UCRL-MA—110662-Pt.3
DE93 005827

EQ3NR, A Computer Program for Geochemical Aqueous Speciation-Solubility Calculations: Theoretical Manual, User's Guide, and Related Documentation (Version 7.0)

Thomas J. Wolery

Preface

This report on the EQ3NR code is a revision of the first EQ3NR User's Guide (Wolery, 1983). It is one of a set of reports documenting version 7.0 (version 3245.1090 under the old numbering system) of the EQ3/6 software package. This set includes:

I. The EQ3/6 Package Overview and Installation Guide (Wolery 1992).

II. The EQPT User's Guide (Daveler and Wolery, 1992).

III. The EQ3NR Theoretical Manual and User's Guide (this report).

IV. The EQ6 Theoretical Manual and User's Guide (Wolery and Daveler, 1992).

EQ3NR, the subject of the present report, is the speciation-solubility code in the EQ3/6 package. EQ6 is the reaction path code in EQ3/6. EQPT is the EQ3/6 data file preprocessor. The present report assumes that the reader is familiar with the contents of the EQ3/6 Package Overview and Installation Guide and the EQPT User's Guide.

The development of EQ3/6 has been supported by a number of programs concerned with geologic disposal of high level nuclear waste, including the Office of Nuclear Waste Isolation, the Salt Repository Project Office, the Waste Isolation Pilot Plant (through Sandia National Laboratory), the Nevada Nuclear Waste Storage Investigations, and the Yucca Mountain Site Characterization Project. Documentation for the package is aimed at satisfying the requirements of the U.S. Nuclear Regulatory Commission for software used for this purpose (Silling, 1983).

The Lawrence Livermore National Laboratory has not certified that EQ3/6 constitutes approved code for the conduct of quality affecting work for the Yucca Mountain Project.

No source codes or data files are reproduced in this report, nor are any computer media containing such items a part of this report or any of the other reports documenting this version of EQ3/6. The software itself must be obtained as described below.

The examples presented in this series of reports correspond to version 7.0 of the software and the R10 set of supporting thermodynamic data files. As of the date of publication of this report, the most recent version of the software is version 7.1 (containing bug fixes, but no enhancements), and the most recent set of data files is R16.

Agencies of the United States Government and their contractors may obtain copies of the software and its documentation from:

Energy Science and Technology Software Center
P. O. Box 1020
Oak Ridge, TN 37831-1020

Telephone: (615) 576-2606

Requests to obtain the software under a licensing agreement should be addressed to:

Technology Transfer Initiatives Program, L-795
Attn: Diana (Cookie) West
Lawrence Livermore National Laboratory
P.O. Box 808
Livermore, CA 94550

Telephone: (510) 423-7678
Fax: (510) 422-6416
Secretary: (510) 422-6416

Comments and questions concerning EQ3/6 exclusive of the thermodynamic data base should be addressed to the code custodian:

Thomas J. Wolery, L-219
Lawrence Livermore National Laboratory
P.O. Box 808
Livermore, CA 94550

E-mail: wolery1@llnl.gov
Telephone: (510) 422-5789
Fax: (510) 422-0208
Secretary: (510) 423-2970

Comments and questions which concern the EQ3/6 thermodynamic data base should be addressed to the data base custodian:

James W. Johnson, L-219
Lawrence Livermore National Laboratory
P.O. Box 808
Livermore, CA 94550

E-mail: johnson@s05.es.llnl.gov
Telephone: (510) 423-7352
Fax: (510) 422-0208
Secretary: (510) 423-2970

