coefficient (**-diffc**) is set to 0.3e-9 m^2/s, and the time step (**-timest**) is defined to be 1.e10 s. Because only one diffusive time period is defined (**-shifts** 1), the total time modeled is equal to the time step, 1e10 s. However, the time step will automatically be divided into a number of time substeps to satisfy stability criteria for the numerical method. The heat retardation factor is set to 3.0 (**-thermal_diffusion** 3.0). For Na^+ the ratio of exchangeable concentration (maximum NaX is 48 mmol/kgw) over solute concentration (maximum Na^+ = 24 mmol/kgw) is 2.0 for all concentrations, and the retardation is therefore $R = 1 + dNaX/dNa^+ = 3.0$, which is numerically equal to the temperature retardation.

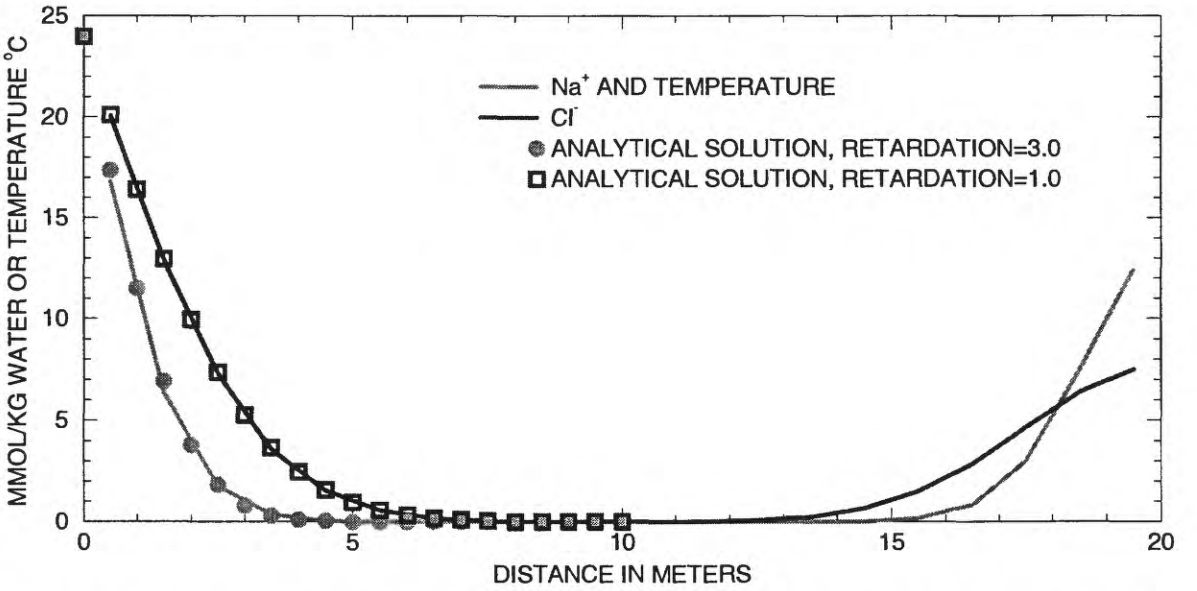

Figure 12.--Simulation results for diffusion from column ends of heat and Na^+ (retardation $R = 3$) and Cl^- ($R = 1$) compared with constant-boundary-condition analytical solution.

The **SELECTED_OUTPUT** data block specifies the name of the selected-output file to be *ex12.sel*. The identifier "**-high_precision** true" is used to obtain an increased number of digits in the printing and the identifiers **-dist** and **-temp** specify that the distance and temperature of each cell will be printed to the file. The **USER_PUNCH** data block is used to print concentrations of sodium, potassium, and chloride to the selected-output file in units of mmol/kgw.

In the model, the temperature is calculated with the (linear) retardation formula; however, the Na^+ concentration is calculated by the cation-exchange reactions. Even though the Na^+ concentration and the temperature are calculated by different methods, the numerical values should be the same because the initial conditions and the transient conditions are numerically equal and the retardation factors are the same. The temperature and the Na^+ concentration are equal to at least 6 digits in the PHREEQC selected-output file, which indicates that the algorithm for the chemical transport calculations is correct for the simplified chemistry considered in this example. A further check on the accuracy is obtained by comparing simulation results with an analytical solution. For an infinite column with $C_x = 0$ for $t = 0$, and diffusion from $x = 0$ with $C_{x=0} = C_0$ for $t > 0$ the analytical solution is

$$C_{x,t} = C_0 erfc\left(\frac{x}{2\sqrt{D_e t / R}}\right), \tag{160}$$

where D_e is the effective diffusion coefficient.

The PHREEQC results are compared with the analytical solution for Cl^- and for temperature and Na^+ in figure 12 and show excellent agreement. Notice that diffusion of Cl^- from the column ends has not yet "touched" in the mid-section, so that the column is still effectively infinite and the analytical solution is appropriate. Although both ends of the column started with the same temperature and concentration, $x = 0$ is maintained at the same temperature and concentrations because of the constant boundary condition. The temperature and concentrations have decreased in cell 20 (plotted at the midpoint of the cell, $x = 19.5$) because of the closed boundary condition that was applied at $x = 20$; no flux of heat or mass through this boundary is allowed and the temperature and concentrations are diminishing because of diffusion into the column. The sodium concentration is not dissipating as rapidly as the chloride concentration because exchange sites must be filled with sodium along the diffusive reach.

Because this example has an analytical solution, it is possible to verify the second-order accuracy of the numerical algorithms used in PHREEQC. For a second-order method, decreasing the cell size by a factor of three should improve the results by about a factor of nine. An input file is given in Attachment C that performs the 20-cell calculation given in this example together with a 60-cell calculation. The deviations from the analytical solution at the end of the time step are calculated at distances from 0.5 to 8.5 m in 0.5 m increments. The results are shown in table 32. As expected for a second-order method, the deviations from the analytical solution decreased by approximately an order of magnitude as the result of decreasing the cell size by a factor of three.

Table 32.--Numerical errors relative to an analytical solution for example 12 for a 20-cell and a 60-cell model

Distance	Error in Cl concentration		Error in Na concentration	
	20-cell model	60-cell model	20-cell model	60-Cell model
0.5	3.32E-05	3.03E-06	5.75E-04	4.42E-05
1.5	8.17E-05	7.66E-06	5.54E-04	6.08E-05
2.5	9.18E-05	9.09E-06	8.29E-05	1.43E-05
3.5	7.15E-05	7.65E-06	-5.07E-05	-5.64E-06
4.5	4.24E-05	4.98E-06	-2.54E-05	-3.26E-06
5.5	2.00E-05	2.61E-06	-5.44E-06	-6.27E-07
6.5	7.81E-06	1.12E-06	-7.20E-07	-6.15E-08
7.5	2.55E-06	3.97E-07	-6.77E-08	-3.48E-09
8.5	5.58E-07	7.65E-08	-4.90E-09	-1.21E-10

Example 13.--1D Transport in a Dual Porosity Column with Cation Exchange

This example demonstrates the capabilities of PHREEQC to calculate flow in a dual-porosity medium with exchange among the mobile and immobile pores via diffusion. The flexible input format and the modular definition of additional solutions and processes in PHREEQC allow inclusion of heterogeneities and various complexities within a 1D column. This example considers a column filled with small clay beads of 2 cm diameter, which act as stagnant zones. Both the first-order exchange approximation and finite differences are applied in this example, and transport of both a conservative and a retarded (by ion exchange) chemical are considered. It is furthermore shown how a heterogeneous column can be modeled by prescribing mixing factors to account for diffusion between mobile and immobile cells with keyword **MIX**.

Stagnant Zone Calculation Using the First-Order Exchange Approximation with Implicit Mixing Factors

The example input file (table 33) is for a column with a uniform distribution of the stagnant porosity along the column. The 20 mobile cells are numbered 1-20. Each mobile cell, *n*, exchanges with one immobile cell, which is numbered 20 + 1 + *n* (cells 22-41 are immobile cells). All cells are given an identical initial solution and exchange complex, but these could be defined differently for each individual cell. It is also possible to distribute the immobile cells over the column non-uniformly, simply by omitting solutions for the stagnant cells that are not present. The connections between the mobile-zone and the stagnant-zone cells and among stagnant zone cells can be varied along the column as well, but this requires that mixing factors among the mobile and immobile cells are prescribed using the keyword **MIX**.

As defined in table 33, the column initially contains a 1 mmol/L KNO_3 solution in both the mobile and the stagnant zone (**SOLUTION** 1-41). A NaCl-NO_3 solution flows into the column (**SOLUTION** 0). An exchange complex with 1 mmol of sites is defined for each cell (**EXCHANGE** 1-41), and exchange coefficients are adapted to give linear retardation $R = 2$ for Na^+ (**EXCHANGE_SPECIES**). The first **TRANSPORT** data block is used to define the physical and flow characteristics of the first transport simulation. The column is 2 m in length and is discretized in 20 cells (**-cells**) of 0.1 m (**-length**). A pulse of 5 shifts (**-shifts**) of the infilling solution (**SOLUTION** 0) is introduced into the column. The length of time for each shift is 3600 s (**-timest**), which results in a velocity in the mobile pores $v_m = 0.1 / 3600 = 2.78\text{e-}5$ m/s. The dispersivity is set to 0.015 m for all cells (**-disp**). The diffusion coefficient is set to 0.0 (**-diffc**).

The stagnant/mobile interchange is defined using the first-order exchange approximation. The stagnant zone consists of spheres with radius $r = 0.01$ m, diffusion coefficient $D_e = 3.\text{e-}10$ m^2/s, and shape factor $f_{s\rightarrow 1} = 0.21$ according to table 1 (see "Transport in Dual Porosity Media"). These variables give an exchange factor $\alpha = 6.8\text{e-}6$ s^{-1}. Mobile porosity is $\theta_m = 0.3$ (**-stag**) and immobile porosity $\theta_{im} = 0.1$. For the first-order exchange approximation in PHREEQC, a single cell immobile zone and the parameters α, θ_m and θ_{im} are specified with **-stag**. This stagnant zone definition causes each cell in the mobile zone (numbered 1-20) to have an associated cell in the immobile zone (numbered 22-41). The **PRINT** data block is used to eliminate all printing to the output file.

Following the pulse of NaCl solution, 10 shifts of 1 mmol KNO_3/L (second **SOLUTION** 0) are introduced into the column. The second **TRANSPORT** data block does not redefine any of the column or flow characteristics, but specifies that results for cells 1 through 20 (**-punch_cells**) be written to the selected output file after 10 shifts (**-punch_frequency**). The data blocks **SELECTED_OUTPUT** and **USER_PUNCH** specify the data to be written to the selected-output file.

Table 33.--Input data set for example 13A: Stagnant zone with implicitly defined mixing factors

```
TITLE Example 13A.--1 mmol/l NaCl/NO3 enters column with stagnant zones.
                    Implicit definition of first-order exchange model.
SOLUTION 0     # 1 mmol/l NaCl
        units   mmol/l
        pH       7.0
        pe      13.0     O2(g)    -0.7
        Na       1.0     # Na has Retardation = 2
        Cl       1.0     # Cl has Retardation = 1, stagnant exchange
        N(5)     1.0     # NO3 is conservative
#       charge imbalance is no problem ...
END
SOLUTION 1-41  # Column with KNO3
        units   mmol/l
        pH       7.0
        pe      13.0    O2(g)     -0.7
        K        1.0
        N(5)     1.0
EXCHANGE 1-41
        -equil  1
        X       1.e-3
EXCHANGE_SPECIES # For linear exchange, make KX exch. coeff. equal to NaX
        K+ + X- = KX
        log_k   0.0
        -gamma  3.5     0.015
END
TRANSPORT
        -cells  20
        -shifts 5
        -flow_d   forward
        -timest 3600
        -bcon   flux  flux
        -diffc  0.0
        -length 0.1
        -disp   0.015
        -stag       1  6.8e-6  0.3        0.1
#   1 stagnant layer^, ^alpha, ^theta(m), ^theta(im)
PRINT
        -reset false
END
SOLUTION 0  # Original solution reenters
        units   mmol/l
        pH       7.0
        pe      13.0    O2(g)     -0.7
        K        1.0
        N(5)     1.0
TRANSPORT
        -shifts 10
        -punch_frequency    10
        -punch_cells        1-20
SELECTED_OUTPUT
        -file   ex13a.sel
        -reset  false
```

```
        -solution
        -distance       true
USER_PUNCH
        -head Cl_mmol Na_mmol
10 PUNCH TOT("Cl")*1000, TOT("Na")*1000
END
```

The mixing factors $mixf_m$ and $mixf_{im}$ for the first-order exchange approximation for this example are derived from equations 121 and 123 as follows:

$$mixf_{im} = \frac{\theta_m}{\theta_m + \theta_{im}} \times \left(1 - \exp\left(-\frac{\alpha\, t(\theta_m + \theta_{im})}{\theta_m \theta_{im}}\right)\right) \tag{161}$$

and

$$mixf_m = mixf_{im} \frac{\theta_{im}}{\theta_m}. \tag{162}$$

The retardation factors R_m and R_{im} are not included here in the formulas for $mixf_{im}$ and $mixf_m$ because in PHREEQC the retardation is a consequence of chemical reactions. According to equations 161 and 162, for this example the mixing factors are calculated to be $mixf_{im} = 0.20886$ and $mixf_m = 0.06962$. The mixing factors differ for the mobile cell and the immobile cell to account for the difference in the volume of mobile and immobile water.

In PHREEQC, a mixing of mobile and stagnant water is done after each diffusion/dispersion step. This means that the time step decreases when the cells are made smaller and when more diffusive steps ("mixruns") are performed. A 20-cell model as in example 13A has one mixrun. A 100 cell model would have 3 mixruns (equation 110 requires n=3 for $mixf < 1/3$), and the time step for calculating $mixf_{im}$ would be (3600/5) / 3 = 240 s. A time step t = 240 s leads to $mixf_{im} = 0.01614$ in the 100 cell model.

Stagnant Zone Calculation Using the First-Order Exchange Approximation with Explicit Mixing Factors

The input file with explicit mixing factors for a uniform distribution of the stagnant zones is given in table 34. The **SOLUTION** data blocks are identical to the previous input file. One stagnant layer without further information is defined (**-stag** 1, in the **TRANSPORT** data block). The mobile/immobile exchange is set by the mix fraction given in the **MIX** data blocks. The results of this input file are identical with the results from the previous input file in which the shortcut notation was used. However, the explicit definition of mix factors illustrates that a non-uniform distribution of the stagnant zones, or other physical properties of the stagnant zone, can be included in PHREEQC simulations by varying the mixing fractions which define the exchange among mobile and immobile cells.

Table 34.--Input data set for example 13B: Stagnant zone with explicitly defined mixing factors

```
TITLE Example 13B.--1 mmol/l NaCl/NO3 enters column with stagnant zones.
                    Explicit definition of first-order exchange factors.
SOLUTION 0     # 1 mmol/l NaCl
        units   mmol/l
        pH       7.0
        pe      13.0     O2(g)    -0.7
        Na       1.0     # Na has Retardation = 2
        Cl       1.0     # Cl has Retardation = 1, stagnant exchange
        N(5)     1.0     # NO3 is conservative
#       charge imbalance is no problem ...
```

```
END
SOLUTION 1-41  # Column with KNO3
        units   mmol/l
        pH       7.0
        pe      13.0   O2(g)     -0.7
        K        1.0
        N(5)     1.0
EXCHANGE 1-41
        -equil  1
        X       1.e-3
EXCHANGE_SPECIES # For linear exchange, make KX exch. coeff. equal to NaX
        K+ + X- = KX
        log_k   0.0
        -gamma  3.5     0.015
END
MIX  1;  1 .93038;      22 .06962       ;MIX  2;         2 .93038;      23 .06962;
MIX  3;  3 .93038;      24 .06962       ;MIX  4;         4 .93038;      25 .06962;
MIX  5;  5 .93038;      26 .06962       ;MIX  6;         6 .93038;      27 .06962;
MIX  7;  7 .93038;      28 .06962       ;MIX  8;         8 .93038;      29 .06962;
MIX  9;  9 .93038;      30 .06962       ;MIX 10;        10 .93038;      31 .06962;
MIX 11; 11 .93038;      32 .06962       ;MIX 12;        12 .93038;      33 .06962;
MIX 13; 13 .93038;      34 .06962       ;MIX 14;        14 .93038;      35 .06962;
MIX 15; 15 .93038;      36 .06962       ;MIX 16;        16 .93038;      37 .06962;
MIX 17; 17 .93038;      38 .06962       ;MIX 18;        18 .93038;      39 .06962;
MIX 19; 19 .93038;      40 .06962       ;MIX 20;        20 .93038;      41 .06962;
#
MIX 22;  1 .20886;      22 .79114       ;MIX 23;         2 .20886;      23 .79114;
MIX 24;  3 .20886;      24 .79114       ;MIX 25;         4 .20886;      25 .79114;
MIX 26;  5 .20886;      26 .79114       ;MIX 27;         6 .20886;      27 .79114;
MIX 28;  7 .20886;      28 .79114       ;MIX 29;         8 .20886;      29 .79114;
MIX 30;  9 .20886;      30 .79114       ;MIX 31;        10 .20886;      31 .79114;
MIX 32; 11 .20886;      32 .79114       ;MIX 33;        12 .20886;      33 .79114;
MIX 34; 13 .20886;      34 .79114       ;MIX 35;        14 .20886;      35 .79114;
MIX 36; 15 .20886;      36 .79114       ;MIX 37;        16 .20886;      37 .79114;
MIX 38; 17 .20886;      38 .79114       ;MIX 39;        18 .20886;      39 .79114;
MIX 40; 19 .20886;      40 .79114       ;MIX 41;        20 .20886;      41 .79114;
TRANSPORT
        -cells  20
        -shifts  5
        -flow_d   forward
        -timest 3600
        -bcon   flux  flux
        -diffc  0.0
        -length 0.1
        -disp   0.015
        -stag   1
PRINT
        -reset false
END
SOLUTION 0  # Original solution reenters
        units   mmol/l
        pH       7.0
        pe      13.0   O2(g)     -0.7
        K        1.0
```

```
        N(5)      1.0
TRANSPORT
        -shifts  10
        -punch_frequency     10
        -punch_cells  1-20
SELECTED_OUTPUT
        -file    ex13b.sel
        -reset  false
        -distance        true
        -solution
USER_PUNCH
        -head Cl_mmol Na_mmol
10 PUNCH TOT("Cl")*1000, TOT("Na")*1000
END
```

Stagnant Zone Calculation Using a Finite Difference Approximation

The stagnant zone consists of spheres with radius $r = 0.01$ m. Diffusion into the spheres induces radially symmetric concentration changes according to the differential equation:

$$\frac{\partial C}{\partial t} = D_e\left(\frac{\partial^2 C}{\partial r} + \frac{2}{r}\frac{\partial C}{\partial r}\right). \tag{163}$$

The calculation in finite differences can therefore be simplified to one (radial) dimension..

Table 35.--Mixing factors for finite difference calculation of diffusion in spheres

Cell	r, m	V_j, m^3	A_{ij}, m^2	h_{ij}, m	f_{bc}	$mixf_{ij}$	$mixf_{jj}$	$mixf_{jk}$
102	0.001	3.35e-8	5.03e-5	0.002	1	0.81	0.19	-
82	.003	2.35e-7	2.01e-4	.002	1	.463	.421	0.116
62	.005	6.37e-7	4.52e-4	.002	1	.384	.446	.170
42	.007	1.24e-6	8.04e-4	.002	1	.350	.453	.197
22	.009	2.04e-6	1.26e-3	.002	1.72	.571	.217	.212
1	-	1.26e-5	-	-	-	-	.907	.093

The calculation follows the theory outlined in the section "Transport in Dual Porosity Media" of this manual. The stagnant zone is divided into a number of layers that mix by diffusion. In this example, the sphere is cut in 5 equidistant layers with $\Delta r = 0.002$ m. Five stagnant layers are defined under keyword **TRANSPORT** with **-stagnant** 5 (table 36). Mixing is specified among adjacent cells in the stagnant layers with **MIX** data blocks; the mixing factors are calculated by equations 128 and 129. For the calculation, the volume V_j of cell j (m^3) is needed, the shared surface area A_{ij} of cell i and j (m^2), the distance between midpoints h_{ij} of cells i and j (m), and the correction factor for boundary cells f_{bc} (dimensionless). The values for mobile cell 1 and associated immobile cells are given in table 35. The cells in the immobile layer are numbered as $n + l \times cells + 1$, where n is the number of a mobile cell, l is the number of the stagnant layer, and *cells* is the total number of mobile cells. In this example, the boundary cell in the stagnant zone for cell number 1 is cell number 22 and the other four stagnant layers are cell numbers 42, 62, 82, and 102, with number 102 being the innermost cell of the sphere, which is connected only to

one other cell (cell 82). The volume of the mobile cell (cell 1) is expressed relative to the volume of a sphere of radius 0.01 m, by multiplying this volume with θ_m/θ_{im} (4.19e-6 0.3 / 0.1 = 1.26e-5). In table 35 the value for f_{bc} is 1.72 as calculated from equation 127. It may be noted that using f_{bc} = 1.81 slightly improves the fit to an analytical solution given in Crank (1975) for diffusion into spheres in a closed vessel (a beaker with solution and clay beads). However, changing f_{bc} to 1.81 has little effect on the concentration profiles for the column which are shown in figure 13

Table 36.--Input data set for example 13C: Stagnant zone with diffusion calculated by finite differences (partial listing)

```
TITLE Example 13C.--1 mmol/l NaCl/NO3 enters column with stagnant zones.
                    5 layer stagnant zone with finite differences.
SOLUTION 0     # 1 mmol/l NaCl
        units   mmol/l
        pH       7.0
        pe      13.0     O2(g)    -0.7
        Na       1.0     # Na has Retardation = 2
        Cl       1.0     # Cl has Retardation = 1, stagnant exchange
        N(5)     1.0     # NO3 is conservative
#       charge imbalance is no problem ...
END
SOLUTION 1-121  # Column with KNO3
        units           mmol/l
        pH       7.0
        pe      13.0    O2(g)            -0.7
        K        1.0
        N(5)     1.0
EXCHANGE 1-121
        -equil  1
        X       1.e-3
EXCHANGE_SPECIES # For linear exchange, make KX exch. coeff. equal to NaX
        K+ + X- = KX
        log_k           0.0
        -gamma  3.5     0.015
END
MIX    1;     1  0.90712;   22  0.09288
MIX   22;     1  0.57098;   22  0.21656;   42  0.21246
MIX   42;    22  0.35027;   42  0.45270;   62  0.19703
MIX   62;    42  0.38368;   62  0.44579;   82  0.17053
MIX   82;    62  0.46286;   82  0.42143;  102  0.11571
MIX  102;    82  0.81000;  102  0.19000
#
#   MIX definitions omitted for mobile cells
#   2-19 and associated immobile cells
#
MIX   20;    20  0.90712;   41  0.09288
MIX   41;    20  0.57098;   41  0.21656;   61  0.21246
MIX   61;    41  0.35027;   61  0.45270;   81  0.19703
MIX   81;    61  0.38368;   81  0.44579;  101  0.17053
MIX  101;    81  0.46286;  101  0.42143;  121  0.11571
MIX  121;   101  0.81000;  121  0.19000
TRANSPORT
```

```
        -cells  20
        -shifts 5
        -flow_direction forward
        -timest 3600
        -tempr  3.0
        -bcond  flux flux
        -diffc  0.0
        -length 0.10
        -disp   0.015
        -stag   5
PRINT
        -reset false
END
SOLUTION 0  # Original solution reenters
        units   mmol/l
        pH       7.0
        pe      13.0   O2(g)    -0.7
        K        1.0
        N(5)     1.0
TRANSPORT
        -shifts  10
        -punch_frequency 10
        -punch_cells  1-20
SELECTED_OUTPUT
        -file      ex13c.sel
        -reset     false
        -solution
        -distance  true
USER_PUNCH
        -head Cl_mmol Na_mmol
10 PUNCH TOT("Cl")*1000, TOT("Na")*1000
END
```

Note in table 36 that 121 solutions are defined, 1-20 for the mobile cells, and the rest for the immobile cells. The input file is identical with the previous one, except for **-stag** 5 and the mixing factors among the cells. Not all the mixing factors are shown in table 36, they are identical for subsequent cells and their neighboring stagnant cells. In this example with clay beads, only radial (1D) diffusion is considered, and only mixing among cells in different layers is defined. However, it is possible to include mixing among the immobile cells of adjacent mobile cells.

Figure 13 compares the concentration profiles in the mobile cells obtained with example 13A (and B) with example 13C. The basic features of the two simulations are the same. The positions of the peaks, as calculated by the two simulations, are similar. The Cl^- peak is near 1.2 m, but would be at about 1.45 m in the absence of stagnant zones. The integrated concentrations in the mobile porosity are about equal for the first-order exchange and the finite difference simulations. The exchange factor $f_{s\rightarrow l} = 0.21$ for the first-order exchange approximation appears to provide adequate accuracy for this simulation. However, the first-order exchange approximation produces lower peaks and more tailing than the more exact solution obtained with finite differences. Discrepancies can also appear as deviations in the breakthrough curve (Van Genuchten, 1985). The first-order exchange model is probably least accurate when applied to simulating the transport behavior of spheres; other shapes of the stagnant area can give a better correspondence. It is clear that the linear exchange model is much easier to apply because any explicit model requires the preparation of extended lists of mixing factors (notice that a separate simulation with **USER_PUNCH**

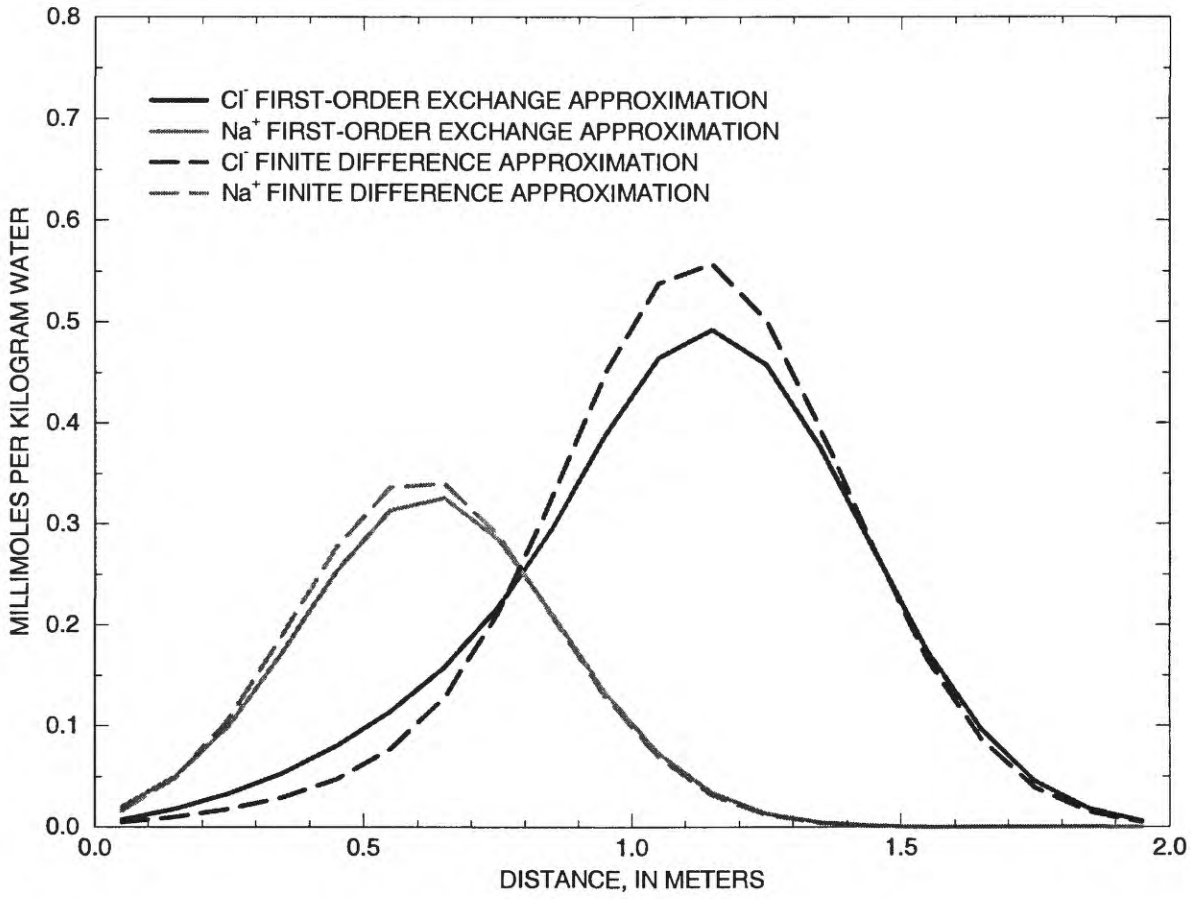

Figure 13.--Results of simulations of transport with diffusion into spherical stagnant zones modeled using finite difference and first-order exchange approximations.

can serve that purpose), which change when the discretization is adapted. The calculation time for a finite difference model with multiple immobile-zone layers may also be considerably longer than for the single immobile-zone layer of the first-order exchange approximation.

Example 14.--Advective Transport, Cation Exchange, Surface Complexation, and Mineral Equilibria

This example uses the phase-equilibrium, cation-exchange, and surface-complexation reaction capabilities of PHREEQC in combination with advective transport capabilities to model the evolution of water in the Central Oklahoma aquifer. The geochemistry of the aquifer has been described by Parkhurst and others (1996). Two predominant water types occur in the aquifer, a calcium magnesium bicarbonate water with pH in the range of 7.0 to 7.5 in the unconfined part of the aquifer and a sodium bicarbonate water with pH in the range of 8.5 to 9.2 in the confined part of the aquifer. In addition, marine-derived sodium chloride brines exist below the aquifer and presumably in fluid inclusions and dead-end pore spaces within the aquifer. Large concentrations of arsenic, selenium, chromium, and uranium occur naturally within the aquifer. Arsenic is associated almost exclusively with the high-pH, sodium bicarbonate water type.

The conceptual model for the calculation of this example assumes that brines initially filled the aquifer. The aquifer contains calcite, dolomite, clays with cation exchange capacity, and hydrous-ferric-oxide surfaces; initially, the cation exchanger and surfaces are in equilibrium with the brine. The aquifer is assumed to be recharged with rainwater that is concentrated by evaporation and equilibrates with calcite and dolomite in the vadose zone. This water then enters the saturated zone and reacts with calcite and dolomite in the presence of the cation exchanger and hydrous-ferric-oxide surfaces.

The calculations use the advective transport capabilities of PHREEQC with just a single cell representing the saturated zone. A total of 200 pore volumes of recharge water are advected into the cell and, with each pore volume, the water is equilibrated with the minerals, cation exchanger, and the surfaces in the cell. The evolution of water chemistry in the cell represents the evolution of the water chemistry at a point near the top of the saturated zone of the aquifer.

Initial Conditions

Parkhurst and others (1996) provide data from which it is possible to estimate the moles of calcite, dolomite, and cation-exchange sites in the aquifer per liter of water. The weight percent ranges from 0 to 2 percent for calcite and 0 to 7 percent for dolomite, with dolomite much more abundant. Porosity is stated to be 0.22. Cation-exchange capacity for the clay ranges from 20 to 50 meq/100 g, with average clay content of 30 percent. For these example calculations, calcite was assumed to be present at 0.1 weight percent and dolomite at 3 weight percent, which, by assuming a rock density of 2.7 kg/L, corresponds to 0.1 mol/L for calcite and 1.6 mol/L for dolomite. The number of cation-exchange sites was estimated to be 1.0 eq/L.

The amount of arsenic on the surface was estimated from sequential extraction data on core samples (Mosier and others, 1991). Arsenic concentrations in the solid phases generally ranged from 10 to 20 ppm., which corresponds to 1.3 to 2.6 mmol/L arsenic. The number of surface sites were estimated from the amount of extractable iron in sediments, which ranged from 1.6 to 4.4 percent (Mosier and others, 1991). A content of 2 percent iron for the sediments corresponds to 3.4 mol/L of iron. However, most of the iron is in goethite and hematite, which have far fewer surface sites than hydrous ferric oxide. The fraction of iron in hydrous ferric oxide was arbitrarily assumed to be 0.1. Thus, a total of 0.34 mol of iron was assumed to be in hydrous ferric oxide, and using a value of 0.2 for the number of sites per mole of iron, a total of 0.07 mol of sites per liter was used in the calculations. A gram formula weight of 89 was used to estimate that the mass of hydrous ferric oxide was 30 g/L. The specific surface area was assumed to be 600 m^2/g.

Table 37.--Input data set for example 14

```
TITLE Example 14.--Transport with equilibrium_phases, exchange, and surface reactions
***********
PLEASE NOTE: This problem requires database file wateq4f.dat!!
***********
SURFACE_SPECIES
        Hfo_wOH + Mg+2 = Hfo_wOMg+ + H+
        log_k -15.
        Hfo_wOH + Ca+2 = Hfo_wOCa+ + H+
        log_k -15.
SOLUTION 1 Brine
        pH      5.713
```

```
        pe      4.0     O2(g)   -0.7
        temp    25.
        units   mol/kgw
        Ca      .4655
        Mg      .1609
        Na      5.402
        Cl      6.642           charge
        C       .00396
        S       .004725
        As      .05 umol/kgw
END
USE solution 1
EQUILIBRIUM_PHASES 1
        Dolomite        0.0     1.6
        Calcite         0.0     0.1
SAVE solution 1
# prints initial condition to the selected-output file
SELECTED_OUTPUT
        -file ex14.sel
        -reset false
        -step
USER_PUNCH
        -head  m_Ca m_Mg m_Na umol_As pH
10 PUNCH TOT("Ca"), TOT("Mg"), TOT("Na"), TOT("As")*1e6, -LA("H+")
END
PRINT
# skips print of initial exchange and initial surface to the selected-output file
        -selected_out false
EXCHANGE 1
        -equil with solution 1
        X       1.0
SURFACE 1
        -equil solution 1
# assumes 1/10 of iron is HFO
        Hfo_w           0.07    600.    30.
END
SOLUTION 0 20 x precipitation
        pH      4.6
        pe      4.0     O2(g)   -0.7
        temp    25.
        units   mmol/kgw
        Ca      .191625
        Mg      .035797
        Na      .122668
        Cl      .133704
        C       .01096
        S       .235153         charge
EQUILIBRIUM_PHASES 0
        Dolomite        0.0     1.6
        Calcite         0.0     0.1
        CO2(g)          -1.5    10.
SAVE solution 0
END
PRINT
```

```
        -selected_out true
ADVECTION
        -cells 1
        -shifts 200
END
```

The brine that initially fills the aquifer was taken from Parkhurst and others (1996) and is given as solution 1 in the input data set for this example (table 37). The pure-phase assemblage containing calcite and dolomite is defined with the **EQUILIBRIUM_PHASES** 1 data block. The brine is first equilibrated with calcite and dolomite and stored again as solution 1. The number of cation exchange sites is defined with **EXCHANGE** 1 and the number of surface sites are defined with **SURFACE** 1. The initial exchange and the initial surface composition are determined by equilibrium with the brine, after equilibration with calcite and dolomite (note that equilibration of exchangers and surfaces, *before* mineral equilibration, will yield different results due to buffering by the sorbed elements). The concentration of arsenic in the brine was determined by trial and error to give a total of approximately 2 mmol arsenic on the surface, which is consistent with the sequential extraction data. The database, *wateq4f.dat* (which includes the element arsenic and surface complexation constants from Dzombak and Morel, 1990), was used for all thermodynamic data except for two surface reactions. After initial runs it was determined that better results were obtained for arsenic concentrations in case the calcium and magnesium surface complexation reactions were removed. The **SURFACE_SPECIES** data block was used to decrease the equilibrium constant for each of these two reactions by about 10 orders of magnitude. This effectively eliminated surface complexation reactions for calcium and magnesium. (Alternatively, these reactions could be removed from the default database.) This is justified if cations and anions do not compete for the same sites; in general competitive sorption between cations and anions is not well known.

Recharge Water

The water entering the saturated zone of the aquifer was assumed to be in equilibrium with calcite and dolomite at a vadose-zone P_{CO_2} of $10^{-1.5}$. The fourth simulation in the input set (the simulation following the third **END** statement) generates this water composition and stores it as solution 0 using **SAVE** (table 37).

Advective Transport Calculations

The **ADVECTION** data block (table 37) provides the necessary information to advect the recharge water into the cell representing the saturated zone. A total of 200 shifts is specified, which is equivalent to 200 pore volumes because there is only a single cell in this calculation.

The results of the calculations are plotted on figure 14. During the initial 5 pore volumes, the high concentrations of sodium, calcium, and magnesium decrease such that sodium is the dominant cation and calcium and magnesium concentrations are small. The pH increases to more than 9.0 and arsenic concentrations increase to close to 2 μ mol/kgw. Over the next 45 pore volumes the pH gradually decreases and the arsenic concentrations decrease to negligible concentrations. At about 100 pore volumes, the calcium and magnesium become the dominant cations and the pH stabilizes at the pH of the infilling recharge water.

The advective transport calculations produce three types of water which are similar to water types observed in the aquifer: the initial brine, a sodium bicarbonate water, and a calcium and magnesium bicarbonate water. The

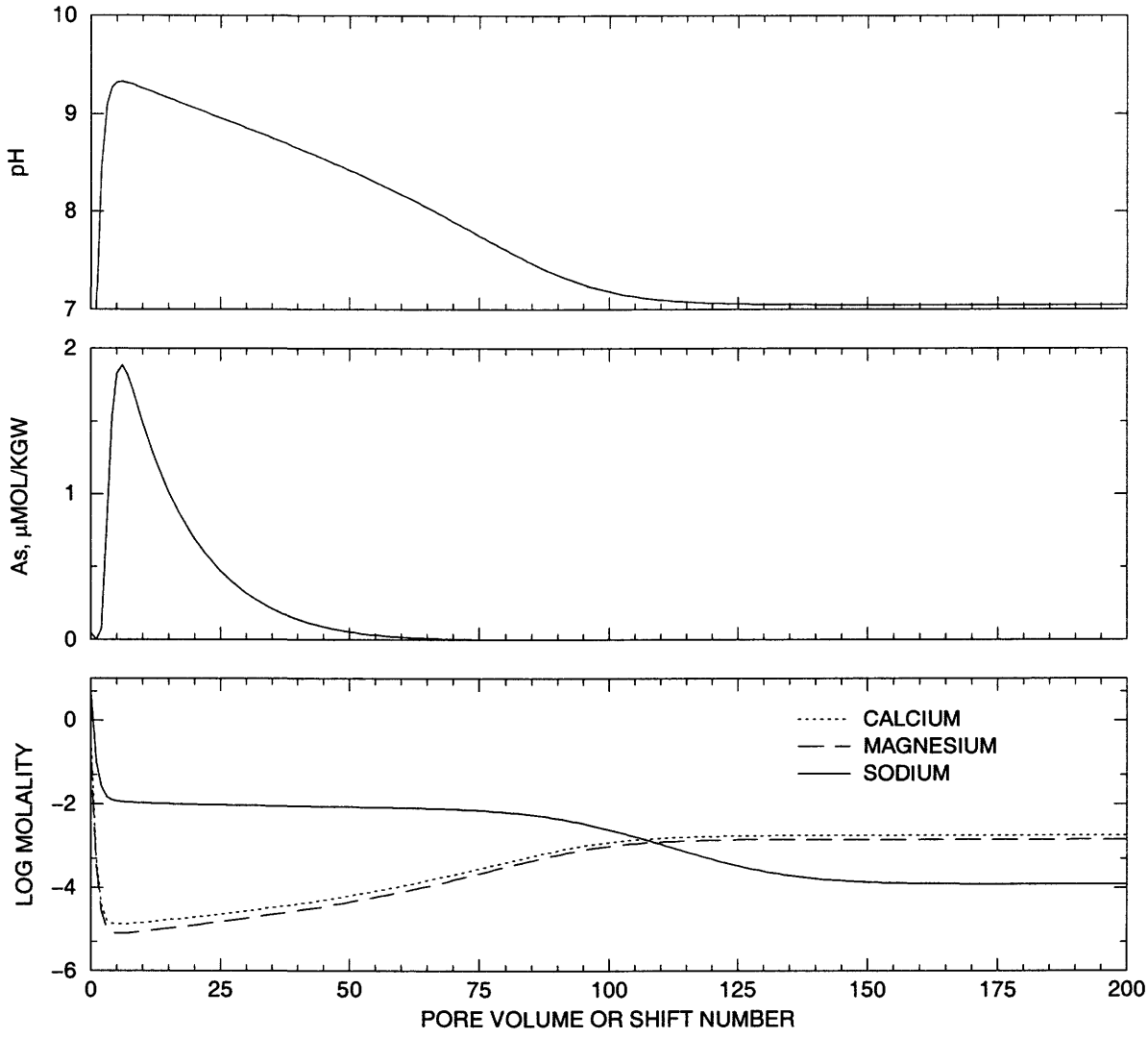

Figure 14.--Results of transport simulation of the chemical evolution of ground water due to calcium magnesium bicarbonate water inflow to an aquifer initially containing a brine, calcite and dolomite, a cation exchanger, and a surface that complexes arsenic.

calculated pH values are consistent with observations of aquifer water. In the sodium dominated waters, the calculated pH is generally greater than 8.0 and sometimes as high as 9.2; in the calcium magnesium bicarbonate waters, the pH is slightly greater than 7.0. Sensitivity calculations indicate that the maximum pH depends on the amount of exchanger present. Decreasing the number of cation exchange sites decreases the maximum pH. Simulated arsenic concentrations are similar to values observed in the aquifer, where the maximum concentrations are 1 to 2 µ mol/kgw. Lower maximum pH values produce lower maximum arsenic concentrations. The stability constant for the surface complexation reactions have been taken directly from the literature; a decrease in the log K for the predominant arsenic complexation reaction tends to decrease the maximum arsenic concentration as well. In conclusion, the model results, which were based largely on measured values and literature thermodynamic data, provide a satisfactory explanation of the variation in major ion chemistry, pH, and arsenic concentrations within the aquifer.

Example 15.--1D Transport: Kinetic Biodegradation, Cell Growth, and Sorption

A test problem for advective-dispersive-reactive transport was developed by Tebes-Steven and Valocchi (1997, 1998). Although based on relatively simple speciation chemistry, the solution to the problem demonstrates several interacting chemical processes that are common to many environmental problems: bacterially mediated degradation of an organic substrate; bacterial cell growth and decay; metal sorption; and aqueous speciation including metal-ligand complexation. In this example, the test problem is solved with PHREEQC, which produces results almost identical to those of Tebes-Steven and Valocchi (1997, 1998). However, care is needed in advective-dispersive transport simulations with PHREEQC, as with any reactive-transport model, to ensure that an accurate numerical solution is obtained.

The test problem models the transport processes when a pulse of water containing NTA (nitrylotriacetate) and cobalt is injected into a column. The problem includes advection and dispersion in the column, aqueous equilibrium reactions, and kinetic reactions for NTA degradation, growth of biomass, and cobalt sorption.

Transport Parameters

The dimensions and hydraulic properties of the column are given in table 38.

Table 38.--Hydraulic and physical properties of the column in example 15

Property	Value
Length of column	10.0 m
Porosity	.4
Bulk density	1.5e6 g/m^3
Grams of sediment per liter (from porosity and bulk density)	3.75e3 g/L
Pore water velocity	1.0 m/hr
Longitudinal dispersivity	.05 m

Aqueous Model

Tebes-Steven and Valocchi (1997) defined an aqueous model to be used for this test problem that includes the identity of the aqueous species and log K's of the species; activity coefficients were assumed to be 1.0. The database file in table 39 was constructed on the basis of their aqueous model. For the PHREEQC simulation, NTA was defined as a new "element" in the **SOLUTION_MASTER_SPECIES** data block named "Nta". From this point on "NTA" will be referred to as "Nta" for consistency with the PHREEQC notation. The gram formula weight of Nta in **SOLUTION_MASTER_SPECIES** is immaterial if input units are moles in the **SOLUTION** data block, and is simply set to 1. The aqueous complexes of Nta are defined in the **SOLUTION_SPECIES** data block. Note that the activity coefficients of all aqueous species are defined with a large value for the a parameter ($1x10^7$) in the **-gamma** identifier, which forces the activity coefficients to be very nearly 1.0.

Table 39.--Database for example 15

```
SOLUTION_MASTER_SPECIES
C         CO2            2.0     61.0173         12.0111
Cl        Cl-            0.0     Cl              35.453
Co        Co+2           0.0     58.93           58.93
E         e-             0.0     0.0             0.0
H         H+             -1.     1.008           1.008
H(0)      H2             0.0     1.008
H(1)      H+             -1.     1.008
N         NH4+           0.0     14.0067         14.0067
Na        Na+            0.0     Na              22.9898
Nta       Nta-3          3.0     1.              1.
O         H2O            0.0     16.00           16.00
O(-2)     H2O            0.0     18.016
O(0)      O2             0.0     16.00
SOLUTION_SPECIES
2H2O = O2 + 4H+ + 4e-
        log_k   -86.08; -gamma  1e7   0.0
2 H+ + 2 e- = H2
        log_k   -3.15;  -gamma  1e7   0.0
H+ = H+
        log_k   0.0;    -gamma  1e7   0.0
e- = e-
        log_k   0.0;    -gamma  1e7   0.0
H2O = H2O
        log_k   0.0;    -gamma  1e7   0.0
CO2 = CO2
        log_k   0.0;    -gamma  1e7   0.0
Na+ = Na+
        log_k   0.0;    -gamma  1e7   0.0
Cl- = Cl-
        log_k   0.0;    -gamma  1e7   0.0
Co+2 = Co+2
        log_k   0.0;    -gamma  1e7   0.0
NH4+ = NH4+
        log_k   0.0;    -gamma  1e7   0.0
Nta-3 = Nta-3
        log_k   0.0;    -gamma  1e7   0.0
Nta-3 + 3H+ = H3Nta
        log_k   14.9;   -gamma  1e7   0.0
Nta-3 + 2H+ = H2Nta-
        log_k   13.3;   -gamma  1e7   0.0
Nta-3 + H+ = HNta-2
        log_k   10.3;   -gamma  1e7   0.0
Nta-3 + Co+2 = CoNta-
        log_k   11.7;   -gamma  1e7   0.0
2 Nta-3 + Co+2 = CoNta2-4
        log_k   14.5;   -gamma  1e7   0.0
Nta-3 + Co+2 + H2O = CoOHNta-2 + H+
        log_k   0.5;    -gamma  1e7   0.0
Co+2 + H2O = CoOH+ + H+
        log_k   -9.7;   -gamma  1e7   0.0
Co+2 + 2H2O = Co(OH)2 + 2H+
        log_k   -22.9;  -gamma  1e7   0.0
```

```
Co+2 + 3H2O = Co(OH)3- + 3H+
        log_k   -31.5;  -gamma  1e7   0.0
CO2 + H2O = HCO3- + H+
        log_k   -6.35;  -gamma  1e7   0.0
CO2 + H2O = CO3-2 + 2H+
        log_k   -16.68; -gamma  1e7   0.0
NH4+ = NH3 + H+
        log_k   -9.3;   -gamma  1e7   0.0
H2O = OH- +  H+
        log_k   -14.0;  -gamma  1e7   0.0
END
```

Initial and Boundary Conditions

The background concentrations in the column are listed in table 40. The column contains no Nta or cobalt initially, but has a biomass of 1.36×10^{-4} g/L. A flux boundary condition is applied at the inlet of the column and for the first 20 hours a solution with Nta and cobalt enters the column; the concentrations in the pulse are also given in table 40. After 20 hours, the background solution is introduced at the inlet until the experiment ends after 75 hours. Na and Cl were not in the original problem definition, but were added for charge balancing sorption reactions for PHREEQC (see "Sorption Reactions" below).

Table 40.--Concentration data for example 15

Constituent	Type	Pulse concentration	Background concentration
H^+	Aqueous	10.0e-6 mol/L	10.0e-6 mol/L
Total C	Aqueous	4.9e-7 mol/L	4.9e-7 mol/L
NH_4^+	Aqueous	.0	.0
O_2	Aqueous	3.125e-5 mol/L	3.125e-5 mol/L
Nta_3^-	Aqueous	5.23e-6 mol/L	.0
Co^{2+}	Aqueous	5.23e-6 mol/L	.0
Na	Aqueous	1.0e-3 mol/L	1.0e-3 mol/L
Cl	Aqueous	1.0e-3 mol/L	1.0e-3 mol/L
Biomass	Immobile	---	1.36e-4 g/L
$CoNta_{(ads)}$	Immobile	---	.0
$Co_{(ads)}$	Immobile	---	.0

Kinetic Degradation of Nta and Cell Growth

Nta is assumed to degrade in the presence of biomass and oxygen by the reaction:
$HNta^{2-} + 1.62O_2 + 1.272H_2O + 2.424H^+ = 0.576C_5H_7O_2N + 3.12H_2CO_3 + 0.424NH_4^+$.
PHREEQC requires kinetic reactants to be defined solely by the moles of each element that enter or leave the solution due to the reaction. Furthermore, the reactants should be charge balanced (no net charge should enter or leave the

solution). The Nta reaction converts 1 mol $HNta^{2-}$ ($C_6H_7O_6N$) to 0.576 mol $C_5H_7O_2N$, where the latter is chemically inert so that its concentration can be discarded. The difference in elemental mass contained in these two reactants provides the stoichiometry of the elements C, H, O and N in the reaction. This stoichiometry is equal to the sum of the elements on the right-hand side of the equation, excluding $C_5H_7O_2N$, minus the sum of the elements on the left-hand side of the equation. The corresponding change in aqueous element concentrations per mole of $HNta^{2-}$ reaction is given in Table 41 (positive coefficients indicate an increase in aqueous concentration, negative coefficient indicates a decrease in aqueous concentration).

Table 41.--Reaction stoichiometry for oxidation of Nta

Component	Coefficient
Nta	-1.0
C	3.12
H	1.968
O	4.848
N	.424

The following multiplicative Monod rate expression is used to describe the rate of Nta degradation:

$$R_{HNTA^{2-}} = -q_m X_m \left(\frac{c_{HNTA^{2-}}}{K_s + c_{HNTA^{2-}}} \right) \left(\frac{c_{O_2}}{K_a + c_{O_2}} \right), \tag{164}$$

where $R_{HNTA^{2-}}$ is the rate of $HNta^{2-}$ degradation (mol/L/hr), q_m is the maximum specific rate of substrate utilization (mol/g cells/hr), X_m is the biomass (g cells/L), K_s is the half-saturation constant for the substrate Nta. (mol/L), K_a is the half-saturation constant for the electron acceptor O_2 (mol/L), and c_i indicate concentration (mol/L). The rate of biomass production is dependent on the rate of substrate utilization and a first-order decay rate for the biomass:

$$R_{cells} = -YR_{HNTA^{2-}} - bX_m, \tag{165}$$

where R_{cells} is the rate of cell growth (g cells/L/hr), Y is the microbial yield coefficient (g cells/mol Nta), and b is the first-order biomass decay coefficient (hr^{-1}). The parameter values for these equations are listed in table 42.