Contents

Glossary of Symbols

a, a'	Symbols used to represent cations in the notation of Harvie, Møller, and Weare (1984); see also X, X'.
a_i	Thermodynamic activity of the i-th aqueous solute species.
a_w	Thermodynamic activity of water.
$a_{\pm,MX}$	Mean activity of neutral electrolyte MX; $log a_{\pm,MX} = \frac{\nu_M log a_M + \nu_X log a_X}{\nu_{MX}}$.
$a_{\sigma\psi}$	Thermodynamic activity of the σ-th component of the ψ-th solid solution phase.
$\hat{a}$	Generalized hard core diameter or "ion size" in aqueous solution.
$\hat{a}_i$	Hard core diameter or "size" of the i-th aqueous solute species.
A	Affinity.
A_j	Thermodynamic affinity of the j-th reaction. The forward direction is implied, which in *EQ3/6* is taken to be that in which the associated species is destroyed, for example by dissolution or dissociation.
A_{+j}	Thermodynamic affinity of the j-th reaction (forward direction).
A_{-j}	Thermodynamic affinity of the j-th reaction (reverse direction).
Ah	Thermodynamic affinity (per electron) of a redox couple with respect to the standard hydrogen electrode; $Ah = F\,Eh$.
A_t	Titration alkalinity, in units of equivalents per kilogram of solvent water or equivalents/L.
$A_{\gamma,e}$	Debye-Hückel A parameter used in writing expressions for $ln\ \gamma_i$.
$A_{\gamma,10}$	Debye-Hückel A parameter used in writing expressions for $log_{10}\ \gamma_i$; $A_{\gamma,10} = 2.303\ A_{\gamma,e}$.
A_Φ	Debye-Hückel A parameter used in writing expressions for $ln\ a_w$.
A_t^*	Titration alkalinity, mg/L of equivalent $CaCO_3$; also denoted as T.
A^*_{t,HCO_3^-}	Titration alkalinity, mg/L of equivalent HCO_3^-.
$A^*_{HCO_3^-}$	Bicarbonate alkalinity, mg/L of equivalent $CaCO_3$.
$A^*_{CO_3^{2-}}$	Carbonate alkalinity, mg/L of equivalent $CaCO_3$.
$A^*_{OH^-}$	Hydroxide alkalinity, mg/L of equivalent $CaCO_3$.
b_{sr}	Stoichiometric reaction coefficient, the number of moles of the s-th aqueous species appearing in the r-th aqueous reaction; it is negative for reactants and positive for products.

$b_{s\phi}$	Stoichiometric reaction coefficient, the number of moles of the s-th aqueous species appearing in the reaction for the dissolution of the ϕ-th pure mineral; it is negative for reactants and positive for products.
b_{sg}	Stoichiometric reaction coefficient, the number of moles of the s-th aqueous species appearing in the reaction for the dissolution of the g-th gas species; it is negative for reactants and positive for products.
b	A parameter theoretically equivalent to the product $\dot{a}B_\gamma$ and appearing in Pitzer's equations with an fixed value of 1.2.
B_γ	Debye-Hückel B parameter used in writing expressions for $ln\ \gamma_i$ or $log_{10}\gamma_i$.
$B_{MX}(I)$	Observable second order interaction coefficient for neutral electrolyte MX (M = cation, X = anion); a function of the ionic strength.
$B'_{MX}(I)$	The derivative of $B_{MX}(I)$ with respect to ionic strength.
$B^{\phi}_{MX}(I)$	The compound function $B_{MX}(I) + IB'_{MX}(I)$.
c, c'	Symbols used to represent cations in the notation of Harvie, Møller, and Weare (1984); see also M, M'.
$C_{i,\ molar}$	Concentration of the i-th solute species in molarity (moles/L).
$C_{i,\ mg/L}$	Concentration of the i-th solute species in mg/L.
$C_{i,\ mg/kg}$	Concentration of the i-th solute species in mg/kg solution.
$C_{TS,\ mg/kg,}$	Total dissolved solutes in mg/kg solution.
$C_{TS,\ mg/L}$	Total dissolved solutes in mg/L.
C^{ϕ}_{MX}	Third order interaction coefficient for neutral electrolyte MX.
C_{MX}	The quantity $\dfrac{C^{\phi}_{MX}}{2\sqrt{\lvert z_M z_X\rvert}}$.
ε	Subscript indexing a chemical element.
ε_T	Total number of chemical elements in a system.
e^-	The electron. In common thermodynamic formalism, this is usually a hypothetical species, not a real one.
E_j	Electrical potential of the j-th redox couple, volts.
E^o_j	Standard state electrical potential of the j-th redox couple, volts.
Eh	Redox potential, volts. Theoretical equilibrium electrical potential of a redox couple; $Eh = \dfrac{2.303RT}{4F}(log f_{O_2} - 4pH - 2log a_w - log K_{Eh})$, where f_{O_2} is understood to be the hypothetical equilibrium oxygen fugacity in aqueous solution.
$f(I)$	Debye-Hückel f function.

$f(I)$	Debye-Hückel f^{γ} function; $f(I) = df/dI$.
f'	The quantity $f'(I)/2$.
f_g	Fugacity of the g-th gas.
f_{O_2}	Oxygen fugacity.
F	(1) The Faraday constant, 23062.3 cal/equiv-volt; (2) Compound electrostatic function used by Harvie, Møller, and Weare (1984) to write Pitzer's equations (see Chapter 3).
g	Subscript denoting a gas species.
g_T	Total number of gas species in a system.
$g(x)$	A function used to describe the ionic strength dependence of the second order interaction coefficient in Pitzer's equations.
$g'(x)$	The derivative of $g(x)$ with respect to x.
G^{EX}	Excess Gibbs energy, as of a solution.
H_{sr}	The factor $\frac{m_{s''}\, u_{s''\,r}}{b_{s''\,r}}$, where s'' denotes the dependent aqueous species which is associated with and destroyed by the r-th aqueous reaction.
H_{zr}	The factor $\frac{m_{s''}\, z_{s''}}{b_{s''\,r}}$ (analogous to H_{sr}).
I	Ionic strength.
IAP	Ion activity product; see Q.
$J(x)$	A function used to describe the higher order electrical interactions term in Pitzer's equations.
$J'(x)$	The derivative of $J(x)$ with respect to x.
J_{ij}	An element of the Jacobian matrix $(\frac{\partial \alpha_i}{\partial z_j})$.
$\underline{J}$	The Jacobian matrix.
K	Thermodynamic equilibrium constant.
K_{Eh}	Thermodynamic equilibrium constant for the half-reaction $2H_2O_{(l)} = O_{2(g)} + 4H^+ + 4e^-$.
K_w	Equilibrium constant for the reaction $H_2O_{(l)} = H^+ + OH^-$
m_i	Molal concentration of the i-th aqueous solute species (no contributions from dependent species).
$m_{T,i}$	Total molal concentration of the i-th aqueous species (includes contributions from dependent species).
M, M'	Symbols denoting cations (see also c, c').
M_i	Molecular weight of the i-th substance, grams per mole; e.g. M_w is the molecular weight of water.

n, n'	Symbols used to represent cations in the notation of Harvie, Møller, and Weare (1984); see also N, N'.
n_i	Number of moles of the i-th aqueous solute species.
n_s	Number of moles of the s-th aqueous species.
$n_{T,s}$	Total number of moles of the s-th (basis) aqueous solute species.
n_w	Number of moles of solvent water.
$n_{T,\varepsilon}$	Total number of moles of the ε-th chemical element.
N, N'	Symbols denoting neutral species (see also n, n').
N_w	Weight fraction of water in aqueous solution.
N_ψ	Site-mixing parameter for the ψ-th solid solution. If $N_\psi \approx 1$, the model is equivalent to a molecular-mixing model.
$O_{2(g)}$	Oxygen gas; in aqueous solution, this refers to a hypothetical species similar to e^-; also symbolized as s_B.
p_g	Partial pressure of the g-th gas, bars.
$p_{k\psi}$	The k-th parameter used to compute the interaction coefficients W_ψ, $W_{i\psi}$, $W_{ij\psi}$, which in turn are used to compute the activity coefficients of end-member components in the ψ-th solid solution.
P	(1) Pressure, bars; (2) Phenolphthalein alkalinity, equivalent mg/L of $CaCO_3$.
pe	Logarithm of the hypothetical electron activity; $pe = F\,Eh/(2.303\,RT) = Ah/(2.303\,RT)$.
pH	The quantity $-\log a_{H^+}$.
$pHCl$	The quantity $-\log a_{H^+} - \log a_{Cl^-}$.
Q	Activity product of a reaction; IAP is used by many others (e.g., Parkhurst et al., 1980) to denote the same quantity. "Q" implies Q_+, the activity product corresponding to the reaction taken in the forward direction.
Q_+	Activity product of a reaction, the same as Q.
Q_-	Reverse activity product of a reaction, equal to $1/Q_+$.
$Q_{+,1/2}$	Activity product of a half reaction.
$Q_{-,1/2}$	Reverse activity product of a half reaction, equal to $1/Q_{+,1/2}$.
r	Subscript denoting an aqueous reaction.
r_T	Total number of reactions for the dissociation/destruction of dependent aqueous species.
R	The gas constant, 1.98726 cal/mol-°K.
s	Subscript denoting an aqueous species ($s = w$ implies $H_2O_{(l)}$).
s'	Subscript denoting s in the range from 1 to s_Q, excluding the cases $s = w$ and $s = s_B$.