Table 42.--Kinetic rate parameters used in example 15

Parameter	Description	Parameter value
K_s	Half-saturation constant for donor	7.64e-7 mol/L
K_a	Half-saturation constant for acceptor	6.25e-6 mol/L
q_m	Maximum specific rate of substrate utilization	1.418e-3 mol Nta/g cells/hr
Y	Microbial yield coefficient	65.14 g cells/mol Nta
b	First-order microbial decay coefficient	0.00208 hr^{-1}

Sorption Reactions

Tebes-Steven and Valocchi (1997) defined kinetic sorption reactions for Co^{2+} and $CoNta^-$ by the rate equation:

$$R_i = -k_m\left(c_i - \frac{s_i}{K_d}\right), \tag{166}$$

where i is either Co^{2+} or $CoNta^-$ (mol/L), s_i is the sorbed concentration (mol/g sediment), k_m is the mass transfer coefficient (hr^{-1}), and K_d is the distribution coefficient (L/g). The values of the coefficients are given in table 43

Table 43.--Sorption coefficients for Co^{2+} and $CoNta^-$

Species	k_m	K_d
Co^{2+}	1 hr^{-1}	5.07e-3 L/g
$CoNta^-$	1 hr^{-1}	5.33e-4 L/g

The values of K_d were defined to give retardation coefficients of 20 and 3 for Co^{2+} and $CoNta^-$ respectively. Because the sorption reactions are defined to be kinetic, the initial moles of these reactants and the rates of reaction are defined with **KINETICS** and **RATES** data blocks; no surface definitions (**SURFACE**, **SURFACE_MASTER_SPECIES**, or **SURFACE_SPECIES**) are needed. Furthermore, all kinetic reactants are immobile, so that the sorbed species are not transported.

When modeling with PHREEQC, kinetic reactants must be charge balanced. For sorption of Co^{2+} and $CoNta^-$, 1 mmol of NaCl was added to the solution definitions to have counter ions for the sorption process. The kinetic sorption reactions were then defined to remove or introduce (depending on the sign of the mole transfer) $CoCl_2$ and NaCoNta, which are charge balanced. To convert from moles sorbed per gram of sediment (s_i) to moles sorbed per liter of water, it is necessary to multiply by the grams of sediment per liter of water, 3.75e3 g/L.

Input Data Set

Table 44 shows the input data set derived from the preceding problem definition. Although rates have been given in units of mol/L/hr, rates in PHREEQC are always mol/s and all rates have been adjusted to seconds in the definition of rate expressions in the input data set. It is assumed that a volume of 1 L of water is in each cell, which is reasonable for the current problem because the mass of water in each solution is nearly 1 kg and the solutions are relatively dilute. If the mass of water in a solution deviates significantly from 1 kg, the assumption of a constant volume may break down.

The 10-meter column was discretized with 10 cells of 1 meter each. The first two **SOLUTION** data blocks define the infilling solution and the initial solution in cells 1 through 10.

The **RATES** data block defines the rate expressions for four kinetic reactions: HNta-2, Biomass, Co_sorption, and CoNta_sorption. The rate expressions are initiated with **-start**, defined with numbered Basic-language statements, and terminated with **-end**. The last statement of each expression is SAVE followed by a variable name. This variable is the number of moles of reaction over the time subinterval and is calculated from an instantaneous rate (mol/s) times the length of the time subinterval (s), which is given by the variable "TIME". Lines 30 and 20 in the first and second rate expressions and line 10 in the third and fourth rate expressions adjust parameters to units of

seconds from units of hours. The function "MOL" returns the concentration of a species (mol/kgw) and the function "M" returns the moles of the reactant for which the rate expression is being defined; "KIN" returns the moles of a kinetic reactant, which may be any of the reactants defined for a cell. The functions "PUT" and "GET" are used to save and retrieve a term that is common to both the HNta-2 and Biomass rate expressions.

The **KINETICS** data block defines the names of the rate expressions that apply to each cell; cells 1 through 10 are defined simultaneously in this example. For each rate expression that applies to a cell, the formula of the reactant (**-formula**) and the moles of the reactant initially present (**-m**, if needed) are defined. It is also possible to define a tolerance (**-tol**), in moles, for the accuracy of the numerical integration for a rate expression. Note that the HNta-2 rate expression generates a negative rate, so that coefficients in the formula that are positive remove elements from solution and coefficients that are negative add elements to solution. In general, if the product of the rate and the coefficient is positive, the element is entering solution and if the product is negative, the element is leaving solution. The biomass reaction adds "H 0.0", or no moles of hydrogen, which specifies that the kinetic reaction for biomass growth does not add or remove elements from solution. The assimilation of carbon and nutrients that is associated with biomass growth is ignored in this simulation.

The **SELECTED_OUTPUT** data block causes the molalities of the aqueous species Nta-3, CoNta-, HNta-2 and Co+2 to be written to the file *ex15.sel*. To each line in the file, the **USER_PUNCH** data block appends the time (in hours), the sorbed concentrations converted to mol/g sediment, and the biomass.

The first **TRANSPORT** data block defines the first 20 hours of the experiment, during which Nta and cobalt are added at the column inlet. The column is defined to have 10 cells (**-cells**) of length 1 m (**-length**). The duration of the advective-dispersive transport simulation is 20 time steps (**-shifts**) of 3600 seconds (**-time_step**). The direction of flow is forward (**-flow_direction**). Each end of the column is defined to have a flux boundary condition (**-boundary_condition**). The dispersivity is 0.05 m (**-dispersivity**) and the diffusion coefficient is set to zero (**-diffusion_coef**). Data are written to the selected-output file only for cell 10 (**-punch_cells**) after each shift (**-punch_frequency**). Data are written to the output file only for cell 10 (**-print_cells**) after every five shifts (**-print_frequency**)

After the first advective-dispersive transport simulation, a new infilling solution is defined (**SOLUTION** 0), which contains no Nta or cobalt. For the associated initial solution calculation, printing to the selected-output file is eliminated and then reinstated (**-selected_out** false and **-selected_out** true, in **PRINT** data blocks).

Finally, the second **TRANSPORT** data block defines the final 55 hours of the experiment, during which Nta and cobalt are not present in the infilling solution. All parameters are the same as in the previous **TRANSPORT** data block, only the number of advection steps (**-shifts**) is increased to 55.

Table 44.--Input data set for example 15

```
TITLE Example 15.--1D Transport: Kinetic Biodegradation, Cell Growth, and Sorption
***********
PLEASE NOTE: This problem requires database file ex15.dat!!
***********
SOLUTION 0 Pulse solution with Nta and cobalt
        units umol/L
        pH      6
        C       .49
        O(0)    62.5
        Nta     5.23
```

```
        Co      5.23
        Na      1000
        Cl      1000
END
SOLUTION 1-10 Background solution initially filling column     # 1-20
        units umol/L
        pH      6
        C       .49
        O(0)    62.5
        Na      1000
        Cl      1000
END
RATES Rate expressions for the four kinetic reactions
#
        HNta-2
        -start
10 Ks = 7.64e-7
20 Ka = 6.25e-6
30 qm = 1.407e-3/3600
40 f1 = MOL("HNta-2")/(Ks + MOL("HNta-2"))
50 f2 = MOL("O2")/(Ka + MOL("O2"))
60 rate = -qm * KIN("Biomass") * f1 * f2
70 moles = rate * TIME
80 PUT(rate, 1)   # save the rate for use in Biomass rate calculation
90 SAVE moles
        -end
#
        Biomass
        -start
10 Y = 65.14
20 b = 0.00208/3600
30 rate = GET(1)  # uses rate calculated in HTNA-2 rate calculation
40 rate = -Y*rate -b*M
50 moles = -rate * TIME
60 if (M + moles) < 0 then moles = -M
70 SAVE moles
        -end
#
        Co_sorption
        -start
10 km = 1/3600
20 kd = 5.07e-3
30 solids = 3.75e3
40 rate = -km*(MOL("Co+2") - (M/solids)/kd)
50 moles = rate * TIME
60 if (M - moles) < 0 then moles = M
70 SAVE moles
        -end
#
        CoNta_sorption
        -start
10 km = 1/3600
20 kd = 5.33e-4
30 solids = 3.75e3
```

```
40 rate = -km*(MOL("CoNta-") - (M/solids)/kd)
50 moles = rate * TIME
60 if (M - moles) < 0 then moles = M
70 SAVE moles
        -end
KINETICS 1-10 Four kinetic reactions for all cells     # 1-20
        HNta-2
                -formula C -3.12 H -1.968 O -4.848 N -0.424 Nta 1.
        Biomass
                -formula        H 0.0
                -m              1.36e-4
        Co_sorption
                -formula CoCl2
                -m      0.0
                -tol 1e-11
        CoNta_sorption
                -formula NaCoNta
                -m      0.0
                -tol 1e-11
SELECTED_OUTPUT
        -file   ex15.sel
        -mol    Nta-3 CoNta- HNta-2 Co+2
USER_PUNCH
        -heading        hours   Co_sorb CoNta_sorb      Biomass
        -start
10 punch TOTAL_TIME/3600 + 1800/3600    # TOTAL_TIME/3600 + 900/3600
20 punch KIN("Co_sorption")/3.75e3
30 punch KIN("CoNta_sorption")/3.75e3
40 punch KIN("Biomass")
        -end
TRANSPORT First 20 hours have Nta and cobalt in infilling solution
        -cells                  10                  #   20
        -length                 1                   #    0.5
        -shifts                 20                  #   40
        -time_step              3600                # 1800
        -flow_direction         forward
        -boundary_condition     flux flux
        -dispersivity           .05
        -correct_disp           true
        -diffusion_coef         0.0e-9
        -punch_cells            10                  #   20
        -punch_frequency        1                   #    2
        -print_cells            10                  #   20
        -print_frequency        5                   #   10
END
PRINT
        -selected_out false
SOLUTION 0 New infilling solution, same as background solution
        units umol/L
        pH      6
        C       .49
        O(0)    62.5
        Na      1000
        Cl      1000
```

```
END
PRINT
        -selected_out true
TRANSPORT Last 55 hours with background infilling solution
        -shifts                55                 #  110
END
```

Grid Convergence

With advective-dispersive-reactive transport simulations, it is always necessary to check the numerical accuracy of the results. In general, there will not be analytical solutions for these complex simulations, so the only test of numerical accuracy is to refine the grid and time step, rerun the simulation, and compare the results. If simulations on two different grids give similar results, there is some assurance that the numerical errors are relatively small. If simulations on two different grids give significantly different results, the grid must be refined again and the process repeated. Unfortunately, doubling the grid size at least quadruples the number of solution calculations that must be made because the number of cells doubles and the time step is halved. If the cell size approaches the size of the dispersivity, it may require even more solution calculations because the number of mix steps in the dispersion calculation will increase as well.

Table 45.--Revised TRANSPORT data block for example 15 for grid refinement to a 20-cell model

```
TRANSPORT Last 55 hours with background infilling solution
     -cells                20
     -length               0.5
     -shifts               40
     -time_step            1800
     -flow_direction       forward
     -boundary condition   flux  flux
     -dispersivity         .05
     -diffusion_coef       0.0e-9
     -punch_cells          20
     -punch_frequency      2
     -print_cells          20
     -print_frequency      10
END
```

To test grid convergence in this example, the number of cells in the column were doubled, for a total of 20 cells. All keyword data blocks that defined compositions for the range 1-10 were changed to 1-20. In addition, the parameters for advective-dispersive transport were adjusted to be consistent with the new number of cells. Table 45 shows the first **TRANSPORT** data block adjusted for 20 cells. The number of cells and number of shifts are doubled; the cell length and time step are halved. To print information for the same location as the 10-cell model (the end of the column), the **-punch_cells** and **-print_cells** are set to cell 20. To print information at the same time in the simulation as the 10-cell model, **-punch_frequency** is set to every two shifts, **-print_frequency** is set to every 10 shifts, and the time step for going from the cell-midpoint to the column-end is halved on line 10 in **USER_PUNCH**. All of the changes to make a 20-cell model are also noted in table 44 by comments at the end of lines.

Results

The distributions of aqueous and immobile constituents in the column at the end of 75 hours are shown in figures 15 and 16 for the 10- and 20-cell models. In the experiment, two pore volumes of water with Nta and cobalt were introduced to the column over the first 20 hours and then followed by 5.5 pore volumes of background water over the next 55 hours. At 10 hours, $HNta^{2-}$ begins to appear at the column outlet along with a rise in the pH (fig. 15). If Nta and cobalt were conservative and dispersion were negligible, the graph would show square pulses that increase at 10 hours and decrease at 30 hours. However, the movement of the Nta and cobalt is retarded relative to conservative movement by the sorption reactions. The peak in Nta and cobalt concentrations occurs in the $CoNta^-$ complex between 30 and 40 hours. The peak in Co^{2+} concentration is even more retarded by its sorption reaction and does not show up until near the end of the experiment.

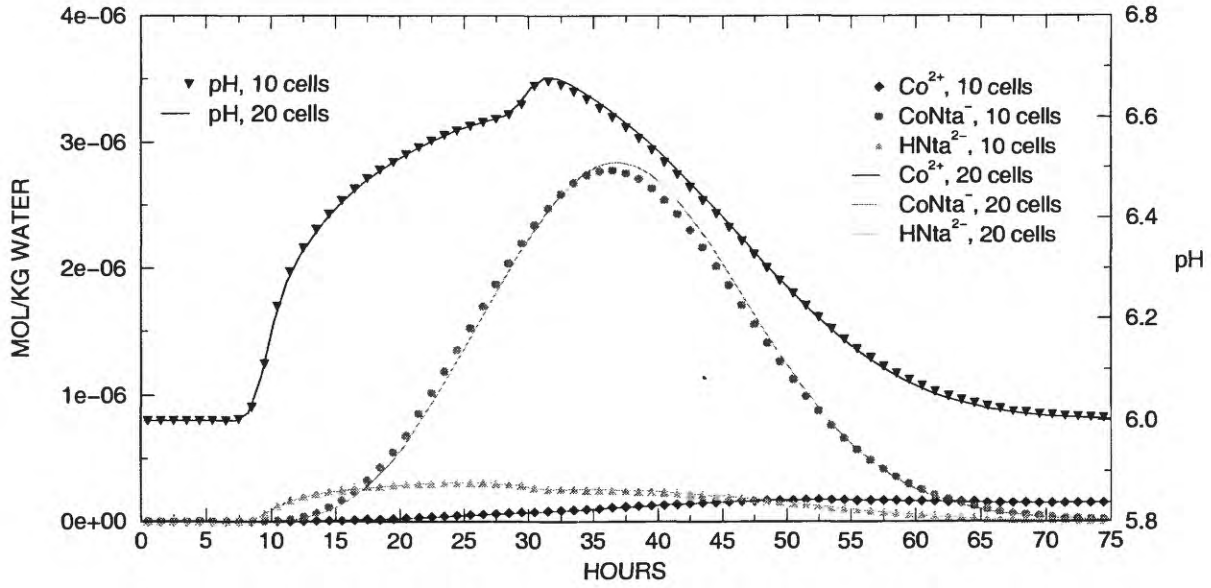

Figure 15.--Aqueous concentrations and pH values at the outlet of the column for Nta and cobalt transport simulations with 10 and 20 cells.

In figure 16, solid phase concentrations are plotted against time for concentrations in the last cell of the column. The sorbed $CoNta^-$ concentration peaks between 30 and 40 hours and slightly lags behind the peak in the dissolved concentration of the $CoNta^-$ complex. Initially, no Nta is present in the column and the biomass decreases slightly over the first 10 hours because of the first-order decay rate for the biomass. As the Nta moves through the cell, the biomass increases as the Nta substrate becomes available. After the peak in Nta has moved through the column, biomass concentrations level off and then begin to decrease because of decay. The K_d for cobalt sorption relates to a greater retardation coefficient than the K_d for $CoNta^-$ sorption, and the sorbed concentration of Co^{2+} appears to be still increasing at the end of the experiment.

Both the 10-cell and the 20-cell models give similar results, which indicates that the numerical errors in the advective-dispersive transport simulation are relatively small, and the results are very similar to results given by Tebes-Steven and Valocchi (1997, 1998). However, Tebes-Steven and Valocchi (1997) included another part to their test problem that increased the rate constants for the sorption reactions from 1 to 1000 hr^{-1}. The increased rate constants generate a stiff set of partial differential equations, in which the rate-limited processes occur on different

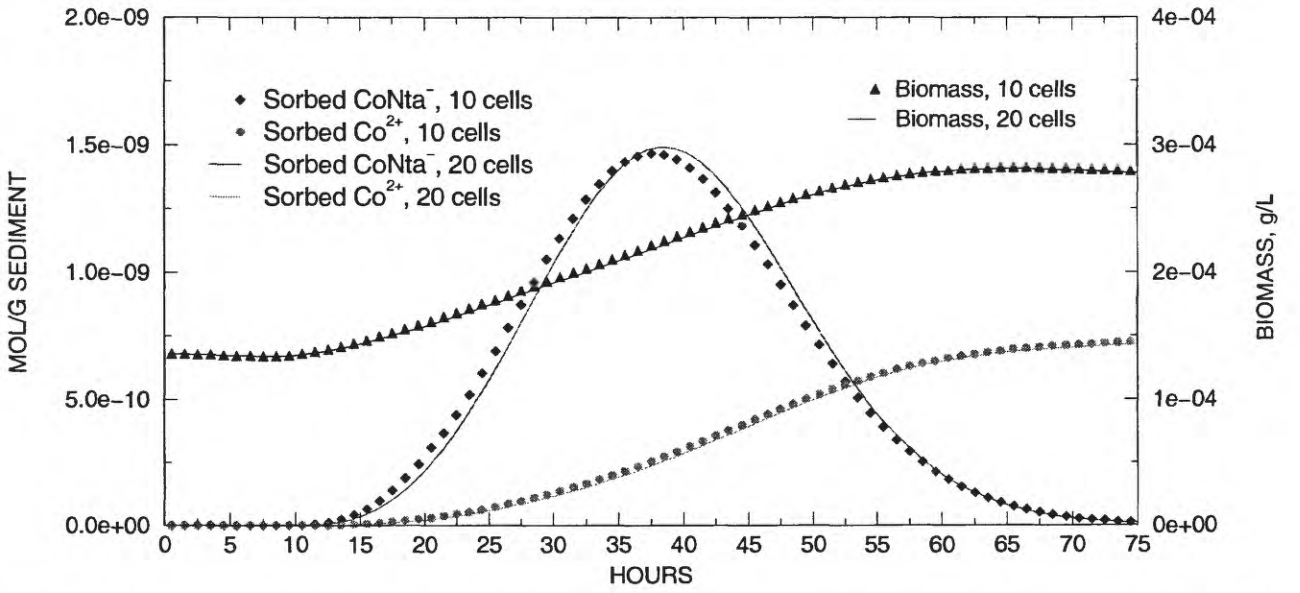

Figure 16.--Concentrations of sorbed species and biomass at the outlet of the column for Nta and cobalt transport simulations with 10 and 20 cells.

time scales. The stiff problem, with very fast sorption reactions, proved intractable for the explicit algorithm of PHREEQC, but could be solved successfully when the fast kinetic sorption reaction was calculated as the equilibrium process that it effectively constituted. However, even with equilibrium sorption, grid convergence was computationally much more intensive; it was necessary to use 100 cells or more to arrive at a satisfactory solution. As an estimate of relative CPU times, the 10-cell model took 270 seconds and the 20-cell model took 732 seconds to run on a Pentium I, 133 MHz computer. A 200-cell model took approximately 600 times more CPU time than the 10-cell model.

Example 16.--Inverse Modeling of Sierra Spring Waters

This example repeats the inverse modeling calculations of the chemical evolution of spring-water compositions in the Sierra Nevada that are described in a classic paper by Garrels and Mackenzie (1967). The same example is described in the manual for the inverse-modeling program NETPATH (Plummer and others, 1991 and 1994). The example uses two spring-water compositions, one from an ephemeral spring, which is less chemically evolved, and one from a perennial spring, which probably has had a longer residence time in the subsoil. The differences in composition between the ephemeral and perennial spring are assumed to be due to reactions between the water and the minerals and gases it contacts. The object of inverse modeling in this example is to find sets of minerals and gases that, when reacted in appropriate amounts, quantitatively account for the differences in composition between the solutions.

NETPATH (Plummer and others 1991, 1994) and PHREEQC are both capable of performing inverse-modeling calculations. NETPATH has two advantages relative to PHREEQC: (1) NETPATH provides a thorough treatment of isotopes, including isotopic mole balance, isotope fractionation, and carbon-14 dating, whereas PHREEQC has only isotope mole-balance capabilities, and (2) NETPATH provides a completely interactive environment for data entry and model development, whereas PHREEQC (version 2) is primarily a batch-oriented program. On the other hand,

Table 46.--Analytical data for spring waters in example 16

[Analyses in millimoles per liter from Garrels and Mackenzie (1967)]

Spring	pH	SiO_2	Ca^{2+}	Mg^{2+}	Na^+	K^+	HCO_3^-	SO_4^{2-}	Cl^-
Ephemeral spring	6.2	0.273	0.078	0.029	0.134	0.028	0.328	0.010	0.014
Perennial spring	6.8	.410	.260	.071	.259	.040	.895	.025	.030

complete graphical user interfaces are available for PHREEQC version 1, which lacks the isotope mole-balance capabilities (Charlton and others, 1997) and for version 2 (PHREEQC for Windows, V.E.A. Post, written commun., 1999, *http://www.geo.vu.nl/users/posv/phreeqc.html*). The major advantage of inverse modeling with PHREEQC relative to NETPATH is the capability to include uncertainties in the analytical data that are used in the calculation of inverse models. This capability produces inverse models that are more robust, that is, small changes in input data do not produce large differences in model-calculated mole transfers. Another advantage of PHREEQC is that any set of elements may be included in the inverse-modeling calculations, whereas NETPATH is limited to a selected, though relatively comprehensive, set of elements.

Table 47.--Reactant compositions and mole transfers given by Garrels and Mackenzie (1967)

[Mole transfer in millimoles per kilogram water, positive numbers indicate dissolution and negative numbers indicate precipitation]

Reactant	Composition	Mole transfer
"Halite"	NaCl	0.016
"Gypsum"	$CaSO_4 \cdot 2H_2O$	.015
Kaolinite	$Al_2Si_2O_5(OH)_4$	-.033
Ca-Montmorillonite	$Ca_{0.17}Al_{2.33}Si_{3.67}O_{10}(OH)_2$	-.081
CO2gas	CO_2	.427
Calcite	$CaCO_3$	.115
Silica	SiO_2	.0
Biotite	$KMg_3AlSi_3O_{10}(OH)_2$	.014
Plagioclase	$Na_{0.62}Ca_{0.38}Al_{1.38}Si_{2.62}O_8$	.175

The analytical data for the two springs are given in table 46. The chemical compositions of minerals and gases postulated to react by Garrels and Mackenzie (1967) and their mole transfers are given in table 47. The selection of the identity and composition of the reactive phases is the most difficult part of inverse modeling. In general the selection is made by knowledge of the flow system and the mineralogy along the flow path; microscopic and chemical analysis of the aquifer material and isotopic composition of the water and minerals provide additional insight for the selection of reactants. It is not necessary to know precisely which minerals are reacting, but it is necessary to have a comprehensive list of potential reactants.

The input data set for this example is given in table 48. The **SOLUTION_SPREAD** data block is used to define the two spring waters. The **INVERSE_MODELING** data block is used to define all of the characteristics

of the inverse-modeling calculations, including the solutions and phases to be used, the mole-balance equations, the uncertainty limits, whether all or only "minimal" models will be printed, and whether ranges of mole transfer that are consistent with the uncertainty limits will be calculated. A series of identifiers (sub-keywords preceded by a hyphen) specify the characteristics of the inverse model.

Table 48.--Input data set for example 16

```
TITLE Example 16.--Inverse modeling of Sierra springs
SOLUTION_SPREAD
        -units mmol/L
# "\t" indicates tab
Number\t  pH\t     Si\t      Ca\t      Mg\t      Na\t       K\t Alkalinity\t   S(6)\t      Cl
1\t       6.2\t  0.273\t   0.078\t   0.029\t   0.134\t   0.028\t      0.328\t   0.01\t   0.014
2\t       6.8\t  0.41 \t   0.26 \t   0.071\t   0.259\t   0.04 \t      0.895\t   0.025\t  0.03
INVERSE_MODELING 1
        -solutions 1 2
        -uncertainty 0.025
        -balances
                Ca      0.05      0.025
        -phases
                Halite
                Gypsum
                Kaolinite               precip
                Ca-montmorillonite      precip
                CO2(g)
                Calcite
                Chalcedony              precip
                Biotite                 dissolve
                Plagioclase             dissolve
        -range
PHASES
Biotite
   KMg3AlSi3O10(OH)2 + 6H+ + 4H2O = K+ + 3Mg+2 + Al(OH)4- + 3H4SiO4
        log_k  0.0      # No log_k, Inverse modeling only
Plagioclase
    Na0.62Ca0.38Al1.38Si2.62O8 + 5.52 H+ + 2.48H2O =\
                0.62Na+ + 0.38Ca+2 + 1.38Al+3 + 2.62H4SiO4
        log_k  0.0      # No log_k, Inverse modeling only
END
```

The identifier **-solutions** selects the solutions to be used by solution number. Two or more solution numbers must be listed after the identifier. If only two solution numbers are given, the second solution is assumed to evolve from the first solution. If more than two solution numbers are given, the last solution listed is assumed to evolve from a mixture of the preceding solutions. The solutions to be used in inverse modeling are defined in the same way as any solutions used in PHREEQC models. Usually the analytical data are entered in a **SOLUTION** or **SOLUTION_SPREAD** data block, but solutions defined by batch-reaction calculations in the current or previous simulations may also be used if they are saved with the **SAVE** keyword.

The **-uncertainty** identifier sets the default uncertainty limit for each analytical datum. In this example a fractional uncertainty limit of 0.025 (2.5 percent) is assumed for all of the analytical data except pH. By default, the uncertainty limit for pH is 0.05 unit. The uncertainty limit for pH can be set to an absolute value (standard units) with **-balances** identifier. The uncertainty limit for any datum for any of the solutions can be set explicitly to a fractional value or an absolute value (moles; equivalents for alkalinity) using the **-balances** identifier.

By default, every inverse model includes mole-balance equations for every element in any of the phases included in **-phases** (except hydrogen and oxygen). If mole-balance equations are needed for elements not included in the phases, that is, for elements with no source or sink (conservative mixing), the **-balances** identifier can be used to include those elements in the formulation of the inverse-modeling equations (see example 17). In addition, the **-balances** identifier can be used to specify uncertainty limits for an element in each solution. For demonstration purposes in the example, the uncertainty limit for calcium is set to 0.05 (5 percent) in solution 1 and 0.025 (2.5 percent) in solution 2.

The phases to be used in the inverse-modeling calculations are defined with the **-phases** identifier. In addition, this identifier can be used to constrain any phases to dissolve only or precipitate only. In this example, kaolinite, Ca-montmorillonite, and chalcedony (SiO_2) are required to precipitate only. This means that kaolinite will be precipitating (negative mole transfer) in any model that contains the phase kaolinite; likewise for Ca-montmorillonite and chalcedony. Biotite and plagioclase are required to dissolve (positive mole transfer) if they are present in an inverse model.

All of the phases used in inverse modeling must be defined in **PHASES** or **EXCHANGE_SPECIES** data blocks, either in the database file or the input file. Thus, all phases defined in the default database file, *phreeqc.dat* or *wateq4f.dat*, are available for use in inverse modeling. Biotite and plagioclase are not in the default database file *phreeqc.dat* and so they are defined explicitly in the **PHASES** data block in the input data set. For simplicity, the log *K*'s are set to zero for these phases, which does not affect inverse modeling because only the mineral stoichiometry is used; however, the saturation indices calculated for these phases will be spurious. All phases used in inverse modeling must have a charge-balanced reaction. This requirement is due to the inclusion of a charge-balance constraint for each solution. Each solution is adjusted to charge balance for each model by adjusting the concentrations of the elements within their uncertainty limits while minimizing the objective function of the optimization method (see "Equations and Numerical Method for Inverse Modeling"). If a solution can not be adjusted to charge balance using the given uncertainty limits, the solution will be noted in the output and no models will be found. Because all of the solutions are charge balanced in the modeling process, phases must also be charge balanced or they will not be included in any models. Note that the reaction for plagioclase (table 48) is on two lines, but the program interprets the two lines to be a single logical line because of the backslash "\" at the end of the first of these two lines.

The **-range** identifier indicates that, in addition to finding all of the inverse models, each model that is found will be subjected to additional calculations to determine the range of values that each mole transfer may have, within the constraints of the uncertainty limits.

The following equations are included for every inverse model: mole balance for each element or valence state of each element in the system (as defined by elements in the phases of **-phases** and each element listed in **-balances**), charge balance for each solution, alkalinity balance for the system, electron balance for the system, and water balance for the system. The unknowns in these equations include the mole transfers of phases, the mole transfers of redox reactions, and the uncertainty unknowns for each element in each solution (excluding hydrogen and oxygen). An uncertainty unknown is included for alkalinity and pH for each solution. The optimization method solves for a set of values for the unknowns that satisfies all of the equations, satisfies all of the uncertainty limits, and simultaneously minimizes the objective function, which is a weighted sum of the uncertainty unknowns (see "Equations and Numerical Method for Inverse Modeling").

Results for the two inverse models found in this example are shown in table 49. The results begin with a listing of three columns for each solution that is part of the model. All columns are values in mol/kgw, except pH (and isotopic values if included). The first column contains the original analytical data for the solution. (Input). The second column contains any adjustments to the analytical data calculated for the model (Delta). These adjustments must be within the specified uncertainty limits. The third column contains the revised analytical data for the solution, which are equal to the original data plus any adjustment (Input+Delta).

After the listing of the solutions, the relative fractions of each solution in the inverse model are printed (Solution fractions). With only two solutions in the model, normally the fraction for each solution will be 1.0. If more than two solutions are included in the inverse model, normally the sum of the fractions of the solutions, excluding the last solution, will equal 1.0. The fractions are actually derived from a mole balance on water, so if hydrated minerals consume or produce significant amounts of water or if evaporation is modeled (see example 17), the numbers may not sum to 1.0. In this example, all fractions are identically 1.0; the amount of water from gypsum dissolution is too small to affect the four significant figures of the mixing fractions. The second and third column for the block giving solution fractions are the minimum and maximum fractional values that can be attained within the constraints of the specified uncertainty limits. These two columns are nonzero only if the **-range** identifier is used.

The next block of data in the listing contains three columns describing the mole transfers for the phases (Phase mole transfers). The first column contains the inverse model that is consistent with the adjusted concentrations printed in the listing of the solutions. In this example, the adjusted solution 1 plus the mole transfers in the first column exactly equals the adjusted solution 2. Mole transfers that are positive indicate dissolution; mole transfers that are negative indicate precipitation. (Note that mole transfers in phase assemblages in batch-reaction calculations are relative to the phase, not relative to solution.) The second and third columns of mole transfers are the minimum and maximum mole transfers of each phase that can be attained within the constraints of the specified uncertainty limits. These two columns are nonzero only if the **-range** identifier is used. In general, these minima and maxima are not independent, that is, obtaining a maximum mole transfer of one phase places very strong constraints on the mole transfers of the other phases in a mole-balance model.

No redox mole transfers were calculated in this inverse model. If any redox mole transfers had been calculated, the moles transferred between valence states of each element would be printed under the heading "Redox mole transfers".

The next block of data prints results related to the extent to which the analytical data were adjusted for this model; if no adjustments were made, all three quantities that are printed would be zero. First the sum of residuals is printed, which is a sum of the uncertainty unknowns weighted by the inverse of the uncertainty limit ($\sum_q \sum_m \frac{\varepsilon_{m,q}}{u_{m,q}} = \sum_q \sum_m \frac{\alpha_q \delta_{m,q}}{u_{m,q}}$). Next, a sum of the adjustment to each element concentration and isotopic composition that is weighted by the inverse of the uncertainty limit is printed ($\sum_q \sum_m \frac{\delta_{m,q}}{u_{m,q}}$, Sum of delta/uncertainty limit). Finally, the maximum fractional adjustment to any element or isotopic composition in any solution is printed (Maximum fractional error in element concentration). All three values apply to the model printed in the left-hand column.

For a given mole-balance model, if no simpler inverse model can be found with any proper subset of the solutions and phases of the model, the statement "Model contains minimum number of phases" is printed for the given model.

After all models are printed, a short summary of the calculations is presented, which lists the number of models found, the number of minimal models found (models with a minimum number of phases), the number of infeasible models that were tested, and the number of calls to the inequality equations solver, cl1 (calculation time is generally proportional to the number of calls to cl1).

The results of the example show that two inverse models exist using the phases suggested by Garrels and Mackenzie (1967). The main reactions are dissolution of calcite and plagioclase, which consume carbon dioxide; kaolinite and Ca-montmorillonite precipitate in the first model, and kaolinite and chalcedony precipitate in the second model. Small amounts of halite, gypsum, and biotite dissolution are required in the models. The results of Garrels and Mackenzie (1967) fall within the range of mole transfers calculated in the first model of PHREEQC for all phases except carbon dioxide. The carbon dioxide mole transfer for the first model differs from Garrels and Mackenzie (1967) because they did not account for the dissolved carbon dioxide in the spring waters. Garrels and Mackenzie (1967) also ignored a small discrepancy in the mole balance for potassium. PHREEQC avoids the potassium imbalance by adjusting concentrations of the elements in the two solutions. The PHREEQC calculations show that two inverse models can be found by adjusting concentrations by no more than the specified uncertainty limits (2.5 percent). Without making the calculations with PHREEQC and considering the magnitude of uncertainties, it is not clear whether the discrepancy in potassium that was ignored by Garrels and Mackenzie is significant. The results of PHREEQC are concordant with the results of NETPATH, except that NETPATH also must ignore the discrepancy in the potassium mole balance.

Table 49.--Selected output for example 16

```
Solution 1:

                        Input          Delta       Input+Delta
             pH      6.200e+00  +   1.246e-02  =   6.212e+00
             Al      0.000e+00  +   0.000e+00  =   0.000e+00
     Alkalinity      3.280e-04  +   5.500e-06  =   3.335e-04
           C(-4)     0.000e+00  +   0.000e+00  =   0.000e+00
           C(4)      7.825e-04  +   0.000e+00  =   7.825e-04
             Ca      7.800e-05  +  -3.900e-06  =   7.410e-05
             Cl      1.400e-05  +   0.000e+00  =   1.400e-05
           H(0)      0.000e+00  +   0.000e+00  =   0.000e+00
              K      2.800e-05  +  -7.000e-07  =   2.730e-05
             Mg      2.900e-05  +   0.000e+00  =   2.900e-05
             Na      1.340e-04  +   0.000e+00  =   1.340e-04
           O(0)      0.000e+00  +   0.000e+00  =   0.000e+00
          S(-2)      0.000e+00  +   0.000e+00  =   0.000e+00
           S(6)      1.000e-05  +   0.000e+00  =   1.000e-05
             Si      2.730e-04  +   0.000e+00  =   2.730e-04

Solution 2:

                        Input          Delta       Input+Delta
             pH      6.800e+00  +  -3.407e-03  =   6.797e+00
             Al      0.000e+00  +   0.000e+00  =   0.000e+00
     Alkalinity      8.951e-04  +  -1.796e-06  =   8.933e-04
           C(-4)     0.000e+00  +   0.000e+00  =   0.000e+00
           C(4)      1.199e-03  +   0.000e+00  =   1.199e-03
             Ca      2.600e-04  +   6.501e-06  =   2.665e-04
             Cl      3.000e-05  +   0.000e+00  =   3.000e-05
           H(0)      0.000e+00  +   0.000e+00  =   0.000e+00
              K      4.000e-05  +   1.000e-06  =   4.100e-05
             Mg      7.101e-05  +  -8.979e-07  =   7.011e-05
             Na      2.590e-04  +   0.000e+00  =   2.590e-04
           O(0)      0.000e+00  +   0.000e+00  =   0.000e+00
          S(-2)      0.000e+00  +   0.000e+00  =   0.000e+00
           S(6)      2.500e-05  +   0.000e+00  =   2.500e-05
             Si      4.100e-04  +   0.000e+00  =   4.100e-04

Solution fractions:                  Minimum         Maximum
   Solution    1      1.000e+00      1.000e+00       1.000e+00
   Solution    2      1.000e+00      1.000e+00       1.000e+00

Phase mole transfers:                Minimum         Maximum
```

```
        Halite      1.600e-05      1.490e-05      1.710e-05   NaCl
        Gypsum      1.500e-05      1.413e-05      1.588e-05   CaSO4:2H2O
     Kaolinite     -3.392e-05     -5.587e-05     -1.224e-05   Al2Si2O5(OH)4
Ca-Montmorillon    -8.090e-05     -1.100e-04     -5.154e-05   Ca0.165Al2.33Si3.67O10(OH)2
       CO2(g)      2.928e-04      2.363e-04      3.563e-04   CO2
       Calcite     1.240e-04      1.007e-04      1.309e-04   CaCO3
       Biotite     1.370e-05      1.317e-05      1.370e-05   KMg3AlSi3O10(OH)2
    Plagioclase    1.758e-04      1.582e-04      1.935e-04   Na0.62Ca0.38Al1.38Si2.62O8

Redox mole transfers:

Sum of residuals (epsilons in documentation):         5.574e+00
Sum of delta/uncertainty limit:                       5.574e+00
Maximum fractional error in element concentration:    5.000e-02

Model contains minimum number of phases.
================================================================================


Solution 1:

                      Input          Delta     Input+Delta
            pH      6.200e+00  +   1.246e-02  =   6.212e+00
            Al      0.000e+00  +   0.000e+00  =   0.000e+00
    Alkalinity      3.280e-04  +   5.500e-06  =   3.335e-04
          C(-4)     0.000e+00  +   0.000e+00  =   0.000e+00
          C(4)      7.825e-04  +   0.000e+00  =   7.825e-04
            Ca      7.800e-05  +  -3.900e-06  =   7.410e-05
            Cl      1.400e-05  +   0.000e+00  =   1.400e-05
          H(0)      0.000e+00  +   0.000e+00  =   0.000e+00
             K      2.800e-05  +  -7.000e-07  =   2.730e-05
            Mg      2.900e-05  +   0.000e+00  =   2.900e-05
            Na      1.340e-04  +   0.000e+00  =   1.340e-04
          O(0)      0.000e+00  +   0.000e+00  =   0.000e+00
         S(-2)      0.000e+00  +   0.000e+00  =   0.000e+00
          S(6)      1.000e-05  +   0.000e+00  =   1.000e-05
            Si      2.730e-04  +   0.000e+00  =   2.730e-04

Solution 2:

                      Input          Delta     Input+Delta
            pH      6.800e+00  +  -3.407e-03  =   6.797e+00
            Al      0.000e+00  +   0.000e+00  =   0.000e+00
    Alkalinity      8.951e-04  +  -1.796e-06  =   8.933e-04
          C(-4)     0.000e+00  +   0.000e+00  =   0.000e+00
          C(4)      1.199e-03  +   0.000e+00  =   1.199e-03
            Ca      2.600e-04  +   6.501e-06  =   2.665e-04
            Cl      3.000e-05  +   0.000e+00  =   3.000e-05
          H(0)      0.000e+00  +   0.000e+00  =   0.000e+00
             K      4.000e-05  +   1.000e-06  =   4.100e-05
            Mg      7.101e-05  +  -8.980e-07  =   7.011e-05
            Na      2.590e-04  +   0.000e+00  =   2.590e-04
          O(0)      0.000e+00  +   0.000e+00  =   0.000e+00
         S(-2)      0.000e+00  +   0.000e+00  =   0.000e+00
          S(6)      2.500e-05  +   0.000e+00  =   2.500e-05
            Si      4.100e-04  +   0.000e+00  =   4.100e-04

Solution fractions:                 Minimum         Maximum
   Solution   1     1.000e+00      1.000e+00      1.000e+00
   Solution   2     1.000e+00      1.000e+00      1.000e+00

Phase mole transfers:               Minimum         Maximum
        Halite      1.600e-05      1.490e-05      1.710e-05   NaCl
        Gypsum      1.500e-05      1.413e-05      1.588e-05   CaSO4:2H2O
     Kaolinite     -1.282e-04     -1.403e-04     -1.159e-04   Al2Si2O5(OH)4
       CO2(g)      3.061e-04      2.490e-04      3.703e-04   CO2
       Calcite     1.106e-04      8.680e-05      1.182e-04   CaCO3
    Chalcedony    -1.084e-04     -1.473e-04     -6.906e-05   SiO2
       Biotite     1.370e-05      1.317e-05      1.370e-05   KMg3AlSi3O10(OH)2
    Plagioclase    1.758e-04      1.582e-04      1.935e-04   Na0.62Ca0.38Al1.38Si2.62O8

Redox mole transfers:

Sum of residuals (epsilons in documentation):         5.574e+00
Sum of delta/uncertainty limit:                       5.574e+00
Maximum fractional error in element concentration:    5.000e-02

Model contains minimum number of phases.
================================================================================


Summary of inverse modeling:

        Number of models found: 2
        Number of minimal models found: 2
        Number of infeasible sets of phases saved: 20
        Number of calls to cl1: 62
```

Example 17.--Inverse Modeling with Evaporation

Evaporation is handled in the same manner as other heterogeneous reactions for inverse modeling. To model evaporation (or dilution) it is necessary to include a phase with the composition H_2O. The important concept in modeling evaporation is the water mole-balance equation that is included in every inverse problem formulation (see

"Equations and Numerical Method for Inverse Modeling"). The moles of water in the initial solutions times their mixing fractions plus water gained or lost by dissolution or precipitation of phases plus water gained or lost through redox reactions must equal the moles of water in the final solution. The equation is approximate because it does not include the moles of water gained or lost in homogeneous hydrolysis and complexation reactions.

Table 50.--Input data set for example 17

```
TITLE Example 17.--Inverse modeling of Black Sea water evaporation
SOLUTION 1  Black Sea water
        units   mg/L
        density 1.014
        pH      8.0     # estimated
        Ca      233
        Mg      679
        Na      5820
        K       193
        S(6)    1460
        Cl      10340
        Br      35
        C       1       CO2(g) -3.5
SOLUTION 2  Composition during halite precipitation
        units   mg/L
        density 1.271
        pH      5.0     # estimated
        Ca      0.0
        Mg      50500
        Na      55200
        K       15800
        S(6)    76200
        Cl      187900
        Br      2670
        C       1       CO2(g) -3.5
INVERSE_MODELING
        -solution 1 2
        -uncertainties .025
        -range
        -balances
                Br
                K
                Mg
        -phases
                H2O(g)  pre
                Calcite pre
                CO2(g)  pre
                Gypsum  pre
                Halite  pre
END
```

This example uses data for the evaporation of Black Sea water that is presented in Carpenter (1978). Two analyses are selected, the initial Black Sea water and a water composition during the stage of evaporation in which halite precipitates. The hypothesis is that evaporation, precipitation of calcite, gypsum, and halite, and loss of carbon dioxide are sufficient to account for the changes in water composition of all of the major ions and bromide.

The input data set (table 50) contains the solution compositions in the **SOLUTION** data blocks. The total carbon in the solutions is unknown but is estimated by assuming that both solutions are in equilibrium with atmospheric carbon dioxide.

The **INVERSE_MODELING** keyword defines the inverse model for this example. Solution 2, the solution during halite precipitation, evolves from solution 1, Black Sea water. Uncertainty limits of 2.5 percent are applied to all data. Water, calcite, carbon dioxide, gypsum, and halite are specified to be the potential reactants (**-phases**). Each of these phases must precipitate, that is, must be removed from the aqueous phase in any valid inverse model.

By default, mole-balance equations for water, alkalinity, and electrons are included in the inverse formulation. In addition, mole-balance equations are included by default for all elements in the specified phases. In this case, calcium, carbon, sulfur, sodium, and chloride mole-balance equations are included by the default. The **-balances** identifier is used to specify additional mole-balance equations for bromide, magnesium, and potassium. In the absence of alkalinity data, the calculated alkalinity of these solutions is controlled by the choice of pH and the assumption that the solutions are in equilibrium with atmospheric carbon dioxide. For reasonable values of pH, alkalinity is a minor contributor to charge balance.

Only one model is found in the inverse calculation. This model indicates that Black Sea water (solution 1) must be concentrated 88 fold to produce solution 2, as shown by the fractions of the two solutions in the inverse-model output (table 51). Thus approximately 88 kg of water in Black Sea water is reduced to 1 kg of water in solution 2. Halite precipitates (19.75 mol) and gypsum precipitates (0.48 mol) during the evaporation process. Note that these mole transfers are relative to 88 kg of water. To find the loss per kilogram of water in Black Sea water, it is necessary to divide by the mixing fraction of solution 1. The result is that 54.9 mol of water, 0.0004 mol of calcite, 0.0004 mol carbon dioxide, 0.0054 mol of gypsum, and 0.22 mol of halite have been removed per kilogram of Black Sea water. (This calculation could be accomplished by making solution 1 from solution 2, taking care to reverse the constraints on minerals from precipitation to dissolution.) All other ions--magnesium, potassium, and bromide--are conservative within the 2.5-percent uncertainty limit that was specified. The inverse modeling shows that, with the given uncertainty limits, evaporation (loss of water), carbon dioxide outgassing, and calcite, halite, and gypsum precipitation are sufficient to account for all of the changes in major ion composition between the two solutions.

Table 51.--Selected output for example 17

```
Solution 1: Black Sea water

                    Input          Delta     Input+Delta
              pH  8.000e+00  +   0.000e+00  =   8.000e+00
      Alkalinity  8.625e-04  +   0.000e+00  =   8.625e-04
              Br  4.401e-04  +   0.000e+00  =   4.401e-04
           C(-4)  0.000e+00  +   0.000e+00  =   0.000e+00
            C(4)  8.284e-04  +   0.000e+00  =   8.284e-04
              Ca  5.841e-03  +   0.000e+00  =   5.841e-03
              Cl  2.930e-01  +   7.845e-04  =   2.938e-01
            H(0)  0.000e+00  +   0.000e+00  =   0.000e+00
               K  4.959e-03  +   1.034e-04  =   5.063e-03
              Mg  2.806e-02  +  -7.016e-04  =   2.736e-02
              Na  2.544e-01  +   0.000e+00  =   2.544e-01
            O(0)  0.000e+00  +   0.000e+00  =   0.000e+00
           S(-2)  0.000e+00  +   0.000e+00  =   0.000e+00
            S(6)  1.527e-02  +   7.768e-05  =   1.535e-02

Solution 2: Composition during halite precipitation

                    Input          Delta     Input+Delta
              pH  5.000e+00  +   2.148e-13  =   5.000e+00
      Alkalinity -9.195e-06  +   0.000e+00  =  -9.195e-06
              Br  3.785e-02  +   9.440e-04  =   3.880e-02
           C(-4)  0.000e+00  +   0.000e+00  =   0.000e+00
            C(4)  7.019e-06  +   0.000e+00  =   7.019e-06
              Ca  0.000e+00  +   0.000e+00  =   0.000e+00
              Cl  6.004e+00  +   1.501e-01  =   6.154e+00
```

```
            H(0)      0.000e+00  +   0.000e+00  =   0.000e+00
              K       4.578e-01  +  -1.144e-02  =   4.463e-01
             Mg       2.353e+00  +   5.883e-02  =   2.412e+00
             Na       2.720e+00  +  -4.500e-02  =   2.675e+00
            O(0)      0.000e+00  +   0.000e+00  =   0.000e+00
           S(-2)      0.000e+00  +   0.000e+00  =   0.000e+00
            S(6)      8.986e-01  +  -2.247e-02  =   8.761e-01

Solution fractions:                    Minimum         Maximum
   Solution   1      8.815e+01       8.780e+01       8.815e+01
   Solution   2      1.000e+00       1.000e+00       1.000e+00

Phase mole transfers:                  Minimum         Maximum
          H2O(g)    -4.837e+03      -4.817e+03      -4.817e+03   H2O
         Calcite    -3.802e-02      -3.897e-02      -3.692e-02   CaCO3
          CO2(g)    -3.500e-02      -3.615e-02      -3.371e-02   CO2
          Gypsum    -4.769e-01      -4.907e-01      -4.612e-01   CaSO4:2H2O
          Halite    -1.975e+01      -2.033e+01      -1.901e+01   NaCl

Redox mole transfers:

Sum of residuals (epsilons in documentation):         1.947e+02
Sum of delta/uncertainty limit:                       7.804e+00
Maximum fractional error in element concentration:    2.500e-02

Model contains minimum number of phases.
================================================================================


Summary of inverse modeling:

        Number of models found: 1
        Number of minimal models found: 1
        Number of infeasible sets of phases saved: 6
        Number of calls to cl1: 22
```

Example 18.--Inverse Modeling of the Madison Aquifer

In this example, inverse modeling, including isotope mole-balance modeling, is applied to the evolution of water in the Madison aquifer in Montana. Plummer and others (1990) used mole-balance modeling to quantify the extent of dedolomitization at locations throughout the aquifer. In the dedolomitization process, anhydrite dissolution causes the precipitation of calcite and dissolution of dolomite. Additional reactions identified by mole-balance modeling include sulfate reduction, cation exchange, and halite and sylvite dissolution (Plummer and others, 1990). $\delta^{13}C$ and $\delta^{34}S$ data were used to corroborate the mole-balance models and carbon-14 was used to estimate ground-water ages (Plummer and others, 1990). Initial and final water samples were selected from a flow path that extends from north-central Wyoming northeast across Montana (Plummer and others, 1990, flow path 3). This pair of water samples was selected specifically because it was one of the few pairs that showed a relatively large discrepancy between previous mole-balance approaches and the mole-balance approach of PHREEQC, which includes uncertainties; results for most sample pairs were not significantly different between the two approaches. In addition, this pair of samples was selected because it was modeled in detail in Plummer and others (1990) to determine the sensitivity of mole-balance results to various model assumptions and because it was used as an example in the NETPATH manual (Plummer and others, 1994, Example 6). Results of PHREEQC calculations are compared to NETPATH calculations. This example is also discussed in Parkhurst (1997).