s''	Subscript implying the species formally associated with the aqueous reaction designated by r ($s'' = r + s_B$).
s_B	Subscript denoting the hypothetical aqueous species $O_{2(g)}$.
s_Q	The total number of aqueous master species; depending on the problem at hand, s_Q is equal to or greater than s_B.
s_T	Total number of aqueous species.
SI	Saturation index for a mineral; $SI = log\ (Q/K)$, where Q and K are the activity product and equilibrium constant, respectively, for the dissolution reaction.
T	(1) Temperature, °K; (2) Titration alkalinity, mg/L of equivalent $CaCO_3$.
V_g	The molar volume of an ideal gas, 22,413.6 ml/mole.
u	Stoichiometric mass balance coefficient calculated from reaction coefficients and certain model constraints; $u_{s's}$ is the stoichiometric factor for computing the contribution of the s-th aqueous species to the mass balance for the s'-th basis species.
w	Subscript denoting water (e.g., a_w, the activity of water).
w_w	Number of kilograms of solvent water.
W_ψ, $W_{i\psi}$, $W_{ij\psi}$	Interaction coefficients used to compute the activity coefficients of end-member components in the ψ-th solid solution.
$\underline{W}$	Array of partial derivatives of $log\, x_w$ with respect to $log\, m_{s'}$, where s' is a basis species. This derivative is zero for $s' = w$ or s_B.
x	A general algebraic variable.
x_i	Mole fraction of the i-th aqueous solute species.
x_w	Mole fraction of water in aqueous solution.
$x_{\sigma\psi}$	Mole fraction of the σ-th end member of the ψ-th solid solution.
X, X'	Symbols denoting anions (see also a, a').
z_s	Electrical charge of the s-th aqueous species.
z	Subscript denoting charge balance (e.g., H_z).
$\underline{z}$	Vector of algebraic master variables.
2.303	Symbol for and approximation of ln 10. As an approximation, this is not sufficiently accurate for general use in calculations; this constant should be computed to full machine accuracy in a computer code in order to avoid both inaccuracy and inconsistency.
α, α_1, α_2	Parameters appearing in Pitzer's equations.
$\underline{\alpha}$	Newton-Raphson residual function vector.
α_z	Residual function for charge balance.

α_s	Residual function for mass balance of the *s*-th basis species.
α_ϕ	Residual function for equilibrium with a pure mineral.
$\alpha_{\sigma\psi}$	Residual function for equilibrium with the σ-th end member of the ψ-th solid solution.
$\underline{\beta}$	Newton-Raphson residual function vector, identical to $\underline{\alpha}$, except that mass balance residual elements are normalized by the corresponding values of total numbers of moles.
β_{max}	The largest absolute value of any element of $\underline{\beta}$.
$\beta_{MX}^{(0)}, \beta_{MX}^{(1)}, \beta_{MX}^{(2)}$	Observable second order interaction coefficient parameters for neutral electrolyte *MX*.
γ_i	Molal activity coefficient of the *i*-th aqueous solute species.
$\gamma_{T,i}$	Stoichiometric molal activity coefficient of the *i*-th aqueous solute species; generally defined only for simple ions.
$\gamma_{\pm,MX}$	Mean molal activity coefficient of aqueous neutral electrolyte *MX*.
$\gamma_{T,\pm,MX}$	Stoichiometric mean molal activity coefficient of aqueous neutral electrolyte *MX*.
$\underline{\delta}$	Newton-Raphson correction term vector.
δ_{max}	The largest absolute value of any element of δ.
δ_{conv}	Convergence function.
δ'	Under-relaxation parameter.
$\Delta G_{f,i}$	Gibbs energy of formation of the *i*-th chemical species.
$\Delta G_{f,i}^o$	Standard state Gibbs energy of formation of the *i*-th chemical species.
ΔG_r	Gibbs energy of reaction of the *r*-th reaction.
ΔG_r^o	Standard state Gibbs energy of reaction of the *r*-th reaction.
ε	Subscript denoting a chemical element.
ε_T	Total number of chemical elements in a system.
ζ_{NMX}	Observable third order interaction coefficient for neutral species N, cation M, and anion X.
$\theta_{MM'}(I)$	Observable second order interaction coefficient for mixtures of neutral electrolytes *MX* and *M'X*; independent of the identity of *X* and a function of the ionic strength.
$\theta'_{MM'}(I)$	The derivative of $\theta_{MM'}(I)$ with respect to ionic strength.
${}^{E}\theta_{MM'}(I)$	The electrostatic part of $\theta_{MM'}(I)$.
${}^{S}\theta_{MM'}$	The short-range part of $\theta_{MM'}(I)$; treated as a constant.
κ	Under-relaxation parameter in Newton-Raphson iteration.