Water Compositions and Reactants

The initial water for mole-balance modeling (solution 1, table 52) is the water identified as the recharge water for flow path 3 (Plummer and others, 1990). This calcium magnesium bicarbonate water is typical of recharge water in a terrain containing calcite and dolomite. The final water (solution 2, table 52) is a sodium calcium sulfate water (with significant chloride concentration) (Plummer and others, 1990, "Mysse Flowing Well"). This water has a charge imbalance of +3.24 meq/kgw. The final water also contains measurable sulfide. An uncertainty limit of 5 percent was assigned to all chemical data, except iron, for the initial water and final water. The 5-percent uncertainty limit was chosen for the initial water because of spatial uncertainty in the location of a recharge water

Table 52.--Analytical data for solutions used in example 18

[Charge balance is milliequivalents per kilogram of water. All other data are in millimoles per kilogram of water, except pH, $\delta^{13}C$, $\delta^{34}S$, and ^{14}C. Fe(2), ferrous iron. TDIC, total dissolved inorganic carbon. $\delta^{13}C$ is carbon-13 composition of TDIC in permil relative to PDB (Pee Dee Belemnite standard). $\delta^{34}S(6)$ is sulfur-34 composition of sulfate in permil relative to CDT (Cañon Diablo Troilite standard). $\delta^{34}S(-2)$ is sulfur-34 composition of total sulfide in permil relative to CDT. ^{14}C is carbon-14 composition in percent modern carbon. ± , indicates the uncertainty limit assigned in inverse modeling. Uncertainty limit for pH was 0.1, uncertainty limit for all other data was 5 percent of value, except iron, which was 100 percent. --, not measured]

Analyte	Solution 1	Solution 2
Temperature, °C	9.9	63.0
pH	7.55	6.61
Ca	1.20	11.28
Mg	1.01	4.54
Na	.02	31.89
K	.02	2.54
Fe(2)	.001	.0004
TDIC	4.30	6.87
SO_4	.16	19.86
H_2S	0	.26
Cl	.02	17.85
$\delta^{13}C$	-7.0±1.4 .	-2.3±0.2
$\delta^{34}S(6)$	9.7±0.9	16.3±1.5
$\delta^{34}S(-2)$	--	-22.1±7.0
^{14}C	52.3	.8
Charge balance	+0.11	+3.24

that is on the same flow path as the final water, and for the final water because it was near the minimum uncertainty limit necessary to obtain charge balance. Iron was assigned an uncertainty limit of 100 percent because of the small concentrations. An uncertainty limit of 0.1 unit was assigned to pH, which is a conservative estimate because of the potential for CO_2 degassing at this sampling site (L.N. Plummer, U.S.Geological Survey, written commun., 1996). $\delta^{13}C$ values increase from the initial water to the final water (-7.0 permil to -2.3 permil), as do $\delta^{34}S$ values (9.7 permil to 16.3 permil). Uncertainty limits for isotopic values of the initial solution were set to one-half the range in isotopic composition in the four recharge waters from flow paths 3 and 4 (Plummer and others, 1990) (table 52). Similarly, uncertainty limits for isotopic values of the final water were set to one-half the range in isotopic composition in the samples from the distal end of flowpath 3 (Plummer and others, 1990) (table 52).

Reactants considered by Plummer and others (1990) were dolomite, calcite, anhydrite, organic matter (CH_2O), goethite, pyrite, Ca/Na_2 cation exchange, halite, sylvite, and CO_2 gas. In their sensitivity calculations, Mg/Na_2 cation exchange and methane were considered as potential reactants. The aquifer was considered to be a

closed system with respect to CO_2, that is, no CO_2 is expected to be gained from or lost to a gas phase, and methane gain or loss was considered to be unlikely (Plummer and others, 1990). Therefore, CO_2 gas and methane were not included as reactants in the PHREEQC mole-balance modeling. (CO_2 gas was included in the NETPATH modeling but mole transfers were reduced to zero by adjusting the $\delta^{34}S$ of anhydrite.) The uncertainty limits for the isotopic compositions of dissolving phases were taken from data presented in Plummer and others (1990) with slight modifications as follows: $\delta^{13}C$ of dolomite, 1 to 5 permil; $\delta^{13}C$ of organic carbon, -30 to -20 permil; $\delta^{34}S$ of anhydrite, 11.5 to 15.5 permil. The $\delta^{13}C$ of precipitating calcite depends on the isotopic evolution of the solution and is affected by isotopic fractionation. The fractionation equations are not included in PHREEQC, so it is necessary to assume a compositional range of calcite that represents the average isotopic composition of the precipitating calcite. The average isotopic composition of precipitating calcite from NETPATH calculations was about -1.5 permil (Plummer and others, 1994) and an uncertainty limit of 1.0 permil was selected to account for uncertainties in fractionation factors. All carbon-14 modeling was done with NETPATH using mole transfers from PHREEQC models. The $\delta^{34}S$ of precipitating pyrite was estimated to be -22 permil (Plummer and others, 1990) with an uncertainty limit of 2 permil; sensitivity analysis indicated that the isotopic value for the precipitating pyrite had little effect on mole transfers. The input data set for PHREEQC is shown in table 53. Note that the log K values for sylvite, CH_2O, and the $Ca_{0.75}Mg_{0.25}/Na_2$ exchange reaction are set to zero in the **PHASES** and **EXCHANGE_SPECIES** data blocks. The stoichiometry of each of these reactants is correct, which is all that is needed for mole-balance modeling; however, any saturation indices or forward modeling using these reactions would be incorrect because the log K values have not been properly defined.

Table 53.--Input data set for example 18

```
TITLE Example 18.--Inverse modeling of Madison aquifer
SOLUTION 1 Recharge number 3
        units   mmol/kgw
        temp    9.9
        pe      0.
        pH      7.55
        Ca      1.2
        Mg      1.01
        Na      0.02
        K       0.02
        Fe(2)   0.001
        Cl      0.02
        S(6)    0.16
        S(-2)   0
        C(4)    4.30
        -i      13C     -7.0    1.4
        -i      34S     9.7     0.9
SOLUTION 2 Mysse
        units   mmol/kgw
        temp    63.
        pH      6.61
        pe      0.
        redox   S(6)/S(-2)
        Ca      11.28
        Mg      4.54
        Na      31.89
```

```
        K       2.54
        Fe(2)   0.0004
        Cl      17.85
        S(6)    19.86
        S(-2)   0.26
        C(4)    6.87
        -i      13C     -2.3    0.2
        -i      34S(6)  16.3    1.5
        -i      34S(-2) -22.1   7
INVERSE_MODELING 1
        -solutions 1 2
        -uncertainty 0.05
        -range
        -isotopes
                13C
                34S
        -balances
                Fe(2)   1.0
                ph      0.1
        -phases
                Dolomite        dis     13C     3.0     2
                Calcite         pre     13C     -1.5    1
                Anhydrite       dis     34S     13.5    2
                CH2O            dis     13C     -25.0   5
                Goethite
                Pyrite          pre     34S     -22.    2
                CaX2            pre
                Ca.75Mg.25X2    pre
                MgX2            pre
                NaX
                Halite
                Sylvite
PHASES
   Sylvite
        KCl = K+ + Cl-
        -log_k  0.0
   CH2O
        CH2O + H2O = CO2 + 4H+ + 4e-
        -log_k  0.0
EXCHANGE_SPECIES
        0.75Ca+2 + 0.25Mg+2 + 2X- = Ca.75Mg.25X2
        log_k   0.0
END
```

Mole-balance calculations included equations for all elements in the reactive phases (listed under the identifier **-phases**) and an equation for $\delta^{34}S$. NETPATH calculations included isotopic fractionation equations to calculate the $\delta^{13}C$ of the final water, whereas PHREEQC calculations included a mole-balance equation on $\delta^{13}C$. The adjusted concentrations (original data plus calculated δ's) from the PHREEQC results were rerun with NETPATH to obtain carbon-14 ages and to consider the fractionation effects of calcite precipitation. One NETPATH calculation used the charge-balancing option to identify the effects of charge-balance errors. The charge-balance option of NETPATH adjusts the concentrations of all cationic elements by a fraction, f, and of all anionic elements by a fraction $\frac{1}{f}$ to

achieve charge balance for the solution. (The charge-balance option of NETPATH was improved in version 2.13 to produce exact charge balance; previous versions produced only approximate charge balance.)

For all NETPATH calculations (including calculations that used PHREEQC-adjusted concentrations), carbon dioxide was included as a potentially reactive phase, but the $\delta^{34}S$ of anhydrite was adjusted to produce zero mole transfer of carbon dioxide. The $\delta^{13}C$ of dolomite and organic matter were adjusted within their uncertainty limits to reproduce the $\delta^{13}C$ of the final solution as nearly as possible.

Madison Aquifer Results and Discussion

The predominant reactions determined by mole-balance modeling are dedolomitization, ion exchange, halite dissolution, and sulfate reduction, as listed in table 54 for various modeling options discussed next. The driving force for dedolomitization is dissolution of anhydrite (about 20 mmol/kgw, table 54), which causes calcite precipitation and dolomite dissolution. Some of the calcium from anhydrite dissolution and (or) magnesium from dolomite dissolution is taken up by ion-exchange sites, which release sodium to solution. About 15 mmol/kgw of halite dissolves. Sulfate and iron oxyhydroxide reduction by organic matter leads to precipitation of pyrite.

Plummer and others (1990) realized that the stoichiometry of the exchange reaction was not well defined and considered two variations on these reactions in the sensitivity analysis of the mole-balance model. Pure Ca/Na_2 exchange and pure Mg/Na_2 exchange were considered as potential reactants (NETPATH *A* and *B*, table 54). When PHREEQC was run with these two reactants, a model was found with Mg/Na_2 (PHREEQC *B*), but no model was found with pure Ca/Na_2 exchange. This difference between NETPATH and PHREEQC results is attributed to the charge imbalance of the solutions. Solution 2 (table 52) has a charge imbalance of 3.24 meq/kgw, which is more than 3 percent relative to the sum of cation and anion equivalents. This is not an exceptionally large percentage error, but the absolute magnitude in milliequivalents is large relative to some of the mole transfers of the mole-balance models. When the charge-balance constraint is included, by using the revised mole-balance equations in PHREEQC, with pure Ca/Na_2 exchange as the only exchange reaction, it is not possible simultaneously to attain mole balance on elements and isotopes, produce charge balance for each solution, and keep uncertainty terms within the specified uncertainty limits. The exchange reaction with the largest calcium component for which a model could be found was about $Ca_{0.75}Mg_{0.25}/Na_2$ (PHREEQC *C*). This exchange reaction was then used in NETPATH to find NETPATH *C*. NETPATH *C'* was calculated by using the charge-balance option of NETPATH with all phases and constraints the same as in NETPATH *C*.

One consistent difference between the NETPATH models without the charge-balance option (NETPATH *A*, *B*, and *C*) and the PHREEQC models is that the amount of organic-matter oxidation and the mole transfers of goethite and pyrite are larger in the PHREEQC models. These differences are attributed to the effects of charge balance on the mole transfers. It has been noted that charge-balance errors frequently manifest themselves as erroneous mole transfers of single component reactants, such as carbon dioxide or organic matter (Plummer and others, 1994). Except for differences in mole transfers in organic matter, goethite, and pyrite, the Mg/Na_2 models are similar (NETPATH *B* and PHREEQC *B*). However, both models imply a negative carbon-14 age which is impossible, as noted by Plummer and others (1990).

The PHREEQC model most similar to the pure Ca/Na_2 exchange model (NETPATH *A*) is the $Ca_{0.75}Mg_{0.25}/Na_2$ model (PHREEQC *C*). This model has larger mole transfers of carbonate minerals and organic matter than the Ca/Na_2 model, which decreases the reaction-adjusted carbon-14 activity and produces a younger ground-water

Table 54.--Mole-balance results for the Madison aquifer example

[Results are in millimoles per kilogram of water, unless otherwise noted. ^{14}C is carbon-14 in percent modern carbon (pmc), $\delta\ ^{13}C$ is carbon-13 in permil PDB, $\delta\ ^{34}S$ is sulfur-34 in permil CDT. CH_2O represents organic matter. Positive numbers for mineral mass transfer indicate dissolution; negative numbers indicate precipitation. For exchange reactions, positive numbers indicate a decrease in calcium and (or) magnesium and an increase in sodium in solution. --, reactant not included in model. $\delta\ ^{34}S$ of pyrite was approximately -22 permil in all models. For comparison to calculated isotopic values: measured $\delta\ ^{13}C$, -2.3; measured $\delta\ ^{34}S$(total) (sulfate plus sulfide), 15.8; measured ^{14}C, 0.8 pmc]

Result	Ca/Na_2	Mg/Na_2		$Ca_{0.75}Mg_{0.25}/Na$		
	NETPATH *A*	NETPATH *B*	PHREEQC *B*	NETPATH *C*	NETPATH *C'* Charge balanced	PHREEQC *C*
Ca/Na_2 exchange	8.3	--	--	--	--	--
$Ca_{0.75}Mg_{0.25}/Na_2$ exchange	--	--	--	8.3	7.6	7.7
Mg/Na_2 exchange	--	8.3	7.7	--	--	--
Dolomite [$CaMg(CO_3)_2$]	3.5	11.8	11.2	5.6	5.3	5.4
Calcite ($CaCO_3$)	-5.3	-21.8	-23.9	-9.4	-12.3	-12.1
Anhydrite ($CaSO_4$)	20.1	20.1	22.9	20.1	22.5	22.5
CH_2O	.8	.8	4.1	.8	4.3	3.5
Goethite (FeOOH)	.1	.1	1.0	.1	1.0	.8
Pyrite (FeS_2)	-.1	-.1	-1.0	-.1	-1.0	-.8
Halite (NaCl)	15.3	15.3	15.3	15.3	15.8	15.3
Sylvite (KCl)	2.5	2.5	2.5	2.5	2.5	2.5
Carbon dioxide (CO_2)	.0	.0	--	.0	.0	--
^{14}C, reaction adjusted	12.5	.6	.4	5.9	3.8	3.8
Apparent age (years)	22,700	-2,200	-5,400	16,500	13,000	12,900
$\delta\ ^{34}S$, Anhydrite	15.6	15.6	12.8	15.6	12.5	13.4
$\delta\ ^{13}C$, Dolomite	3.6	1.0	3.0	1.9	5.0	5.0
$\delta\ ^{13}C$, CH_2O	-25.0	-30.0	-21.4	-25.0	-20.0	-20.0
Calculated $\delta\ ^{13}C$, final water	-2.3	-2.2	-3.0	-2.3	-4.3	-3.3
Calculated $\delta\ ^{34}S$, final water	15.8	15.8	16.1	15.8	15.9	16.0

age, 12,900 (PHREEQC *C*) compared to 22,700 (NETPATH *A*). This large change in the calculated age can be attributed to differences in the reactions involving carbon. Two effects can be noted, the change in the exchange reaction and the adjustments for charge-balance errors. The effect of the change in exchange reaction can be estimated from the differences between NETPATH *A*, which contains pure Ca/Na_2 exchange and NETPATH *C*, which contains $Ca_{0.75}Mg_{0.25}/Na_2$ exchange, but neither model includes corrections for charge imbalances in the solution compositions. The increase in Mg in the exchange reaction causes larger mole transfers of calcite and dolomite and

decreases the calculated age from 22,700 to 16,500 years. The effects of charge-balance errors are estimated by the differences between NETPATH *C* and *C'*, which differ only in that the NETPATH charge-balance option was used in NETPATH *C'*. Charge balancing the solutions produces larger mole transfers of organic matter and calcite and decreases the calculated age from 16,500 to 13,000 years. The mole transfers and calculated age for NETPATH *C'* are similar to PHREEQC *C*, but differ slightly because the uncertainty terms in the PHREEQC model have been calculated to achieve not only charge balance but also to reproduce as closely as possible the observed $\delta\ ^{13}C$ of the final solution.

One advantage of the revised mole-balance formulation in PHREEQC is that much of the sensitivity analysis that was formerly accomplished by setting up and running multiple models can now be done by including uncertainty limits for all chemical and isotopic data simultaneously. For example, one run of the revised mole-balance formulation determines that no pure Ca/Na_2 model can be found even if any or all of the chemical data were adjusted by as much as plus or minus 10 percent. This kind of information would be very difficult and time consuming to establish with previous mole-balance formulations. Another improvement is the explicit inclusion of charge-balance constraints. In this example, including the charge-balance constraint requires a change in the exchange reaction and adjustments to solution composition, which have the combined effect of lowering the estimated maximum age of the ground water by about 10,000 years. If Mg/Na_2 exchange is the sole exchange reaction, the age would be modern. Thus, the estimated range in age is large, 0 to 13,000 years. However, because the calcium to magnesium ratio in solution is approximately 2.5:1 and the cation-exchange constants for calcium and magnesium are approximately equal (Appelo and Postma, 1993), the combined exchange reaction with a dominance for calcium is more plausible, which gives more credence to the older age. Furthermore, comparisons with other carbon-14 ages in the aquifer and with ground-water flow-model ages also indicate that the older end of the age range is more reasonable.

REFERENCES CITED

Aagaard, P. and Helgeson, H.C., 1982. Thermodynamic and kinetic constraints on reaction rates among minerals and aqueous solutions-I. theoretical considerations. American Journal of Science, v. 282, p. 237-285.

Allison, J.D., Brown, D.S., and Novo-Gradac, K.J., 1990, MINTEQA2/PRODEFA2--A geochemical assessment model for environmental systems--version 3.0 user's manual: Environmental Research Laboratory, Office of Research and Development, U.S. Environmental Protection Agency, Athens, Georgia, 106 p.

Appelo, C.A.J., 1994a, Cation and proton exchange, pH variations, and carbonate reactions in a freshening aquifer: Water Resources Research, v. 30, p. 2793-2805.

Appelo, C.A.J., 1994b. Some calculations on multicomponent transport with cation exchange in aquifers: Ground Water, v. 32, p. 968-975.

Appelo, C.A.J., Beekman, H.E. and Oosterbaan, A.W.A., 1984, Hydrochemistry of springs from dolomite reefs in the southern Alps of Northern Italy: International Association of Hydrology, Scientific Publication 150, p. 125-138.

Appelo, C.A.J., and Postma, D., 1999, A consistent model for surface complexation on birnessite (δ-MnO_2) and its application to a column experiment, Geochimica et Cosmochimica Acta, v.63, p. 3039-3048.

Appelo, C.A.J., and Postma, D., 1993, Geochemistry, groundwater and pollution: Rotterdam, A.A. Balkema, 536 p.

Ball, J.W. and Nordstrom, D.K., 1991, WATEQ4F—User's manual with revised thermodynamic data base and test cases for calculating speciation of major, trace and redox elements in natural waters: U.S. Geological Survey Open-File Report 90–129, 185 p.

Barrodale, I., and Roberts, F.D.K., 1980, L1 solution to linear equations subject to linear equality and inequality constraints: Association for Computing Machinery, Transactions on Mathematical Software, v. 6, p. 231-235.

Borkovec, Michal, and Westall, John, 1983, Solution of the \Poisson-Boltzmann equation for surface excesses of ions in the diffuse layer at the oxide-electrolyte interface: Journal of Electroanalytical Chemistry, v. 150, p. 325-337.

Carpenter, A.B., 1978, Origin and chemical evolution of brines in sedimentary basins: Thirteenth Annual Forum on the Geology of Industrial Minerals, eds. Johnson, K.S. and Russell, J.A., Oklahoma Geological Survey Circular 79, p. 60-77.

Cash, J.R. and Karp, A.H., 1990, A Variable Order Runge-Kutta Method for Initial Value Problems with Rapidly Varying Right-Hand Sides: Transactions on Mathematical Software, v. 16, no. 3, p. 201-222.

Charlton, S.R., Macklin, C.L., and Parkhurst, D.L., 1997, PHREEQCI--A graphical user interface for the geochemical computer program PHREEQC: U.S. Geological Survey Water-Resources Investigations Report 97-4222, 9 p.

Crank, J., 1975, The mathematics of diffusion, 2nd ed., Oxford Press, 414 p.

Davis, J.A., and Kent, D.B, 1990, Surface complexation modeling in aqueous geochemistry, *in* Hochella, M.F., and White, A.F., eds., Mineral-Water Interface Geochemistry: Washington D.C., Mineralogical Society of America, Reviews in Mineralogy, v. 23, Chapt. 5, p.177-260.

Delany, J.M., Puigdomenech, I. and Wolery, T.J., 1986. Precipitation kinetics option for the EQ6 geochemical reaction path code. Lawrence Livermore National Laboratory, University of California, Livermore, 44 p.

De Marsily, G., 1986. Quantitative hydrogeology: Orlando, Academic Press, 440 p.

Dzombak, D.A., and Morel, F.M.M., 1990, Surface complexation modeling—Hydrous ferric oxide: New York, John Wiley, 393 p.

Fehlberg, E., 1969. Klassische Runge-Kutta-Formeln fünfter und siebenter Ordnung mit Schrittweiten-Kontrolle: Computing, v. 4, p. 93-106.

Gaines, G.L., and Thomas, H.C., 1953, Adsorption studies on clay minerals. II. A formulation of the thermodynamics of exchange adsorption: Journal of Chemical Physics, v. 21, p. 714-718.

Garrels, R.M., and Christ, C.L., 1965, Solutions, minerals, and equilibria: New York, Harper and Row, 450 p.

Garrels, R.M, and Mackenzie, F.T., 1967, Origin of the chemical composition of springs and lakes, *in* Equilibrium concepts in natural water systems: American Chemical Society, Advances in Chemistry Series no. 67, p. 222-242.

Glynn, P.D., 1990, Modeling solid-solution reactions in low-temperature aqueous systems, *in* Bassett, R.L. and Melchior, D. eds., Chemical modeling in aqueous systems II: Washington D.C., American Chemical Society Symposium Series 416, Chapt. 6, p. 74-86.

Glynn, P.D., 1991, MBSSAS: A code for the computation of Margules parameters and equilibrium relations in binary solid-solution aqueous-solution systems: Computers and Geosciences, v. 17, no. 7, p. 907-966.

Glynn, P.D., and Parkhurst, D.L., 1992, Modeling non-ideal solid-solution aqueous-solution reactions in mass-transfer computer codes, *in* Kharaka, Y.K., and Maest, A.S. eds., Proceedings of the 7th International Symposium on Water-Rock Interaction: Park City, Utah, July 13-18, 1992, Balkema, Rotterdam, p. 175-179.

Glynn, P.D., and Reardon, E.J., 1990, Solid-solution aqueous-solution equilibria--Thermodynamic theory and representation: American Journal of Science, v. 290, p. 164-201.

Glynn, P.D., Reardon, E.J., Plummer, L.N., and Busenberg, Eurybiades, 1990, Reaction paths and equilibrium end-points in solid solution aqueous-solution systems: Geochimica et Cosmochimica Acta, v. 54, p. 267-282.

Harvie, C.E., Moller, N., and Weare, J.H., 1984, The prediction of mineral solubilities in natural waters: The Na-K-Mg-Ca-H-Cl-SO_4-OH-HCO_3-CO_3-CO_2-H_2O system to high ionic strengths at 25°C: Geochimica et Cosmochimica Acta, v. 48, p. 723-751.

Harvie, C.E., and Weare, J.H., 1980, The prediction of mineral solubilities in natural waters: The Na-K-Mg-Ca-Cl-SO_4-H_2O system from zero to high concentration at 25°C: Geochimica et Cosmochimica Acta, v. 44, p. 981-997.

Helgeson, H.C., Brown, T.H., Nigrini, A., and Jones, T.A., 1970, Calculation of mass transfer in geochemical processes involving aqueous solutions: Geochimica et Cosmochimica Acta, v. 34, p. 569-592.

Helgeson, H.C., Garrels, R.M., and Mackenzie, F.T., 1969, Evaluation of irreversible reactions in geochemical processes involving minerals and aqueous solutions - II. Applications: Geochimica et Cosmochimica Acta, v. 33, p. 455-481.

Langmuir, Donald, 1997, Aqueous environmental geochemistry: Upper Saddle River, New Jersey, Prentice Hall, 600 p.

Lapidus, L., and Amundson, N.R., 1952, Mathematics of adsorption in beds. VI. The effect of longitudinal diffusion in ion-exchange and chromatographic columns: Journal of Physical Chemistry, v. 56, p. 984-988.

Lindstrom, F.T., Haque, R., Freed, V.H. and Boersma, L., 1967, Theory on the movement of some herbicides in soils--Linear diffusion and convection of chemicals: Environmental Science and Technology, v. 1, p. 561-565.

Mosier, E.L., Papp, C.S.E., Motooka, J.M., Kennedy, K.R., and Riddle, G.O., 1991, Sequential extraction analyses of drill core samples, Central Oklahoma Aquifer: U.S. Geological Survey Open-File Report 91-347, 42 p.

Nordstrom, D.K., Plummer, L.N., Langmuir, Donald, Busenberg, Eurybiades, May, H.M., Jones, B.F., and Parkhurst, D.L., 1990, Revised chemical equilibrium data for major water-mineral reactions and their limitations, *in* Bassett, R.L. and Melchior, D. eds., Chemical modeling in aqueous systems II: Washington D.C., American Chemical Society Symposium Series 416, Chapt. 31, p. 398–413.

Nordstrom, D.K., Plummer, L.N., Wigley, T.M.L., Wolery, T.J., Ball, J.W., Jenne, E.A., Bassett, R.L., Crerar, D.A., Florence, T.M., Fritz, B., Hoffman, M., Holdren, G.R., Jr., Lafon, G.M., Mattigod, S.V., McDuff, R.E., Morel, F., Reddy, M.M., Sposito, G., and Thrailkill, J., 1979, A comparison of computerized chemical models for equilibrium calculations in aqueous systems: *in* Chemical Modeling in aqueous systems, speciation, sorption, solubility, and kinetics, Jenne, E.A., ed., Series 93, American Chemical Society, p. 857-892.

Parkhurst, D.L., 1995, User's guide to PHREEQC--A computer program for speciation, reaction-path, advective-transport, and inverse geochemical calculations: U.S. Geological Survey Water-Resources Investigations Report 95-4227, 143 p.

Parkhurst, D.L., 1997, Geochemical mole-balance modeling with uncertain data: Water Resources Research, v. 33, no. 8, p. 1957-1970.

Parkhurst, D.L., Christenson, Scott, and Breit, G.N., 1996, Ground-water-quality assessment of the Central Oklahoma Aquifer--Geochemical and geohydrologic investigations: U. S. Geological Survey Water-Supply Paper 2357-C, 101 p.

Parkhurst, D. L., Plummer, L.N., and Thorstenson, D.C., 1982, BALANCE--A computer program for calculating mass transfer for geochemical reactions in ground water: U.S. Geological Survey Water-Resources Investigations Report 82-14, 29 p.

Parkhurst, D.L., Thorstenson, D.C., and Plummer, L.N., 1980, PHREEQE--A computer program for geochemical calculations: U.S. Geological Survey Water-Resources Investigations Report 80-96, 195 p. (Revised and reprinted August, 1990.)

Pitzer, K.S., 1979, Theory--Ion interaction approach: *in* R.M. Pytkowicz, ed., Activity Coefficients in Electrolyte Solutions, v. 1, CRC Press, Inc., Boca Raton, Florida, p. 157-208.

Plummer, L.N., 1984, Geochemical modeling: A comparison of forward and inverse methods, *in* Hitchon, B., and Wallick, E.I., eds., First Canadian/American Conference on Hydrogeology, Practical Applications of Ground Water Geochemistry: Worthington, Ohio, National Water Well Association, p. 149-177.

Plummer, L.N. and Back, W.W. 1980, The mass balance approach--Application to interpreting the chemical evolution of hydrologic systems: American Journal of Science, v. 280, p. 130-142.

Plummer, L.N., Busby, J.F., Lee, R.W., and Hanshaw, B.B., 1990, Geochemical modeling of the Madison aquifer in parts of Montana, Wyoming, and South Dakota: Water Resources Research, v. 26, p. 1981-2014.

Plummer, L.N., and Busenberg, Eurybiades, 1987, Thermodynamics of aragonite-strontianite solid solutions--Results from stoichiometric dissolution at 25 and 76 C: Geochimica et Cosmochimica Acta, v. 51, p. 1393-1411.

Plummer, L.N., Parkhurst, D.L., Fleming, G.W., and Dunkle, S.A., 1988, A computer program incorporating Pitzer's equations for calculation of geochemical reactions in brines: U.S. Geological Survey Water-Resources Investigations Report 88–4153, 310 p.

Plummer, L.N., Parkhurst, D.L., and Thorstenson, D.C., 1983, Development of reaction models for groundwater systems: Geochimica et Cosmochimica Acta, v. 47, p. 665-685.

Plummer, L.N., Prestemon, E.C., and Parkhurst, D.L., 1991, An interactive code (NETPATH) for modeling net geochemical reactions along a flow path: U.S. Geological Survey Water-Resources Investigations Report 91–4087, 227 p.

Plummer, L.N., Prestemon, E.C., and Parkhurst, D.L., 1994, An interactive code (NETPATH) for modeling net geochemical reactions along a flow path, version 2.0: U.S. Geological Survey Water-Resources Investigations Report 94-4169, 130 p.

Plummer, L.N., Wigley, T.M.L., and Parkhurst, D.L., 1978, The kinetics of calcite dissolution in CO_2-water systems at 5 to 60 C and 0.0 to 1.0 atm CO2: American Journal of Science, v. 278, p. 179-216.

Press, W.H., Teukolsky, S.A., Vetterling, W.T., and Flannery, B.P., 1992, Numerical Recipes in C--The Art of Scientific Computing, second edition: Cambridge University Press, 994 p.

Robie, R.A., Hemingway, B.S., and Fisher, J.R., 1978, Thermodynamic properties of minerals and related substances at 298.15 K and 1 bar (10^5 pascals) pressure and at higher temperatures: U.S. Geological Survey Bulletin 1452, 456 p.

Russell, E.W., 1973. Soil conditions and plant growth: Longman, London, 849 p.

Schecher, W.D., and McAvoy, D.C., 1991, MINEQL$^+$--A chemical equilibrium program for personal computers--User's manual version 2.1: Edgewater, Maryland, Environmental research Software.

Singer, P.C. and Stumm, W., 1970, Acid mine drainage--the rate limiting step: Science, v. 167, p. 1121-1123.

Sverdrup, H.U., 1990, The kinetics of base cation release due to chemical weathering: Lund University Press, Lund, 246 p.

Tebes-Steven, Caroline, and Valocchi, A.J., 1997, Reactive transport simulation with equilibrium speciation and kinetic biodegradation and adsorption/desorption reactions: A Workshop on Subsurface Reactive Transport Modeling, Pacific Northwest National Laboratory, Richland, Washington, October 29-November 1, 1997 (http://terrassa.pnl.gov:2080/~kash/workshop/bmark.htm).

Tebes-Stevens, C., Valocchi A.J., VanBriesen J.M., Rittmann B.E., 1998, Multicomponent transport with coupled geochemical and microbiological reactions--model description and example simulations: Journal of Hydrology, v. 209, p. 8-26.

Toride, N., Leij, F.J., and Van Genuchten, M.T., 1993, A comprehensive set of analytical solutions for non-equilibrium solute transport with first-order decay and zero-order production: Water Resources Research, v. 29, p. 2167-2182.

Toride, N., Leij, F.J., and Van Genuchten, M.T., 1995, The CXTFIT code for estimating transport parameters from laboratory or field tracer experiments, version 2.0: U.S. Salinity Laboratory Research Report 137, Riverside, California.

Truesdell, A.H., and Jones, B.F., 1974, WATEQ, A computer program for calculating chemical equilibria of natural waters: Journal of Research, U.S. Geological Survey, v. 2, p. 233-274.

Van Cappellen, P. and Wang, Y., 1996, Cycling of iron and manganese in surface sediments: American Journal of Science, v. 296, p. 197-243.

Van Genuchten, M.Th., 1985, A general approach for modeling solute transport in structured soils: IAH Memoirs, v. 17, p. 513-526.

Waite, T.D., Davis, J.A., Payne, T.E., Waychunas, G.A., and Xu, N., 1994, Uranium(VI) adsorption to ferrihydrite--Application of a surface complexation model: Geochimica et Cosmochimica Acta, v. 59, p. 5465-5478.

Williamson, M.A. and Rimstidt, J.D., 1994, The kinetics and electrochemical rate-determining step of aqueous pyrite oxidation: Geochimica et Cosmochimica Acta, v. 58, p. 5443-5454.

Wolery, T. J., 1979, Calculation of chemical equilibrium between aqueous solution and minerals--The EQ3/6 software package: Lawrence Livermore National Laboratory Report UCRL-52658, Livermore, CA.

Wolery, T. J., Jackson, K.J., Bourcier, W.L., Bruton, C.J., Viani, B.E., Knauss, K.G., and Delany, J.N., 1990, Current status of the EQ3/6 software package for geochemical modeling, *in* Melchior, D.C., and Bassett, R.L., eds., Chemical Modeling of Aqueous Systems II: Washington D. C., American Chemical Society Symposium Series 416, p. 104-116.

Yanenko, N., 1971, The method of fractional steps: Springer, New York.

Yeh, G.T., and Tripathi, V.S., 1989, A critical evaluation of recent developments in hydrogeochemical transport models of reactive multichemical components: Water Resources Research, v. 25, p. 93-108.

Attachment A--Listing of Notation

A Temperature dependent constant in the activity coefficient equation.

A_0 Initial surface area of kinetic reacting solid (m^2).

A_{ij} Interfacial area between cells i and j (m^2).

A_r Specific surface area of a surface that is related to a pure phase or kinetic reactant, m^2/mol.

A_s Specific surface area of surface s, m^2/g.

A_{surf} Surface area of a surface-complexation material, m^2

α Temperature dependent constant for diffuse layer surface model, 0.02931 $[(L/mol)^{1/2}\ C\ m^{-2}]$ at 25°C.

α Factor for mobile-immobile exchange, s^{-1}.

α_L Dispersivity (m).

α_p Mole transfer of phase p into (positive) or out of (negative) solution, mol.

α_q Mixing fraction for aqueous phase q.

α_r Aqueous transfer of an element between valence states, mol.

a_{Alk} Activity of the master species for alkalinity.

a_e Activity of the master species for exchanger e.

a_i^o Ion size parameter for aqueous species i for extended Debye-Hückel equation or simply a fitted parameter for WATEQ Debye-Hückel equation.

a_i Activity of aqueous species i.

a_{i_e} Activity of exchange species i_e.

a_{i_s} Activity of surface species i_s.

$a_{m'}$ Activity of an aqueous master species, but excluding a_{Alk}, a_{H^+}, a_{e^-}, and a_{H_2O}.

a_m Activity of a master species, including all aqueous, exchange, and surface master species.

$a_{p_{ss}}$ Activity of solid-solution component p_{ss}.

a_s Activity of the master species for surface s.

a_{Ψ_s} Master unknown for the surface potential of surface s, $a_{\Psi_s} = \frac{F\Psi_s}{2RT}$.

B Temperature dependent constant in the activity coefficient equation.

$b_{Alk,\,i}$ Number of equivalents of alkalinity per mole of aqueous species i.

$b_{e,\,i_e}$ Number of exchange sites of exchanger e occupied by exchange species i_e.

b_i Debye-Hückel fitting parameter for aqueous species i.

$b_{m,\,g}$ Stoichiometry of element m in gas component g.

$b_{m,\,i}$ Stoichiometry of element m in aqueous species i.

$b_{m',\,i}$ Stoichiometry of element m' in aqueous species i.

$b_{m,\,i_e}$ Stoichiometry of element m in exchange species i_e.

$b_{m,\,i_{(s_k)}}$ Stoichiometry of element m in surface species $i_{(s_k)}$.

$b_{m,\,p}$ Stoichiometry of element m in phase p.

$b_{m,\,p_{ss}}$ Stoichiometry of element m in solid-solution component p_{ss}.

$b_{s,\,i_{(s_k)}}$ Number of sites of surface type s_k occupied by surface species $i_{(s_k)}$.

$\Gamma_{i,\,s}$ Surface excess of aqueous species i for surface s, mol m^{-2}.

γ_i Activity coefficient of aqueous species i, kg H_2O/mol.

C Concentration in water, mol/kgw.

C_{im} Concentration in stagnant water, mol/kgw.

C_m Concentration in mobile water, mol/kgw.

c_i Concentration of aqueous species i, mol m^{-3}.

$c_{i,k}$ Stoichiometric coefficient of species i in the kinetic reaction k.

$c_{m,g}$ Stoichiometric coefficient of master species m in the dissolution reaction for gas component g.

$c_{m,i}$ Stoichiometric coefficient of master species m in the association reaction for aqueous species i.

c_{m,i_e} Stoichiometric coefficient of master species m in the association reaction for exchange species i_e.

c_{m,i_s} Stoichiometric coefficient of master species m in the association reaction for surface species i_s.

$c_{m,p}$ Stoichiometric coefficient of master species m in the dissolution reaction for phase p.

$c_{m,r}$ Stoichiometric coefficient of secondary master species m in redox reaction r.

$\delta_{m,q}$ Uncertainty term for the moles of an element or element valence state m in solution q calculated in inverse modeling, mol.

$\delta_{R^i_{m,q}}$ Uncertainty term for the isotopic ratio of isotope i for a valence state m in the aqueous solution q calculated in inverse modeling.

$\delta_{R^i_{e,p}}$ Uncertainty term for the isotopic ratio of isotope i for element e in the phase p, calculated in inverse modeling.

D_e Effective diffusion coefficient (m^2/s).

D_L Hydrodynamic dispersion coefficient (m^2/s).

E Number of exchangers.

e Index number for exchangers.

ε dielectric constant for water, 78.5, unitless.

ε_0 dielectric permittivity of a vacuum, 8.854×10^{-12} C V^{-1} m^{-1}.

F Faraday constant, 96,485 Coulomb/mol.

f_{Alk} Alkalinity balance equation.

f_e Mole-balance equation for exchanger e.

f_g Equation relating aqueous and gas-phase partial pressures for gas component, g.

f_H Mole-balance equation for hydrogen.

f_{H_2O} Equation for activity of water in an aqueous solution.

f_m Mole-balance equation for element or element valence state. exchanger, or surface, m.

$f_{m'}$ Mole-balance equation for element or element valence state m, excluding alkalinity, hydrogen, and oxygen and also excluding the charge balance equation.

f_O Mole-balance equation for oxygen.

$f_{P_{total}}$ Equation that sums the partial pressures of all gas components, as calculated from aqueous species.

f_p Saturation index equation for phase p.

$f_{p_{ss}}$ Saturation index equation for component p_{ss} in solid solution ss.

f_s Mole-balance equation for surface s.

f_z Charge-balance equation for aqueous solution.

$f_{z,s}$ Charge-balance equation for surface s, used in explicit diffuse layer calculation.

f_μ Equation for ionic strength in an aqueous solution.

f_{Ψ_s} Charge-potential equation for surface s, used when diffuse layer composition is not explicitly calculated.

g Index number for gas-phase components.

$g_{i,s}$ Ratio of concentration of aqueous species i in surface excess for surface s to concentration in the bulk solution.

θ_{im} Porosity of the stagnant (immobile) zone (a fraction of total volume, unitless).

θ_m Water filled porosity of the mobile part (a fraction of total volume, unitless).

θ Total porosity (unitless).

i Aqueous species index number.

i_e Exchange species index number for exchange site e.

$i_{(s_k)}$ Surface species index number for surface site type s_k.

J Total number of master unknowns for a calculation.

κ_e Effective thermal diffusion coefficient (m^2/s).

κ_t Heat conductivity of the aquifer, including pore water and solid ($kJ°C^{-1}m^{-1}s^{-1}$).

κ_L Thermal dispersion coefficient (m^2/s).

K_g Equilibrium constant for gas component g.

K_i Equilibrium constant for aqueous species i.

$K_{i_s}^{int}$ Intrinsic equilibrium constant for association reaction for surface species i_s.

K_p Equilibrium constant for phase p.

K_s Number of site types for surface s.

k Specific heat ($kJ°C^{-1}kg^{-1}$); k_w for water, k_s for solid.

μ Ionic strength.

M Total number of master species.

M_{aq} Total number of aqueous master species.

M_e Number of valence states of element e.

m Index number for master species.

m' Index number for aqueous master species, excluding H^+, e^-, H_2O, and the alkalinity master species.

m_{0k} Initial moles of kinetic reactant k.

m_i Molality of the aqueous species i, mol/kgw.

$m_{i,s}$ Surface excess of aqueous species i, mol/kgw.

m_k Moles of kinetic reactant k at a given time.

ν valence of a symmetric electrolyte.

N_{aq} Number of aqueous species.

N_e Number of exchange species for exchanger e.

N_g Number of gas components in the gas phase.

N_{gas} Total moles of gas in the gas phase.

N_p Number of phases in the phase assemblage.

N_s Number of surface species for surface s.

N_{ss} Number of components in solid solution ss.

n_g Moles of gas component g in the gas phase.

n_i Moles of aqueous species i in the system.

$n_{i,s}$ Moles of aqueous species i the diffuse layer of surface s.

n_{i_e} Moles of exchange species i_e in the system.

$n_{i_{(s_k)}}$ Moles of surface species $i_{(s_k)}$ in the system.

n_p Moles of phase p in the phase assemblage.

$n_{p_{ss}}$ Moles of solid-solution component p_{ss} in solid solution ss.

n_r Moles of a reactant, either a pure phase or a kinetically controlled reactant.

P Total number of reactants in inverse modeling.

P_g Partial pressure of gas component g, atm.

P_{total} Total pressure in the gas phase, atm.

p Index number for phases in phase assemblage.

p_{ss} Index number for components in solid solution *ss*.

Ψ_s Surface potential for surface s, V.

Q Number of aqueous solutions.

q Index number for an aqueous solution in a set of aqueous solutions.

q Concentration in the solid phase, expressed as mol/kgw in the pores.

ρ Density (kg/m^3); ρ_w for water, ρ_s for solid.

R Gas constant, kJ mol^{-1} K^{-1}.

R Retardation factor (unitless).

R_T Temperature retardation factor (unitless).

R Total number of aqueous redox reactions in inverse modeling.

$R^i_{e,p}$ Isotopic ratio of isotope i for element e in phase p.

R_{im} Retardation in the stagnant zone, unitless.

R_k Overall rate of reaction for substance k, mol/kgw/s.

R_m Retardation in the mobile zone, unitless.

$R^i_{m,q}$ Isotopic ratio of isotope i for valence state m in aqueous solution q.

r_k Specific rate of reaction for solid k, mol/m^2/s.

σ_s Surface-charge density for surface s, C/m^2.

S Number of surfaces.

s Index number for surfaces.

s_i Concentration of i in the solid phase, mol/kg solid.

S_s Mass of surface s, g.

SI_p Saturation index for phase p.

$SI_{p,target}$ Specified target saturation index for phase p.

SS Number of solid solutions in solid-solution assemblage.

ss Index number for solid solutions.

T Temperature, K.

T_{Alk} Total number of equivalents of alkalinity in solution.

T_e Total number of equivalents of exchange sites for exchanger e.

T_m Total quantity of m, an element, element valence, exchanger site, surface site, or alkalinity, mol or for alkalinity, eq.

$T_{m'}$ Total quantity of a dissolved element or element valence state excluding alkalinity, hydrogen, oxygen, and electrons, mol.

$T_{m,q}$ Total moles of an element, element valences, or alkalinity, m, in solution q, mol or for alkalinity, eq.

T_s Total number of equivalents of surface sites for surface s.

T_z Charge imbalance for the system during reaction and transport calculations, eq.

$T_{z,e}$ Charge imbalance for the exchanger e, eq.

$T_{z,q}$ Charge imbalance for the aqueous phase q, eq.

$T_{z,s}$ Charge imbalance for the surface s, eq.

t Time, s.

t_s Thickness of diffuse layer for surface s, m.

$u_{m,q}$ Uncertainty limit assigned to element m in solution q, mol.

v Pore water flow velocity (m/s).

V Amount of solution in kinetic reactions, kg H_2O.

V_{total} Total volume of a fixed-volume gas phase, L.

W_{aq} Mass of water in the aqueous phase, excluding any water in diffuse layer of surfaces, kg.

W_{bulk} Total mass of water in the system, includes aqueous phase and water in the diffuse layer of surfaces, kg.

W_s Mass of water in the diffuse layer of surface s, kg.

x Distance, m.

Z_q Charge imbalance in solution q in inverse modeling, eq.

z_i Charge on aqueous species i.

z_{i_e} Charge on exchange species i_e. (Normally equal to zero).

z_{i_s} Charge on surface species i_s.

$\tilde{z}_m$ Charge on aqueous master species minus alkalinity assigned to the master species.

Attachment B--Description of Database Files and Listing

Three database files are distributed with the program: *phreeqc.dat*, *wateq4f.dat*, and *minteq.dat*. Each of these database files contains **SOLUTION_MASTER_SPECIES**, **SOLUTION_SPECIES**, **PHASES**, **SURFACE_MASTER_SPECIES**, and **SURFACE_SPECIES** data blocks. *Phreeqc.dat* and *wateq4f.dat* also have **EXCHANGE_MASTER_SPECIES**, **EXCHANGE_SPECIES** and **RATES** data blocks.

The file named *phreeqc.dat* contains the thermodynamic data for aqueous species and gas and mineral phases that are essentially the same as those found in the latest release of the program PHREEQE (Parkhurst and others, 1980). Only minor modifications have been made to make the data consistent with the tabulations in Nordstrom and others (1990) and WATEQ4F (Ball and Nordstrom, 1991). The database file contains data for the following elements: aluminum, barium, boron, bromide, cadmium, calcium, carbon, chloride, copper, fluoride, hydrogen, iron, lead, lithium, magnesium, manganese, nitrogen, oxygen, phosphorous, potassium, silica, sodium, strontium, sulfur, and zinc. The thermodynamic data for cation exchange are taken from Appelo and Postma (1993, p. 160) and converted to log K, accounting for valence of the exchanging species. The thermodynamic data for surface species are taken from Dzombak and Morel (1990); acid base surface reactions are taken from table 5.7 and other cation and anion reactions are taken from tables in chapter 10. Preliminary rate expressions for K-feldspar (Sverdrup, 1990), albite (Sverdrup, 1990), calcite (modified from Plummer and others, 1978), pyrite (Williamson and Rimstidt, 1994), organic carbon ("Organic_c") (additive Monod kinetics for oxygen, nitrate, and sulfate), and pyrolusite (Postma, D. and Appelo, C.A.J., 2000, *Geochim. Cosmochim. Acta, in press*) are included from various sources. Examples of **KINETICS** data block for each of these expressions are included in the definitions in the **RATES** data block in *phreeqc.dat*.

The file named *wateq4f.dat* contains thermodynamic data for the aqueous species and gas and mineral phases that are essentially the same as WATEQ4F (Ball and Nordstrom, 1991). In addition to data for the elements in the database file, *phreeqc.dat*, the database file *wateq4f.dat* contains data for the elements: arsenic, cesium, iodine, nickel, rubidium, selenium, silver, and uranium. The WATEQ4F-derived database file also includes complexation

constants for two generalized organic ligands, fulvate and humate. Some additional gases are included; some carbonate reactions retain the chemical equations used in PHREEQE. Cation exchange data from Appelo and Postma (1993) as well as surface complexation reactions from Dzombak and Morel (1990) have been included. The rate expressions in *phreeqc.dat* are also included in *wateq4f.dat*.

The file named *minteq.dat* contains thermodynamic data for the aqueous species and gas and mineral phases that are derived from the database files of MINTEQA2 (Allison and others, 1990). The database file contains data for the following elements: aluminum, barium, boron, bromide, cadmium, calcium, carbon, chloride, copper, fluoride, hydrogen, iron, lead, lithium, magnesium, manganese, nitrogen, oxygen, phosphorous, potassium, silica, sodium, strontium, sulfur, and zinc. It also has data for the following organic ligands: benzoate, p-acetate, isophthalate, diethylamine, n-butylamine, methylamine, dimethylamine, tributylphosphate, hexylamine, ethylenediamine, n-propylamine, isopropylamine, trimethylamine, citrate, NTA, EDTA, propanoate, butanoate, isobutyrate, 2-methylpyridine, 3-methylpyridine, 4-methylpyridine, formate, isovalerate, valerate, acetate, tartrate, glycine, salicylate, glutamate, and phthalate.

A listing of the file, *phreeqc.dat* follows. In the interest of space, the other files are not included in this attachment, but are included with the program distribution.