$\lambda_{ij}(I)$	Second-order interaction coefficient for the i-th and j-th aqueous solute species; in general, this is a function of the ionic strength.
$\lambda'_{ij}(I)$	The derivative of $\lambda_{ij}(I)$ with respect to ionic strength.
$\lambda_{MX}^{(0)}, \lambda_{MX}^{(1)}, \lambda_{MX}^{(2)}$	Second order interaction coefficient parameters for cation M and anion X.
λ_w	Rational (mole fraction) activity coefficient of water; $a_w = \lambda_w x_w$.
$\lambda_{\sigma\psi}$	Rational (mole fraction) activity coefficient of the σ-th end member of the ψ-th solid solution.
$^{E}\lambda_{MM'}(I)$	The electrostatic part of $\lambda_{MM'}(I)$.
$^{S}\lambda_{MM'}$	The short-range part of $\lambda_{MM'}(I)$; treated as a constant.
μ_{ijk}	Third-order interaction coefficient for the i-th, j-th, and k-th aqueous solute species.
ν_M	Number of cations M produced by dissociation of the aqueous neutral electrolyte MX.
ν_{MX}	Number of cations M and anions X produced by dissociation of the aqueous neutral electrolyte MX.
ν_X	Number of anions X produced by dissociation of the aqueous neutral electrolyte MX.
$\rho_{g/ml}$	Solution density, g/ml.
σ, σ'	Symbols denoting end-member components of a solid solution.
$\sigma_{T,\psi}$	Total number of end members in the ψ-th solid solution.
τ_s	Alkalinity factor, the number of hydrogen ion neutralizing equivalents per mole of the s-th aqueous species.
ϕ	(a) Subscript denoting a pure mineral; (b) the osmotic coefficient of the aqueous solution.
$\Phi_{MM'}(I)$	Harvie, Møller, and Weare's (1984) notation for $\theta_{MM'}(I)$.
$\Phi'_{MM'}(I)$	Harvie, Møller, and Weare's (1984) notation for $\theta'_{MM'}(I)$.
χ_g	Fugacity coefficient of the g-th gas.
ψ	Subscript denoting a solid solution.
ψ_T	Total number of solid solutions in a system.
$\Psi_{MM'X}$	Observable third order interaction coefficient for neutral electrolytes MX and $M'X$.
Ω	Water constant; 1000 divided by the molecular weight of water; about 55.51.
$\aleph_{ij}$	Ionic activity combination parameter; $\aleph_{ij} = \lvert z_j \rvert \log a_i - \frac{z_i z_j}{\lvert z_i \rvert} \log a_j$
+	Subscript denoting a reaction proceeding in the forward sense; the convention in this report equates this with dissociation, dissolution, or destruction of the associated species.

Subscript denoting a reaction proceeding in the backward sense; the convention in this report equates this with association, precipitation, or formation of the associated species.

EQ3NR, A Computer Program for Geochemical Aqueous Speciation-Solubility Calculations: Theoretical Manual, User's Guide, and Documentation (Version 7.0)

Abstract

EQ3NR is an aqueous solution speciation-solubility modeling code. It is part of the EQ3/6 software package for geochemical modeling. It computes the thermodynamic state of an aqueous solution by determining the distribution of chemical species, including simple ions, ion pairs, and complexes, using standard state thermodynamic data and various equations which describe the thermodynamic activity coefficients of these species. The input to the code describes the aqueous solution in terms of analytical data, including total (analytical) concentrations of dissolved components and such other parameters as the *pH*, *pHCl*, *Eh*, *pe*, and oxygen fugacity. The input may also include a desired electrical balancing adjustment and various constraints which impose equilibrium with specified pure minerals, solid solution end-member components (of specified mole fractions), and gases (of specified fugacities). The code evaluates the degree of disequilibrium in terms of the saturation index ($SI = \log Q/K$) and the thermodynamic affinity ($A = -2.303\ RT \log Q/K$) for various reactions, such as mineral dissolution or oxidation-reduction in the aqueous solution itself. Individual values of *Eh*, *pe*, oxygen fugacity, and *Ah* (redox affinity) are computed for aqueous redox couples. Equilibrium fugacities are computed for gas species. The code is highly flexible in dealing with various parameters as either model inputs or outputs. The user can specify modification or substitution of equilibrium constants at run time by using options on the **input** file. The output consists of an **output** file and a **pickup** file, which can be used to initialize an EQ6 reaction path calculation. The chief numerical method employed is a hybrid Newton-Raphson technique. This is supported by a set of algorithms which create and optimize starting values. EQ3NR reads a secondary unformatted data file (**data1**) that is created from a primary formatted data file (**data0**) by EQPT, the EQ3/6 data file preprocessor. There is currently a set of five data (**data0**) files. Three of these may be used with either the Davies equation or the B-dot equation to describe the activity coefficients of the aqueous species. Their use is restricted to modeling dilute solutions. The other two of these use Pitzer's equations and are suitable for modeling solutions to high concentrations, though with fewer chemical components. The temperature range of the thermodynamic data on the data files varies from 25°C only to 0-300°C. EQ3NR may be used by itself or to initialize a a reaction path calculation by EQ6, its companion code in the EQ3/6 package. EQ3NR and the other codes in the EQ3/6 package are written in FORTRAN 77 and have been developed to run under the UNIX operating system on computers ranging from workstations to supercomputers.

1. Introduction

EQ3NR is a speciation-solubility code for modeling the thermodynamic state of an aqueous solution. In essence, this involves a static calculation that is usually based on water chemistry an analysis. The purpose of such a calculation is usually to find the detailed distribution of chemical species and to assess the degree of equilibrium (or disequilibrium) pertaining to various reactions, usually those involving other phases. EQ3NR can not be used to directly model the

chemical evolution of such a water. However, it can be used to initialize such a calculation, which can be made by the companion code EQ6 (Wolery and Daveler, 1992).