Table 55.--Attachment B. *phreeqc.dat:* Database file derived from PHREEQE

```
SOLUTION_MASTER_SPECIES
#
#element  species        alk     gfw_formula     element_gfw
#
H         H+             -1.     H               1.008
H(0)      H2             0.0     H
H(1)      H+             -1.     0.0
E         e-             0.0     0.0             0.0
O         H2O            0.0     O               16.00
O(0)      O2             0.0     O
O(-2)     H2O            0.0     0.0
Ca        Ca+2           0.0     Ca              40.08
Mg        Mg+2           0.0     Mg              24.312
Na        Na+            0.0     Na              22.9898
K         K+             0.0     K               39.102
Fe        Fe+2           0.0     Fe              55.847
Fe(+2)    Fe+2           0.0     Fe
Fe(+3)    Fe+3           -2.0    Fe
Mn        Mn+2           0.0     Mn              54.938
Mn(+2)    Mn+2           0.0     Mn
Mn(+3)    Mn+3           0.0     Mn
Al        Al+3           0.0     Al              26.9815
Ba        Ba+2           0.0     Ba              137.34
Sr        Sr+2           0.0     Sr              87.62
Si        H4SiO4         0.0     SiO2            28.0843
Cl        Cl-            0.0     Cl              35.453
C         CO3-2          2.0     HCO3            12.0111
C(+4)     CO3-2          2.0     HCO3
C(-4)     CH4            0.0     CH4
Alkalinity CO3-2         1.0     Ca0.5(CO3)0.5   50.05
S         SO4-2          0.0     SO4             32.064
S(6)      SO4-2          0.0     SO4
S(-2)     HS-            1.0     S
N         NO3-           0.0     N               14.0067
N(+5)     NO3-           0.0     N
N(+3)     NO2-           0.0     N
N(0)      N2             0.0     N
N(-3)     NH4+           0.0     N
B         H3BO3          0.0     B               10.81
P         PO4-3          2.0     P               30.9738
F         F-             0.0     F               18.9984
Li        Li+            0.0     Li              6.939
Br        Br-            0.0     Br              79.904
Zn        Zn+2           0.0     Zn              65.37
Cd        Cd+2           0.0     Cd              112.4
Pb        Pb+2           0.0     Pb              207.19
Cu        Cu+2           0.0     Cu              63.546
Cu(+2)    Cu+2           0.0     Cu
Cu(+1)    Cu+1           0.0     Cu

SOLUTION_SPECIES

H+ = H+
        log_k           0.000
        -gamma          9.0000    0.0000

e- = e-
        log_k           0.000
```

```
H2O = H2O
        log_k           0.000

Ca+2 = Ca+2
        log_k           0.000
        -gamma    5.0000    0.1650

Mg+2 = Mg+2
        log_k           0.000
        -gamma    5.5000    0.2000

Na+ = Na+
        log_k           0.000
        -gamma    4.0000    0.0750

K+ = K+
        log_k           0.000
        -gamma    3.5000    0.0150

Fe+2 = Fe+2
        log_k           0.000
        -gamma    6.0000    0.0000

Mn+2 = Mn+2
        log_k           0.000
        -gamma    6.0000    0.0000

Al+3 = Al+3
        log_k           0.000
        -gamma    9.0000    0.0000

Ba+2 = Ba+2
        log_k           0.000
        -gamma    5.0000    0.0000

Sr+2 = Sr+2
        log_k           0.000
        -gamma    5.2600    0.1210

H4SiO4 = H4SiO4
        log_k           0.000

Cl- = Cl-
        log_k           0.000
        -gamma    3.5000    0.0150

CO3-2 = CO3-2
        log_k           0.000
        -gamma    5.4000    0.0000

SO4-2 = SO4-2
        log_k           0.000
        -gamma    5.0000   -0.0400

NO3- = NO3-
        log_k           0.000
        -gamma    3.0000    0.0000

H3BO3 = H3BO3
        log_k           0.000

PO4-3 = PO4-3
        log_k           0.000
        -gamma    4.0000    0.0000

F- = F-
        log_k           0.000
        -gamma    3.5000    0.0000

Li+ = Li+
        log_k           0.000
        -gamma    6.0000    0.0000

Br- = Br-
        log_k           0.000
        -gamma    3.0000    0.0000

Zn+2 = Zn+2
        log_k           0.000
        -gamma    5.0000    0.0000

Cd+2 = Cd+2
        log_k           0.000

Pb+2 = Pb+2
        log_k           0.000

Cu+2 = Cu+2
        log_k           0.000
        -gamma    6.0000    0.0000

H2O = OH- + H+
        log_k          -14.000
        delta_h 13.362  kcal
        -analytic       -283.971       -0.05069842  13323.0    102.24447      -1119669.0
        -gamma    3.5000    0.0000

2 H2O = O2 + 4 H+ + 4 e-
```

```
        log_k           -86.08
        delta_h 134.79 kcal

2 H+ + 2 e- = H2
        log_k           -3.15
        delta_h -1.759 kcal

CO3-2 + H+ = HCO3-
        log_k           10.329
        delta_h -3.561  kcal
        -analytic       107.8871        0.03252849  -5151.79    -38.92561       563713.9
        -gamma    5.4000    0.0000

CO3-2 + 2 H+ = CO2 + H2O
        log_k           16.681
        delta_h -5.738  kcal
        -analytic       464.1965        0.09344813  -26986.16    -165.75951      2248628.9

CO3-2 + 10 H+ + 8 e- = CH4 + 3 H2O
        log_k           41.071
        delta_h -61.039 kcal

SO4-2 + H+ = HSO4-
        log_k           1.988
        delta_h 3.85    kcal
        -analytic       -56.889         0.006473    2307.9       19.8858         0.0

HS- = S-2 + H+
        log_k           -12.918
        delta_h 12.1    kcal
        -gamma    5.0000    0.0000

SO4-2 + 9 H+ + 8 e- = HS- + 4 H2O
        log_k           33.65
        delta_h -60.140 kcal
        -gamma    3.5000    0.0000

HS- + H+ = H2S
        log_k           6.994
        delta_h -5.300  kcal
        -analytical  -11.17   0.02386  3279.0


NO3- + 2 H+ + 2 e- = NO2- + H2O
        log_k           28.570
        delta_h -43.760 kcal
        -gamma    3.0000    0.0000

2 NO3- + 12 H+ + 10 e- = N2 + 6 H2O
        log_k           207.080
        delta_h -312.130        kcal

NH4+ = NH3 + H+
        log_k           -9.252
        delta_h 12.48   kcal
        -analytic    0.6322    -0.001225     -2835.76

NO3- + 10 H+ + 8 e- = NH4+ + 3 H2O
        log_k           119.077
        delta_h -187.055        kcal
        -gamma    2.5000    0.0000

NH4+ + SO4-2 = NH4SO4-
        log_k           1.11

H3BO3 = H2BO3- + H+
        log_k           -9.240
        delta_h 3.224   kcal
#       -analytical 24.3919    0.012078   -1343.9    -13.2258

H3BO3 + F- = BF(OH)3-
        log_k           -0.400
        delta_h 1.850   kcal

H3BO3 + 2 F- + H+ = BF2(OH)2- + H2O
        log_k           7.63
        delta_h 1.618   kcal

H3BO3 + 2 H+ + 3 F- = BF3OH- + 2 H2O
        log_k           13.67
        delta_h -1.614  kcal

H3BO3 + 3 H+ + 4 F- = BF4- + 3 H2O
        log_k           20.28
        delta_h -1.846  kcal

PO4-3 + H+ = HPO4-2
        log_k           12.346
        delta_h -3.530  kcal
        -gamma    4.0000    0.0000

PO4-3 + 2 H+ = H2PO4-
        log_k           19.553
        delta_h -4.520  kcal
        -gamma    4.5000    0.0000

H+ + F- = HF
        log_k           3.18
        delta_h 3.18    kcal
```

```
        -analytic       -2.033          0.012645        429.01

H+ + 2 F- = HF2-
        log_k           3.760
        delta_h 4.550   kcal

Ca+2 + H2O = CaOH+ + H+
        log_k           -12.780

Ca+2 + CO3-2 = CaCO3
        log_k           3.224
        delta_h 3.545   kcal
        -analytic       -1228.732       -0.299440       35512.75        485.818

Ca+2 + CO3-2 + H+ = CaHCO3+
        log_k           11.435
        delta_h -0.871          kcal
        -analytic       1317.0071       0.34546894      -39916.84       -517.70761      563713.9
        -gamma      5.4000      0.0000

Ca+2 + SO4-2 = CaSO4
        log_k           2.300
        delta_h 1.650   kcal

Ca+2 + HSO4- = CaHSO4+
        log_k           1.08

Ca+2 + PO4-3 = CaPO4-
        log_k           6.459
        delta_h 3.100   kcal

Ca+2 + HPO4-2 = CaHPO4
        log_k           2.739
        delta_h 3.3 kcal

Ca+2 + H2PO4- = CaH2PO4+
        log_k           1.408
        delta_h 3.4 kcal

Ca+2 + F- = CaF+
        log_k           0.940
        delta_h 4.120   kcal

Mg+2 + H2O = MgOH+ + H+
        log_k           -11.440
        delta_h 15.952 kcal

Mg+2 + CO3-2 = MgCO3
        log_k           2.98
        delta_h 2.713   kcal
        -analytic       0.9910          0.00667

Mg+2 + H+ + CO3-2 = MgHCO3+
        log_k           11.399
        delta_h -2.771          kcal
        -analytic       48.6721         0.03252849      -2614.335       -18.00263       563713.9

Mg+2 + SO4-2 = MgSO4
        log_k           2.370
        delta_h 4.550   kcal

Mg+2 + PO4-3 = MgPO4-
        log_k           6.589
        delta_h 3.100   kcal

Mg+2 + HPO4-2 = MgHPO4
        log_k           2.87
        delta_h 3.3 kcal

Mg+2 + H2PO4- = MgH2PO4+
        log_k           1.513
        delta_h 3.4 kcal

Mg+2 + F- = MgF+
        log_k           1.820
        delta_h 3.200   kcal

Na+ + H2O = NaOH + H+
        log_k           -14.180

Na+ + CO3-2 = NaCO3-
        log_k           1.270
        delta_h 8.910   kcal

Na+ + HCO3- = NaHCO3
        log_k           -0.25

Na+ + SO4-2 = NaSO4-
        log_k           0.700
        delta_h 1.120   kcal

Na+ + HPO4-2 = NaHPO4-
        log_k           0.29

Na+ + F- = NaF
        log_k           -0.240

K+ + H2O = KOH + H+
        log_k           -14.460
```

```
K+ + SO4-2 = KSO4-
        log_k           0.850
        delta_h 2.250   kcal
        -analytical      3.106  0.0    -673.6


K+ + HPO4-2 = KHPO4-
        log_k           0.29

Fe+2 + H2O = FeOH+ + H+
        log_k           -9.500
        delta_h 13.200  kcal

Fe+2 + Cl- = FeCl+
        log_k           0.140

Fe+2 + CO3-2 = FeCO3
        log_k           4.380

Fe+2 + HCO3- = FeHCO3+
        log_k           2.0

Fe+2 + SO4-2 = FeSO4
        log_k           2.250
        delta_h 3.230   kcal

Fe+2 + HSO4- = FeHSO4+
        log_k           1.08

Fe+2 + 2HS- = Fe(HS)2
        log_k           8.95

Fe+2 + 3HS- = Fe(HS)3-
        log_k           10.987

Fe+2 + HPO4-2 = FeHPO4
        log_k           3.6

Fe+2 + H2PO4- = FeH2PO4+
        log_k           2.7

Fe+2 + F- = FeF+
        log_k           1.000

Fe+2 = Fe+3 + e-
        log_k           -13.020
        delta_h 9.680   kcal
        -gamma    9.0000    0.0000

Fe+3 + H2O = FeOH+2 + H+
        log_k           -2.19
        delta_h 10.4    kcal

Fe+3 + 2 H2O = Fe(OH)2+ + 2 H+
        log_k           -5.67
        delta_h 17.1    kcal

Fe+3 + 3 H2O = Fe(OH)3 + 3 H+
        log_k           -12.56
        delta_h 24.8    kcal

Fe+3 + 4 H2O = Fe(OH)4- + 4 H+
        log_k           -21.6
        delta_h 31.9    kcal

2 Fe+3 + 2 H2O = Fe2(OH)2+4 + 2 H+
        log_k           -2.95
        delta_h 13.5    kcal

3 Fe+3 + 4 H2O = Fe3(OH)4+5 + 4 H+
        log_k           -6.3
        delta_h 14.3    kcal

Fe+3 + Cl- = FeCl+2
        log_k           1.48
        delta_h 5.6     kcal

Fe+3 + 2 Cl- = FeCl2+
        log_k           2.13

Fe+3 + 3 Cl- = FeCl3
        log_k           1.13

Fe+3 + SO4-2 = FeSO4+
        log_k           4.04
        delta_h 3.91    kcal

Fe+3 + HSO4- = FeHSO4+2
        log_k           2.48

Fe+3 + 2 SO4-2 = Fe(SO4)2-
        log_k           5.38
        delta_h 4.60            kcal

Fe+3 + HPO4-2 = FeHPO4+
        log_k           5.43
        delta_h 5.76            kcal
```

```
Fe+3 + H2PO4- = FeH2PO4+2
        log_k           5.43

Fe+3 + F- = FeF+2
        log_k           6.2
        delta_h 2.7             kcal

Fe+3 + 2 F- = FeF2+
        log_k           10.8
        delta_h 4.8             kcal

Fe+3 + 3 F- = FeF3
        log_k           14.0
        delta_h 5.4             kcal

Mn+2 + H2O = MnOH+ + H+
        log_k           -10.590
        delta_h 14.400  kcal

Mn+2 + Cl- = MnCl+
        log_k           0.610

Mn+2 + 2 Cl- = MnCl2
        log_k           0.250

Mn+2 + 3 Cl- = MnCl3-
        log_k           -0.310

Mn+2 + CO3-2 = MnCO3
        log_k           4.900

Mn+2 + HCO3- = MnHCO3+
        log_k           1.95

Mn+2 + SO4-2 = MnSO4
        log_k           2.250
        delta_h 3.370   kcal

Mn+2 + 2 NO3- = Mn(NO3)2
        log_k           0.600
        delta_h -0.396  kcal

Mn+2 + F- = MnF+
        log_k           0.840

Mn+2 = Mn+3 + e-
        log_k           -25.510
        delta_h 25.800  kcal

Al+3 + H2O = AlOH+2 + H+
        log_k           -5.00
        delta_h 11.49   kcal
        -analytic       -38.253         0.0             -656.27         14.327

Al+3 + 2 H2O = Al(OH)2+ + 2 H+
        log_k           -10.1
        delta_h 26.90           kcal
        -analytic       88.500          0.0             -9391.6         -27.121

Al+3 + 3 H2O = Al(OH)3 + 3 H+
        log_k           -16.9
        delta_h 39.89           kcal
        -analytic       226.374         0.0             -18247.8        -73.597

Al+3 + 4 H2O = Al(OH)4- + 4 H+
        log_k           -22.7
        delta_h 42.30           kcal
        -analytic       51.578          0.0             -11168.9        -14.865

Al+3 + SO4-2 = AlSO4+
        log_k           3.5
        delta_h 2.29 kcal

Al+3 + 2SO4-2 = Al(SO4)2-
        log_k           5.0
        delta_h 3.11 kcal

Al+3 + HSO4- = AlHSO4+2
        log_k           0.46

Al+3 + F- = AlF+2
        log_k           7.000
        delta_h 1.060   kcal

Al+3 + 2 F- = AlF2+
        log_k           12.700
        delta_h 1.980   kcal

Al+3 + 3 F- = AlF3
        log_k           16.800
        delta_h 2.160   kcal

Al+3 + 4 F- = AlF4-
        log_k           19.400
        delta_h 2.200   kcal

Al+3 + 5 F- = AlF5-2
        log_k           20.600
        delta_h 1.840   kcal
```

```
Al+3 + 6 F- = AlF6-3
        log_k           20.600
        delta_h -1.670  kcal

H4SiO4 = H3SiO4- + H+
        log_k           -9.83
        delta_h 6.12            kcal
        -analytic       -302.3724     -0.050698      15669.69      108.18466    -1119669.0

H4SiO4 = H2SiO4-2 + 2 H+
        log_k           -23.0
        delta_h 17.6            kcal
        -analytic       -294.0184     -0.072650      11204.49      108.18466    -1119669.0

H4SiO4 + 4 H+ + 6 F- = SiF6-2 + 4 H2O
        log_k           30.180
        delta_h -16.260         kcal

Ba+2 + H2O = BaOH+ + H+
        log_k           -13.470

Ba+2 + CO3-2 = BaCO3
        log_k           2.71
        delta_h 3.55            kcal
        -analytic       0.113         0.008721

Ba+2 + HCO3- = BaHCO3+
        log_k           0.982
        delta_h 5.56 kcal
        -analytical     -3.0938         0.013669        0.0     0.0     0.0

Ba+2 + SO4-2 = BaSO4
        log_k           2.700

Sr+2 + H2O = SrOH+ + H+
        log_k           -13.290
        -gamma    5.0000    0.0000

Sr+2 + CO3-2 + H+ = SrHCO3+
        log_k           11.509
        delta_h 2.489           kcal
        -analytic       104.6391      0.04739549   -5151.79      -38.92561      563713.9
        -gamma    5.4000    0.0000

Sr+2 + CO3-2 = SrCO3
        log_k           2.81
        delta_h 5.22    kcal
        -analytic       -1.019         0.012826

Sr+2 + SO4-2 = SrSO4
        log_k           2.290
        delta_h 2.080   kcal

Li+ + H2O = LiOH + H+
        log_k           -13.640

Li+ + SO4-2 = LiSO4-
        log_k           0.640

Cu+2 + e- = Cu+
        log_k           2.720
        delta_h 1.650   kcal
        -gamma    2.5000    0.0000

Cu+2 + H2O = CuOH+ + H+
        log_k           -8.000
        -gamma    4.0000    0.0000

Cu+2 + 2 H2O = Cu(OH)2 + 2 H+
        log_k           -13.680

Cu+2 + 3 H2O = Cu(OH)3- + 3 H+
        log_k           -26.900

Cu+2 + 4 H2O = Cu(OH)4-2 + 4 H+
        log_k           -39.600

Cu+2 + SO4-2 = CuSO4
        log_k           2.310
        delta_h 1.220   kcal

Zn+2 + H2O = ZnOH+ + H+
        log_k           -8.96
        delta_h 13.4 kcal

Zn+2 + 2 H2O = Zn(OH)2 + 2 H+
        log_k           -16.900

Zn+2 + 3 H2O = Zn(OH)3- + 3 H+
        log_k           -28.400

Zn+2 + 4 H2O = Zn(OH)4-2 + 4 H+
        log_k           -41.200

Zn+2 + Cl- = ZnCl+
        log_k           0.43
        delta_h 7.79 kcal
```

```
Zn+2 + 2 Cl- = ZnCl2
        log_k           0.45
        delta_h 8.5 kcal

Zn+2 + 3Cl- = ZnCl3-
        log_k           0.5
        delta_h 9.56 kcal

Zn+2 + 4Cl- = ZnCl4-2
        log_k           0.2
        delta_h 10.96 kcal

Zn+2 + CO3-2 = ZnCO3
        log_k           5.3

Zn+2 + 2CO3-2 = Zn(CO3)2-2
        log_k           9.63

Zn+2 + HCO3- = ZnHCO3+
        log_k           2.1

Zn+2 + SO4-2 = ZnSO4
        log_k           2.37
        delta_h 1.36 kcal

Zn+2 + 2SO4-2 = Zn(SO4)2-2
        log_k           3.28

Cd+2 + H2O = CdOH+ + H+
        log_k           -10.080
        delta_h 13.1 kcal

Cd+2 + 2 H2O = Cd(OH)2 + 2 H+
        log_k           -20.350

Cd+2 + 3 H2O = Cd(OH)3- + 3 H+
        log_k           -33.300

Cd+2 + 4 H2O = Cd(OH)4-2 + 4 H+
        log_k           -47.350

Cd+2 + Cl- = CdCl+
        log_k           1.980
        delta_h 0.59 kcal

Cd+2 + 2 Cl- = CdCl2
        log_k           2.600
        delta_h 1.24 kcal

Cd+2 + 3 Cl- = CdCl3-
        log_k           2.400
        delta_h 3.9 kcal

Cd+2 + CO3-2 = CdCO3
        log_k           2.9

Cd+2 + 2CO3-2 = Cd(CO3)2-2
        log_k           6.4

Cd+2 + HCO3- = CdHCO3+
        log_k           1.5

Cd+2 + SO4-2 = CdSO4
        log_k           2.460
        delta_h 1.08 kcal

Cd+2 + 2SO4-2 = Cd(SO4)2-2
        log_k           3.5

Pb+2 + H2O = PbOH+ + H+
        log_k           -7.710

Pb+2 + 2 H2O = Pb(OH)2 + 2 H+
        log_k           -17.120

Pb+2 + 3 H2O = Pb(OH)3- + 3 H+
        log_k           -28.060

Pb+2 + 4 H2O = Pb(OH)4-2 + 4 H+
        log_k           -39.700

2 Pb+2 + H2O = Pb2OH+3 + H+
        log_k           -6.360

Pb+2 + Cl- = PbCl+
        log_k           1.600
        delta_h 4.38 kcal

Pb+2 + 2 Cl- = PbCl2
        log_k           1.800
        delta_h 1.08 kcal

Pb+2 + 3 Cl- = PbCl3-
        log_k           1.700
        delta_h 2.17 kcal

Pb+2 + 4 Cl- = PbCl4-2
        log_k           1.380
        delta_h 3.53 kcal
```

```
Pb+2 + CO3-2 = PbCO3
        log_k           7.240

Pb+2 + 2 CO3-2 = Pb(CO3)2-2
        log_k           10.640

Pb+2 + HCO3- = PbHCO3+
        log_k           2.9

Pb+2 + SO4-2 = PbSO4
        log_k           2.750

Pb+2 + 2 SO4-2 = Pb(SO4)2-2
        log_k           3.470

Pb+2 + NO3- = PbNO3+
        log_k           1.170

PHASES

Calcite
        CaCO3 = CO3-2 + Ca+2
        log_k           -8.480
        delta_h -2.297 kcal
        -analytic       -171.9065     -0.077993      2839.319      71.595

Aragonite
        CaCO3 = CO3-2 + Ca+2
        log_k           -8.336
        delta_h -2.589 kcal
        -analytic       -171.9773     -0.077993      2903.293      71.595

Dolomite
        CaMg(CO3)2 = Ca+2 + Mg+2 + 2 CO3-2
        log_k           -17.090
        delta_h -9.436 kcal

Siderite
        FeCO3 = Fe+2 + CO3-2
        log_k           -10.890
        delta_h -2.480 kcal

Rhodochrosite
        MnCO3 = Mn+2 + CO3-2
        log_k           -11.130
        delta_h -1.430 kcal

Strontianite
        SrCO3 = Sr+2 + CO3-2
        log_k           -9.271
        delta_h -0.400 kcal
        -analytic       155.0305      0.0           -7239.594     -56.58638

Witherite
        BaCO3 = Ba+2 + CO3-2
        log_k           -8.562
        delta_h 0.703 kcal
        -analytic       607.642       0.121098      -20011.25     -236.4948

Gypsum
        CaSO4:2H2O = Ca+2 + SO4-2 + 2 H2O
        log_k           -4.580
        delta_h -0.109 kcal
        -analytic       68.2401       0.0           -3221.51      -25.0627

Anhydrite
        CaSO4 = Ca+2 + SO4-2
        log_k           -4.360
        delta_h -1.710 kcal
        -analytic       197.52        0.0           -8669.8       -69.835

Celestite
        SrSO4 = Sr+2 + SO4-2
        log_k           -6.630
        delta_h -1.037 kcal
        -analytic       -14805.9622    -2.4660924   756968.533    5436.3588      -40553604.0

Barite
        BaSO4 = Ba+2 + SO4-2
        log_k           -9.970
        delta_h 6.350 kcal
        -analytic       136.035       0.0           -7680.41      -48.595

Hydroxyapatite
        Ca5(PO4)3OH + 4 H+ = H2O + 3 HPO4-2 + 5 Ca+2
        log_k           -3.421
        delta_h -36.155 kcal

Fluorite
        CaF2 = Ca+2 + 2 F-
        log_k           -10.600
        delta_h 4.690 kcal
        -analytic       66.348        0.0           -4298.2       -25.271

SiO2(a)
        SiO2 + 2 H2O = H4SiO4
        log_k           -2.710
        delta_h 3.340 kcal
```

```
        -analytic       -0.26           0.0           -731.0

Chalcedony
        SiO2 + 2 H2O = H4SiO4
        log_k           -3.550
        delta_h 4.720 kcal
        -analytic       -0.09           0.0           -1032.0

Quartz
        SiO2 + 2 H2O = H4SiO4
        log_k           -3.980
        delta_h 5.990 kcal
        -analytic       0.41          0.0           -1309.0

Gibbsite
        Al(OH)3 + 3 H+ = Al+3 + 3 H2O
        log_k           8.110
        delta_h -22.800 kcal

Al(OH)3(a)
        Al(OH)3 + 3 H+ = Al+3 + 3 H2O
        log_k           10.800
        delta_h -26.500 kcal

Kaolinite
        Al2Si2O5(OH)4 + 6 H+ = H2O + 2 H4SiO4 + 2 Al+3
        log_k           7.435
        delta_h -35.300 kcal

Albite
        NaAlSi3O8 + 8 H2O = Na+ + Al(OH)4- + 3 H4SiO4
        log_k           -18.002
        delta_h 25.896 kcal

Anorthite
        CaAl2Si2O8 + 8 H2O = Ca+2 + 2 Al(OH)4- + 2 H4SiO4
        log_k           -19.714
        delta_h 11.580 kcal

K-feldspar
        KAlSi3O8 + 8 H2O = K+ + Al(OH)4- + 3 H4SiO4
        log_k           -20.573
        delta_h 30.820  kcal

K-mica
        KAl3Si3O10(OH)2 + 10 H+ = K+ + 3 Al+3 + 3 H4SiO4
        log_k           12.703
        delta_h -59.376 kcal

Chlorite(14A)
        Mg5Al2Si3O10(OH)8 + 16H+ = 5Mg+2 + 2Al+3 + 3H4SiO4 + 6H2O
        log_k           68.38
        delta_h -151.494 kcal

Ca-Montmorillonite
        Ca0.165Al2.33Si3.67O10(OH)2 + 12 H2O = 0.165Ca+2 + 2.33 Al(OH)4- + 3.67 H4SiO4 + 2 H+
        log_k           -45.027
        delta_h 58.373  kcal

Talc
        Mg3Si4O10(OH)2 + 4 H2O + 6 H+ = 3 Mg+2 + 4 H4SiO4
        log_k           21.399
        delta_h -46.352 kcal

Illite
        K0.6Mg0.25Al2.3Si3.5O10(OH)2 + 11.2H2O = 0.6K+ + 0.25Mg+2 + 2.3Al(OH)4- + 3.5H4SiO4 + 1.2H+
        log_k           -40.267
        delta_h 54.684 kcal

Chrysotile
        Mg3Si2O5(OH)4 + 6 H+ = H2O + 2 H4SiO4 + 3 Mg+2
        log_k           32.200
        delta_h -46.800 kcal
        -analytic       13.248        0.0           10217.1       -6.1894

Sepiolite
        Mg2Si3O7.5OH:3H2O + 4 H+ + 0.5H2O = 2 Mg+2 + 3 H4SiO4
        log_k           15.760
        delta_h -10.700 kcal

Sepiolite(d)
        Mg2Si3O7.5OH:3H2O + 4 H+ + 0.5H2O = 2 Mg+2 + 3 H4SiO4
        log_k           18.660

Hematite
        Fe2O3 + 6 H+ = 2 Fe+3 + 3 H2O
        log_k           -4.008
        delta_h -30.845 kcal

Goethite
        FeOOH + 3 H+ = Fe+3 + 2 H2O
        log_k           -1.000
        delta_h         -14.48 kcal

Fe(OH)3(a)
        Fe(OH)3 + 3 H+ = Fe+3 + 3 H2O
        log_k           4.891

Pyrite
```

```
        FeS2 + 2 H+ + 2 e- = Fe+2 + 2 HS-
        log_k           -18.479
        delta_h 11.300 kcal

FeS(ppt)
        FeS + H+ = Fe+2 + HS-
        log_k           -3.915

Mackinawite
        FeS + H+ = Fe+2 + HS-
        log_k           -4.648

Sulfur
        S + 2H+ + 2e- = H2S
        log_k           4.882
        delta_h -9.5 kcal

Vivianite
        Fe3(PO4)2:8H2O = 3 Fe+2 + 2 PO4-3 + 8 H2O
        log_k           -36.000

Pyrolusite
        MnO2 + 4 H+ + 2 e- = Mn+2 + 2 H2O
        log_k           41.380
        delta_h -65.110 kcal

Hausmannite
        Mn3O4 + 8 H+ + 2 e- = 3 Mn+2 + 4 H2O
        log_k           61.030
        delta_h -100.640 kcal

Manganite
        MnOOH + 3 H+ + e- = Mn+2 + 2 H2O
        log_k           25.340

Pyrochroite
        Mn(OH)2 + 2 H+ = Mn+2 + 2 H2O
        log_k           15.200

Halite
        NaCl = Na+ + Cl-
        log_k           1.582
        delta_h 0.918 kcal

CO2(g)
        CO2 = CO2
        log_k           -1.468
        delta_h -4.776 kcal
        -analytic       108.3865      0.01985076   -6919.53      -40.45154      669365.0

O2(g)
        O2 = O2
        log_k           -2.960
        delta_h -1.844  kcal

H2(g)
        H2 = H2
        log_k           -3.150
        delta_h -1.759  kcal

H2O(g)
        H2O = H2O
        log_k           1.51
        delta_h         -44.03   kJ
#    Stumm and Morgan, from NBS and Robie, Hemmingway, and Fischer (1978)

N2(g)
        N2 = N2
        log_k           -3.260
        delta_h -1.358  kcal

H2S(g)
        H2S = H2S
        log_k           -0.997
        delta_h -4.570  kcal

CH4(g)
        CH4 = CH4
        log_k           -2.860
        delta_h -3.373  kcal

NH3(g)
        NH3 = NH3
        log_k           1.770
        delta_h -8.170  kcal

Melanterite
        FeSO4:7H2O = 7 H2O + Fe+2 + SO4-2
        log_k           -2.209
        delta_h 4.910           kcal
        -analytic       1.447        -0.004153      0.0           0.0          -214949.0

Alunite
        KAl3(SO4)2(OH)6 + 6 H+ = K+ + 3 Al+3 + 2 SO4-2 + 6H2O
        log_k           -1.400
        delta_h -50.250 kcal

Jarosite-K
        KFe3(SO4)2(OH)6 + 6 H+ = 3 Fe+3 + 6 H2O + K+ + 2 SO4-2
```

```
        log_k           -9.210
        delta_h -31.280 kcal

Zn(OH)2(e)
        Zn(OH)2 + 2 H+ = Zn+2 + 2 H2O
        log_k           11.50

Smithsonite
        ZnCO3 = Zn+2 + CO3-2
        log_k           -10.000
        delta_h -4.36   kcal

Sphalerite
        ZnS + H+ = Zn+2 + HS-
        log_k           -11.618
        delta_h 8.250   kcal

Willemite           289
        Zn2SiO4 + 4H+ = 2Zn+2 + H4SiO4
        log_k           15.33
        delta_h -33.37  kcal

Cd(OH)2
        Cd(OH)2 + 2 H+ = Cd+2 + 2 H2O
        log_k           13.650

Otavite             315
        CdCO3 = Cd+2 + CO3-2
        log_k           -12.1
        delta_h -0.019  kcal

CdSiO3              328
        CdSiO3 + H2O + 2H+ = Cd+2 + H4SiO4
        log_k           9.06
        delta_h -16.63  kcal

CdSO4               329
        CdSO4 = Cd+2 + SO4-2
        log_k           -0.1
        delta_h -14.74  kcal

Cerrusite           365
        PbCO3 = Pb+2 + CO3-2
        log_k           -13.13
        delta_h 4.86    kcal

Anglesite           384
        PbSO4 = Pb+2 + SO4-2
        log_k           -7.79
        delta_h 2.15    kcal

Pb(OH)2             389
        Pb(OH)2 + 2H+ = Pb+2 + 2H2O
        log_k           8.15
        delta_h -13.99  kcal

EXCHANGE_MASTER_SPECIES
        X       X-
EXCHANGE_SPECIES
        X- = X-
        log_k           0.0

        Na+ + X- = NaX
        log_k   0.0
        -gamma  4.0     0.075

        K+ + X- = KX
        log_k   0.7
        -gamma  3.5     0.015
        delta_h  -4.3   # Jardine & Sparks, 1984

        Li+ + X- = LiX
        log_k   -0.08
        -gamma  6.0     0.0
        delta_h  1.4    # Merriam & Thomas, 1956

        NH4+ + X- = NH4X
        log_k   0.6
        -gamma  2.5     0.0
        delta_h  -2.4   # Laudelout et al., 1968

        Ca+2 + 2X- = CaX2
        log_k   0.8
        -gamma  5.0     0.165
        delta_h  7.2    # Van Bladel & Gheyl, 1980

        Mg+2 + 2X- = MgX2
        log_k   0.6
        -gamma  5.5     0.2
        delta_h  7.4    # Laudelout et al., 1968

        Sr+2 + 2X- = SrX2
        log_k   0.91
        -gamma  5.26    0.121
        delta_h  5.5    # Laudelout et al., 1968

        Ba+2 + 2X- = BaX2
        log_k   0.91
        -gamma  5.0     0.0
```

```
        delta_h  4.5    # Laudelout et al., 1968

        Mn+2 + 2X- = MnX2
        log_k   0.52
        -gamma  6.0     0.0

        Fe+2 + 2X- = FeX2
        log_k   0.44
        -gamma  6.0     0.0

        Cu+2 + 2X- = CuX2
        log_k   0.6
        -gamma  6.0     0.0

        Zn+2 + 2X- = ZnX2
        log_k   0.8
        -gamma  5.0     0.0

        Cd+2 + 2X- = CdX2
        log_k   0.8

        Pb+2 + 2X- = PbX2
        log_k   1.05

        Al+3 + 3X- = AlX3
        log_k   0.41
        -gamma  9.0     0.0

        AlOH+2 + 2X- = AlOHX2
        log_k   0.89
        -gamma  0.0     0.0
SURFACE_MASTER_SPECIES
        Hfo_s  Hfo_sOH
        Hfo_w  Hfo_wOH
SURFACE_SPECIES
#   All surface data from
#   Dzombak and Morel, 1990
#
#
#   Acid-base data from table 5.7
#
#   strong binding site--Hfo_s,

        Hfo_sOH = Hfo_sOH
        log_k  0.0

        Hfo_sOH  + H+ = Hfo_sOH2+
        log_k  7.29    # = pKa1,int

        Hfo_sOH = Hfo_sO- + H+
        log_k  -8.93   # = -pKa2,int

#   weak binding site--Hfo_w

        Hfo_wOH = Hfo_wOH
        log_k  0.0

        Hfo_wOH  + H+ = Hfo_wOH2+
        log_k  7.29    # = pKa1,int

        Hfo_wOH = Hfo_wO- + H+
        log_k  -8.93   # = -pKa2,int

###############################################
#             CATIONS                         #
###############################################
#
#   Cations from table 10.1 or 10.5
#
#   Calcium
        Hfo_sOH + Ca+2 = Hfo_sOHCa+2
        log_k  4.97

        Hfo_wOH + Ca+2 = Hfo_wOCa+ + H+
        log_k -5.85
#   Strontium
        Hfo_sOH + Sr+2 = Hfo_sOHSr+2
        log_k  5.01

        Hfo_wOH + Sr+2 = Hfo_wOSr+ + H+
        log_k -6.58

        Hfo_wOH + Sr+2 + H2O = Hfo_wOSrOH + 2H+
        log_k -17.60
#   Barium
        Hfo_sOH + Ba+2 = Hfo_sOHBa+2
        log_k  5.46

        Hfo_wOH + Ba+2 = Hfo_wOBa+ + H+
        log_k  -7.2                    # table 10.5
#
#   Cations from table 10.2
#
#   Cadmium
        Hfo_sOH + Cd+2 = Hfo_sOCd+ + H+
        log_k  0.47

        Hfo_wOH + Cd+2 = Hfo_wOCd+ + H+
        log_k   -2.91
```

```
#   Zinc
        Hfo_sOH + Zn+2 = Hfo_sOZn+ + H+
        log_k  0.99

        Hfo_wOH + Zn+2 = Hfo_wOZn+ + H+
        log_k   -1.99
#   Copper
        Hfo_sOH + Cu+2 = Hfo_sOCu+ + H+
        log_k  2.89

        Hfo_wOH + Cu+2 = Hfo_wOCu+ + H+
        log_k  0.6                      # table 10.5
#   Lead
        Hfo_sOH + Pb+2 = Hfo_sOPb+ + H+
        log_k  4.65

        Hfo_wOH + Pb+2 = Hfo_wOPb+ + H+
        log_k  0.3                      # table 10.5
#
#   Derived constants table 10.5
#
#   Magnesium
        Hfo_wOH + Mg+2 = Hfo_wOMg+ + H+
        log_k -4.6
#   Manganese
        Hfo_sOH + Mn+2 = Hfo_sOMn+ + H+
        log_k  -0.4                     # table 10.5

        Hfo_wOH + Mn+2 = Hfo_wOMn+ + H+
        log_k -3.5                      # table 10.5
# Iron
        Hfo_sOH + Fe+2 = Hfo_sOFe+ + H+
        log_k   0.7     # LFER using table 10.5

        Hfo_wOH + Fe+2 = Hfo_wOFe+ + H+
        log_k -2.5      # LFER using table 10.5

###############################################
#             ANIONS                          #
###############################################
#
#   Anions from table 10.6
#
#   Phosphate
        Hfo_wOH + PO4-3 + 3H+ = Hfo_wH2PO4 + H2O
        log_k   31.29

        Hfo_wOH + PO4-3 + 2H+ = Hfo_wHPO4- + H2O
        log_k   25.39

        Hfo_wOH + PO4-3 + H+ = Hfo_wPO4-2 + H2O
        log_k   17.72
#
#   Anions from table 10.7
#
#   Borate
        Hfo_wOH + H3BO3 = Hfo_wH2BO3 + H2O
        log_k   0.62
#
#   Anions from table 10.8
#
#   Sulfate
        Hfo_wOH + SO4-2 + H+ = Hfo_wSO4- + H2O
        log_k   7.78

        Hfo_wOH + SO4-2 = Hfo_wOHSO4-2
        log_k   0.79
#
#   Derived constants table 10.10
#
        Hfo_wOH + F- + H+ = Hfo_wF + H2O
        log_k   8.7

        Hfo_wOH + F- = Hfo_wOHF-
        log_k   1.6
#
# Carbonate: Van Geen et al., 1994 reoptimized for HFO
# 0.15 g HFO/L has 0.344 mM sites == 2 g of Van Geen's Goethite/L
#
#        Hfo_wOH + CO3-2 + H+ = Hfo_wCO3- + H2O
#        log_k   12.56
#
#        Hfo_wOH + CO3-2 + 2H+= Hfo_wHCO3 + H2O
#        log_k   20.62

# 9/19/96
#       Added analytical expression for H2S, NH3, KSO4.
#       Added species CaHSO4+.
#       Added delta H for Goethite.

RATES

############
#K-feldspar
############
#
# Sverdrup, H.U., 1990, The kinetics of base cation release due to
# chemical weathering: Lund University Press, Lund, 246 p.
#
```

```
# Example of KINETICS data block for K-feldspar rate:
#       KINETICS 1
#       K-feldspar
#               -m0 2.16  # 10% K-fsp, 0.1 mm cubes
#               -m  1.94
#               -parms 1.36e4  0.1

K-feldspar
 -start
   1 rem specific rate from Sverdrup, 1990, in kmol/m2/s
   2 rem parm(1) = 10 * (A/V, 1/dm) (recalc's sp. rate to mol/kgw)
   3 rem parm(2) = corrects for field rate relative to lab rate
   4 rem temp corr: from p. 162. E (kJ/mol) / R / 2.303 = H in H*(1/T-1/298)

   10    dif_temp = 1/TK - 1/298
   20    pk_H = 12.5 + 3134 * dif_temp
   30    pk_w = 15.3 + 1838 * dif_temp
   40    pk_OH = 14.2 + 3134 * dif_temp
   50    pk_CO2 = 14.6 + 1677 * dif_temp
   #60   pk_org = 13.9 + 1254 * dif_temp  # rate increase with DOC
   70    rate = 10^-pk_H * ACT("H+")^0.5 + 10^-pk_w + 10^-pk_OH * ACT("OH-")^0.3
   71    rate = rate + 10^-pk_CO2 * (10^SI("CO2(g)"))^0.6
   #72   rate = rate + 10^-pk_org * TOT("Doc")^0.4
   80    moles = parm(1) * parm(2) * rate * (1 - SR("K-feldspar")) * time
   81 rem decrease rate on precipitation
   90    if SR("K-feldspar") > 1 then moles = moles * 0.1
   100   save moles
 -end

############
#Albite
############
#
# Sverdrup, H.U., 1990, The kinetics of base cation release due to
# chemical weathering: Lund University Press, Lund, 246 p.
#
# Example of KINETICS data block for Albite rate:
#       KINETICS 1
#       Albite
#               -m0 0.43  # 2% Albite, 0.1 mm cubes
#               -parms 2.72e3  0.1

Albite
 -start
   1 rem specific rate from Sverdrup, 1990, in kmol/m2/s
   2 rem parm(1) = 10 * (A/V, 1/dm) (recalc's sp. rate to mol/kgw)
   3 rem parm(2) = corrects for field rate relative to lab rate
   4 rem temp corr: from p. 162. E (kJ/mol) / R / 2.303 = H in H*(1/T-1/298)

   10    dif_temp = 1/TK - 1/298
   20    pk_H = 12.5 + 3359 * dif_temp
   30    pk_w = 14.8 + 2648 * dif_temp
   40    pk_OH = 13.7 + 3359 * dif_temp
   #41 rem         ^12.9 in Sverdrup, but larger than for oligoclase...
   50    pk_CO2 = 14.0 + 1677 * dif_temp
   #60   pk_org = 12.5 + 1254 * dif_temp # ...rate increase for DOC
   70    rate = 10^-pk_H * ACT("H+")^0.5 + 10^-pk_w + 10^-pk_OH * ACT("OH-")^0.3
   71    rate = rate + 10^-pk_CO2 * (10^SI("CO2(g)"))^0.6
   #72   rate = rate + 10^-pk_org * TOT("Doc")^0.4
   80    moles = parm(1) * parm(2) * rate * (1 - SR("Albite")) * time
   81 rem decrease rate on precipitation
   90    if SR("Albite") > 1 then moles = moles * 0.1
   100   save moles
 -end

########
#Calcite
########
#
# Plummer, L.N., Wigley, T.M.L., and Parkhurst, D.L., 1978,
# American Journal of Science, v. 278, p. 179-216.
#
# Example of KINETICS data block for calcite rate:
#
#       KINETICS 1
#       Calcite
#               -tol    1e-8
#               -m0     3.e-3
#               -m      3.e-3
#               -parms  5.0      0.6
Calcite
  -start
   1 rem        Modified from Plummer and others, 1978
   2 rem        parm(1) = A/V, 1/m     parm(2) = exponent for m/m0

   10 si_cc = si("Calcite")
   20 if (m <= 0  and si_cc < 0) then goto 200
   30  k1 = 10^(0.198 - 444.0 / (273.16 + tc) )
   40  k2 = 10^(2.84 - 2177.0 / (273.16 + tc) )
   50  if tc <= 25 then k3 = 10^(-5.86 - 317.0 / (273.16 + tc) )
   60  if tc > 25 then k3 = 10^(-1.1 - 1737.0 / (273.16 + tc) )
   70   t = 1
   80   if m0 > 0 then t = m/m0
   90   if t = 0 then t = 1
   100   moles = parm(1) * (t)^parm(2)
   110   moles = moles * (k1 * act("H+") + k2 * act("CO2") + k3 * act("H2O"))
   120   moles = moles * (1 - 10^(2/3*si_cc))
   130   moles = moles * time
   140  if (moles > m) then moles = m
```

```
   150 if (moles >= 0) then goto 200
   160  temp = tot("Ca")
   170  mc  = tot("C(4)")
   180  if mc < temp then temp = mc
   190  if -moles > temp then moles = -temp
   200 save moles
  -end

#######
#Pyrite
#######
#
# Williamson, M.A. and Rimstidt, J.D., 1994,
# Geochimica et Cosmochimica Acta, v. 58, p. 5443-5454.
#
# Example of KINETICS data block for pyrite rate:
#       KINETICS 1
#       Pyrite
#               -tol    1e-8
#               -m0     5.e-4
#               -m      5.e-4
#               -parms  2.0     0.67     .5      -0.11
Pyrite
  -start
   1 rem        Williamson and Rimstidt, 1994
   2 rem        parm(1) = log10(A/V, 1/dm)      parm(2) = exp for (m/m0)
   3 rem        parm(3) = exp for O2            parm(4) = exp for H+

   10 if (m <= 0) then goto 200
   20 if (si("Pyrite") >= 0) then goto 200
   20  rate = -10.19 + parm(1) + parm(3)*lm("O2") + parm(4)*lm("H+") + parm(2)*log10(m/m0)
   30  moles = 10^rate * time
   40 if (moles > m) then moles = m
   200 save moles
  -end

##########
#Organic_C
##########
#
# Example of KINETICS data block for Organic_C rate:
#       KINETICS 1
#       Organic_C
#               -tol    1e-8
#              # m in mol/kgw
#               -m0     5e-3
#               -m      5e-3
Organic_C
 -start
   1  rem      Additive Monod kinetics
   2  rem      Electron acceptors: O2, NO3, and SO4

   10 if (m <= 0) then goto 200
   20  mO2 = mol("O2")
   30  mNO3 = tot("N(5)")
   40  mSO4 = tot("S(6)")
   50   rate = 1.57e-9*mO2/(2.94e-4 + mO2) + 1.67e-11*mNO3/(1.55e-4 + mNO3)
   60   rate = rate + 1.e-13*mSO4/(1.e-4 + mSO4)
   70  moles = rate * m * (m/m0) * time
   80 if (moles > m) then moles = m
   200 save moles
 -end

###########
#Pyrolusite
###########
#
# Postma, D. and Appelo, C.A.J., 2000, GCA 64, in press
#
# Example of KINETICS data block for Pyrolusite
#       KINETICS 1-12
#       Pyrolusite
#               -tol    1.e-7
#               -m0     0.1
#               -m      0.1
Pyrolusite
  -start
   5    if (m <= 0.0) then goto 200
   7    sr_pl = sr("Pyrolusite")
   9    if abs(1 - sr_pl) < 0.1 then goto 200
   10   if (sr_pl > 1.0) then goto 100
   #20 rem      initially 1 mol Fe+2 = 0.5 mol pyrolusite. k*A/V = 1/time (3 cells)
   #22 rem       time (3 cells) = 1.432e4.  1/time = 6.98e-5
   30   Fe_t = tot("Fe(2)")
   32   if Fe_t < 1.e-8 then goto 200
   40   moles = 6.98e-5 * Fe_t  * (m/m0)^0.67 * time * (1 - sr_pl)
   50   if moles > Fe_t / 2 then moles = Fe_t / 2
   70   if moles > m then moles = m
   90   goto 200
   100  Mn_t = tot("Mn")
   110  moles = 2e-3 * 6.98e-5 * (1-sr_pl) * time
   120  if moles <= -Mn_t then moles = -Mn_t
   200  save moles
  -end
END
```