EQ3NR is part of the EQ3/6 software package (see Wolery, 1992). This report describes EQ3NR in version 7.0 (version 3245.1090 in the old numbering system) of this package (see the EQ3/6 Package Overview and Installation Guide, Wolery, 1992). Other codes in the package include EQPT (Daveler and Wolery, 1992), a data file preprocessor, and EQ6 (Wolery and Daveler, 1992), a reaction path code. The relationship of the EQ3NR code to EQ6, EQPT, and the set of supporting thermodynamic data files is shown in Figure 1. This figure depicts the flow of information involving these codes. At present, there are five distinct data files, denoted by the suffixes **com**, **sup**, **nea**, **hmw**, and **pit**. These are provided in formatted ASCII and are called **data0** files. EQPT processes these one at a time (looking for a file named simply **data0**, though these files are normally stored under names which include the relevant suffixes) and writes a corresponding unformatted data file, which is called simply **data1**. These are also normally stored under names including the relevant suffixes. To run EQ3NR or EQ6, the user must provide one of these files, which is known to each code simply as **data1**.

The user must select which of the five data files is most appropriate to a given problem. Each data file corresponds to a general formalism for treating the activity coefficients of the aqueous species and contains the relevant activity coefficient data as well as standard state thermodynamic data. The activity coefficient formalisms currently built into EQ3/6 are discussed in Chapter 3. The **com**, **sup**, and **nea** data files are specific to a general extended Debye-Hückel formalism and can be used by EQ3NR and EQ6 with either the Davies (1962) equation or the B-dot equation (Helgeson, 1969). These equations are only valid in relatively dilute solutions. The **hmw** and **pit** data files are specific to the formalism proposed by Pitzer (1973, 1975) and can be used to model solutions extending to high concentrations. However, the scope of chemical components covered is smaller. The temperature limits on the data files also vary, from 25°C only to 0-300°C.

Some important data file characteristics are given in Table 1. The **com** (for "composite") data file is the largest of the three data files specific to the extended Debye-Hückel formalism. It is a product of Lawrence Livermore National Laboratory (LLNL) drawing on many data sources, including those on which the other four data files are based. The **sup** data file is based entirely on SUPCRT92 (Johnson, Oelkers, and Helgeson, 1992), a data base and program for dealing with thermodynamic data based on the work of Helgeson and Kirkham (1974ab, 1976), Helgeson et al. (1978), Tanger and Helgeson (1988), Shock and Helgeson (1988, 1989, 1990), Shock, Helgeson, and Sverjensky (1989), Johnson and Norton (1991), and Shock et al. (1992). The **nea** data file is based entirely on Grenthe et al. (1989, draft report), a product of the Data Bank of the Nuclear Energy Agency of the European Community. This report has recently been published as Grenthe et al. (1992)The **hmw** data file is based on Harvie, Møller, and Weare (1984). The **pit** data file is based mostly on data summarized by Pitzer (1979). All five data files are maintained at LLNL in a relational data base described by Delany and Lundeen (1991). This relational data base is part of the Yucca Mountain Site Characterization Project's Technical Data Base.

The **sup** data file has a high level of internal consistency among the standard state thermodynamic data. In addition, the temperature-pressure dependence of these data are represented by a suite of equations of state for minerals, gases, and aqueous species that are well established in the geochemical literature (see references noted above). This data file covers a wide range of