Attachment C--Input File To Investigate the Order of the Numerical Method For Example 12

The following input data file performs calculations for example 12, which models the diffusion of heat and solute from a constant boundary condition at one end and a closed boundary condition at the other. The input file is simplified from that given in example 12 by eliminating one of the transport calculations and setting up the initial conditions directly with **SOLUTION** data blocks.

For the constant boundary condition, an analytical solution exists for the temperature and concentration as a function of time and distance. To test the accuracy of the numerical method, the **TRANSPORT** calculation is performed with two discretizations, a 20-cell model and a 60-cell model. For a second-order method, decreasing the cell size by a factor of three should decrease the discrepancy with the analytical solution by a factor of about nine. The deviations from the analytical solutions for each model at the end of the time step are calculated and stored with PUT statements for distances from 0.5 to 8.5 m (in 0.5 m increments) by the Basic program in the **USER_PUNCH** data block. The stored results are printed in an additional simulation with a **USER_PRINT** data block. The final simulation is a dummy calculation for pure water, none of the results are printed. However, the calculation causes **USER_PRINT** to be invoked. The Basic program in the **USER_PRINT** data block uses GET statements and prints the results stored by the **USER_PUNCH** Basic program during the previous **TRANSPORT** simulations.

Table 56.--Input data set for example 12 demonstrating second-order accuracy of the numerical method

```
TITLE Example 12a.--Advective and diffusive transport of heat and solutes.
      Constant boundary condition at one end, closed at other.
      The problem is designed so that temperature should equal Na-conc
      (in mmol/kgw) after diffusion. Compares with analytical solution
      for 20-cell and 60-cell models.
EXCHANGE_SPECIES
  Na+ + X- = NaX
    log_k       0.0
    -gamma      4.0     0.075
  H+ + X- = HX
    log_k       -99.
    -gamma      9.0     0.0
  K+ + X- = KX
    log_k       0.0
    -gamma       3.5     0.015
#
#   20-cell model, initial conditions
#
SOLUTION 0   Fixed temp 24C, and NaCl conc (first type boundary cond) at inlet
    units  mol/kgw
    temp 24
    pH  7.0
    pe  12.0   O2(g) -0.67
    Na  24.e-3
    Cl  24.e-3
SOLUTION 1-19   24.0 mM KNO3
    units mol/kgw
    temp  0                 # Incoming solution 0C
    pH    7.0
```

```
    pe   12.0   O2(g) -0.67
    K    24.e-3
    N(5) 24.e-3
EXCHANGE 1-19
    KX    0.048
SOLUTION 20  Same as soln 0 in cell 20 at closed column end (second type boundary cond)
    units  mol/kgw
    temp 24
    pH  7.0
    pe  12.0   O2(g) -0.67
    Na  24.e-3
    Cl  24.e-3
EXCHANGE 20
    NaX    0.048
PRINT
   -reset   false
END
#
#   20-cell model, transport
#
TRANSPORT                    # Diffuse 24C, NaCl solution from column end
   -cells   20
   -shifts 1
   -flow_d diffusion
   -bcon   constant  closed
   -length  1.0
   -thermal_diffusion  3.0  # Heat is retarded equal to Na
   -disp    0.0             # No dispersion
   -diffc  0.3e-9           # m^2/s
   -timest 1.0e+10          # 317 years, 19 substeps will be used
SELECTED_OUTPUT
   -file    ex12a.sel
   -high_precision  true
   -reset   false
   -dist    true
   -temp    true
USER_PUNCH
  -head Na_mmol K_mmol Cl_mmol Cl-analytic  Na_analytic
  10  PUNCH TOT("Na")*1000, TOT("K")*1000, TOT("Cl")*1000
#
#   Calculate deviation from analytical solution for Cl and Na
#
  20  DATA 0.254829592, -0.284496736, 1.421413741, -1.453152027, 1.061405429
  30  x = DIST
  40  if (x > 8.5 OR SIM_TIME <= 0) THEN END
  50  IF (ABS(x MOD 0.5) > 1e-3) OR (TC <= 0) THEN END
  60  READ a1, a2, a3, a4, a5
  70      REM  calculate error in Cl
  80  Arg = x / (2*SQRT(3e-10 * SIM_TIME / 1.0))
  90  e = 1/(1 + 0.3275911 * Arg)
  100 erfc_Cl = (e*(a1+e*(a2+e*(a3+e*(a4+e*a5)))))*EXP(-Arg*Arg);
  110     REM  calculate error in Na
  120 Arg = x / (2*SQRT(3e-10 * SIM_TIME / 3.0))
  130 e = 1/(1 + 0.3275911 * Arg)
```

```
  140 erfc_Na = (e*(a1+e*(a2+e*(a3+e*(a4+e*a5)))))*EXP(-Arg*Arg);
  150     REM punch results
  160 error_Cl = 0.024 * erfc_Cl - TOT("Cl")
  170 error_Na = 0.024 * erfc_Na - TOT("Na")
  180 PUNCH error_Cl, error_Na
  190     REM store results
  200 j = x - 0.5
  210 PUT(error_Cl, SIM_NO, j, 1)
  220 PUT(error_Na, SIM_NO, j, 2)
  500 END
END
#
#   60-cell model, initial conditions
#
SELECTED_OUTPUT
   -user_punch false
SOLUTION 0  Fixed temp 24C, and NaCl conc (first type boundary cond) at inlet
    units  mol/kgw
    temp 24
    pH  7.0
    pe  12.0   O2(g) -0.67
    Na  24.e-3
    Cl  24.e-3
SOLUTION 1-59   24.0 mM KNO3
    units mol/kgw
    temp  0                 # Incoming solution 0C
    pH    7.0
    pe   12.0   O2(g) -0.67
    K    24.e-3
    N(5) 24.e-3
EXCHANGE 1-59
    KX    0.048
SOLUTION 60  Same as soln 0 in cell 60 at closed column end (second type boundary cond)
    units  mol/kgw
    temp 24
    pH  7.0
    pe  12.0   O2(g) -0.67
    Na  24.e-3
    Cl  24.e-3
EXCHANGE 60
    NaX    0.048
END
#
#   60-cell model, transport
#
TRANSPORT                   # Diffuse 24C, NaCl solution from column end
   -cells   60
   -shifts 1
   -flow_d diffusion
   -bcon   constant  closed
   -thermal_diffusion  3.0  # Heat is retarded equal to Na
   -disp    0.0             # No dispersion
   -diffc  0.3e-9           # m^2/s
   -length .33333333333333333
```

```
   -timest 1.0e+10          # 317 years
   -punch_cell 1-60
SELECTED_OUTPUT
   -high_precision  true
   -user_punch true
   -reset   false
   -dist    true
   -temp    true
END
#
#   Print comparison with analytical solution for
#   Cl and Na in 20-cell and 60-cell models
#
SOLUTION                    # Initial solution calculation for pure water
                            # A calculation is needed to invoke USER_PRINT
PRINT
   -reset false             # Initial solution calculation not printed
   -user_print true
SELECTED_OUTPUT
   -high_precision false    # Controls precision for USER_PRINT too.
USER_PRINT
  10  PRINT "             Error in Cl concentration Error in Na concentration"
  20  PRINT "             ------------------------- -------------------------"
  30  PRINT "    Distance      20-cell      60-cell      20-cell      60-cell"
  40  PRINT " "
  50  FOR j = 0 TO 8
  60    PRINT j + 0.5, GET(2, j, 1), GET(4, j, 1), GET(2, j, 2), GET(4, j, 2)
  70  NEXT j
END
```