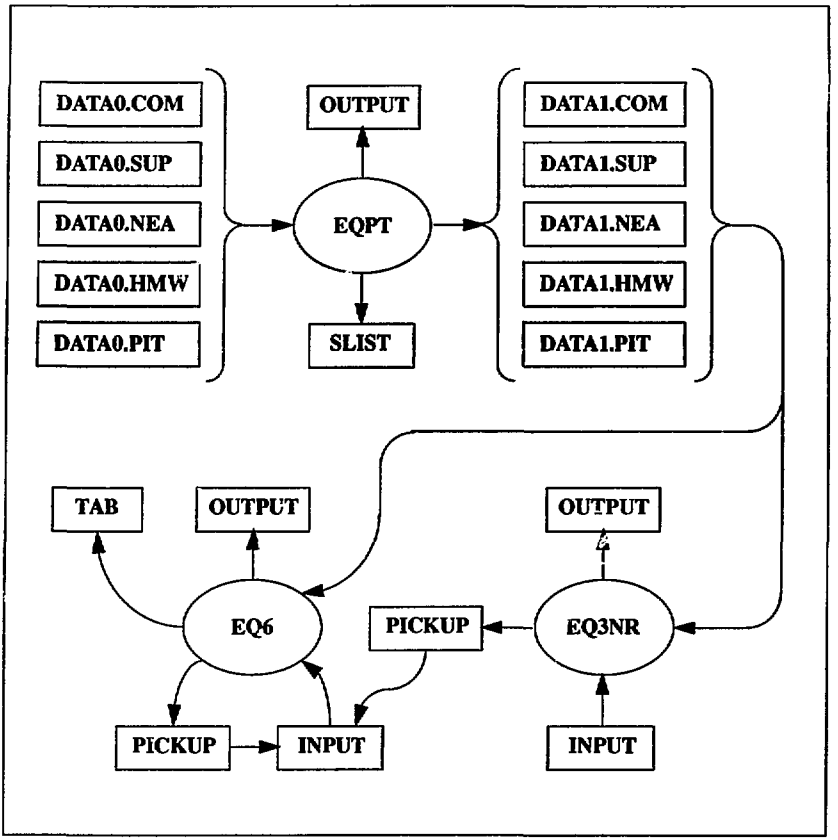

Figure 1. The flow of information among the computer codes EQPT, EQ3NR, and EQ6. Computer codes are represented by ovals, files by rectangles.

chemical elements and species of interest in the study of rock/water interactions (e.g., components which make up the major rock-forming and ore-forming minerals). It also includes a large number of organic species, mostly of small carbon number (C_2-C_8). The **nea** data file is something of a specialty item. Its strongest point is a thorough representation of the thermodynamics of uranium species.

The **com** (composite) data file encompasses a much broader range of chemical elements and species. It includes the data found on the **sup** and **nea** data files, with preference given to data from

Table 1. Major characteristics of the current five EQ3/6 data files ("R10" versions).

File Name (Suffix)	Source	Activity Coefficient Formalism	Temperature Limits	Number of Chemical Elements	Number of Basis Species	Number of Aqueous Species	Number of Pure Minerals	Number of Solid Solutions	Number of Gas Species
com	GEMBOCHS (LLNL)	Extended Debye-Hückel	0-300°C	78	147	852	886	12	76
sup	SUPCRT92	Extended Debye-Hückel	0-300°C	69	105	315	130	0	16
nea	NEA draft report	Extended Debye-Hückel	0-300°C	32	50	138	188	0	76
hmw	Harvie, Møller, and Weare (1984)	Pitzer's Equations	25°C only	9	13	17	51	0	3
pit	Pitzer (1979)	Pitzer's Equations	0-100°C	52	62	68	381	0	38

the former in cases of overlap. It also includes some data found in the **hmw** data file, as well as other data which do not appear in any of the other data files. Some of these data are estimates based on correlations or extrapolations (as to higher temperature), and are not tied directly to experimental measurements. The **com** data file thus represents a melange of data, which by its nature offers less assurance of internal consistency. However, this offers the only means presently available for modeling aqueous solutions with a high degree of compositional complexity, such as the fluids expected to be found in and about a facility for the geologic disposal of industrial or nuclear waste (e.g., the potential repository for high-level nuclear waste at Yucca Mountain, Nevada).

The **hmw** data file has the highest degree of internal consistency of any of the five data files, including mutual consistency of activity coefficient data and standard state thermodynamic data. It can be applied to dilute waters or concentrated brines. However, it only treats the set of components present in the "sea-salt" system (the major cations and anions present in seawater, including carbonate and bicarbonate). The geochemically important components aluminum and silica are not included. Also, this data file is limited to a temperature of 25°C. The **pit** data file can also be applied to concentrated brines. It covers a larger set of components, but these mostly involve other cations and anions of strong electrolytes. Examples include lithium and bromide. This data file nominally covers the temperature range of 0-100°C. However, it represents a melange of data, not a carefully crafted internally consistent set.

The data file preprocessor EQPT (Daveler and Wolery, 1992) performs a number of functions. It checks the composition, charge, and reaction coefficient data on a **data0** file for internal consistency and fits interpolating polynomials to various temperature dependent data which are organized on the **data0** file on temperature grids. Such data include certain aqueous species activity coefficient parameters, such as Debye-Hückel $A_{\gamma,10}$ and B_{γ}, and the equilibrium constants for the reactions represented on the data file. In addition, in the case of data files specific to the formalism of Pitzer's equations, observable interaction coefficients are mapped to a set of conventionally defined primitive interaction coefficients (see Chapter 3). EQPT then writes the **data1** file corresponding to the input **data0** file. For details of the contents and structure of **data0** and **data1** files, see Daveler and Wolery (1992). Run-time alteration of the values of selected equilibrium constants can subsequently be selected by the user on the EQ3NR **input** file (see Chapter 6). EQPT also writes to a screen file and an **output** file, both of which are generally significant only if an error condition is encountered. In addition, it writes an **slist** (species list) file. This is very useful to the user, as it lists the species that are represented on the data file and identifies which species are in the strict and auxiliary basis sets (See Chapter 5).

A speciation-solubility problem to be run with EQ3NR is described on the EQ3NR **input** file. This is the subject of Chapter 6 of this report. Examples are presented in Chapter 7. The code then produces an **output** file describing the results of the calculation. While the code is running, it writes to a screen file, primarily to apprise the user of what is happening. It also writes a **pickup** file, which contains a compact description of the aqueous solution (see Chapter 8). This is required for a subsequent EQ6 calculation; it corresponds to the bottom part of the EQ6 **input** file. It has no other real use. EQ6 in turn writes its own **output** file, as well as a **tab** file which contains certain data in tabular form suitable for supporting local graphics postprocessing. This code also writes to the screen file while it is running. In addition, EQ6 writes its own **pickup** file, which

may be used as an input file to restart a reaction path calculation from the point at which a previous run stopped.

The input to the code consists of a chemical analysis of a water and specification of various user-defined options. The input usually consists mostly of analytical values for concentrations of dissolved components. These represent total values that do not distinguish between contributions from simple ions, ion pairs, and aqueous complexes, species which may exist in solution in mutual equilibrium. In addition, analytical data may or may not distinguish a dissolved component by oxidation state. The *pH* is also normally an input parameter. A new alternative parameter called *pHCl* can be input in place of *pH* to overcome the liquid junction potential problem in measuring *pH* in concentrated solutions (see Chapter 2). The *Eh* (redox potential) is also a common input parameter, though its usage is somewhat problematical (see Chapter 2). One may specify the oxygen fugacity or *pe* instead, though this is no less problematical. It is also possible to specify a redox couple to define the redox state. For example, one might specify the ferrous-ferric couple if one had two total concentration values, one for Fe^{2+} and another for Fe^{3+}. It is best to treat as many couples as possible by this method. That way, redox equilibrium can be tested instead of merely assumed.

The basic input constraints (total concentrations, *pH*, etc.) are associated on a one-to-one basis with master or *basis* species. Basis species (see Chapters 2 and 5) represent the chemical components of the aqueous solution. They also function as basic elements for writing chemical reactions in a standardized format that is convenient for chemical modeling. The solvent, water, is a basis species, but is an exception in a speciation-solubility problem in that no input constraint is associated with it. The basis species used to write oxidation-reduction reactions in EQ3/6 is oxygen gas, which is treated as a fictive aqueous species. An input for it is required only if the problem has a redox aspect. The other basis species consist of simple species such as Na^+ and Cl^- and a few more complex species such as SO_4^{2-}. A minimum basis set has one species representing each chemical element and its associated mass balance, plus one more representing oxidation-reduction and charge balance. The minimum basis is called the *strict* basis. EQ3NR also has an *auxiliary* basis, which consists of species which are related via associated chemical reactions to the strict basis species, but for which the user may choose to impose constraints other than equilibrium with the latter. Most auxiliary basis species represent a chemical element in a different oxidation state.

If desired, the concentration of a specified ion may be adjusted to satisfy electrical balance. An option to constrain the carbonate system by specifying the alkalinity has been deleted from the present version of EQ3NR. The reasons behind this action and suggestions for alternative measures are discussed in Chapter 2. It is also possible to constrain various species by certain equilibrium assumptions instead of analytical data. For example, the concentration of dissolved calcium may be constrained to satisfy equilibrium with calcite. It is also possible to constrain the concentration of a species to satisfy equilibrium with a solid solution end-member component of specified mole fraction. Similarly, the concentration of a species may be constrained to satisfy equilibrium with a gas species of specified fugacity.

EQ3NR computes the distribution of chemical species present in the model. Essentially, this involves partitioning the input total concentrations. The code thus determines the concentrations,

activity coefficients, and thermodynamic activities of all species present. This in turn permits evaluation of the saturation indices ($SI = log\ Q/K$, where Q is the activity product and K the equilibrium constant) and thermodynamic affinities ($A = -2.303\ RT\ log\ Q/K$, where R is the gas constant and T the absolute temperature) of various reactions, chiefly for the dissolution of minerals. However, these functions are also evaluated for certain reactions occurring internally in the aqueous solution and which are normally only assumed to be in equilibrium (the **input** file requires additional data to do this). In the case of aqueous redox reactions, the theoretical *Eh*, *pe*, oxygen fugacity, and redox affinity (*Ah*) are computed. Differences in the values of these corresponding parameters for two redox couples are measures of the degree of disequilibrium between them. The equilibrium fugacities of various gas species are also determined.

The results of these calculations depend on the supporting data read from the data file. The use of different data files may give different results. Different results may be obtained not only because of the use of different values of standard state thermodynamic data (e.g., equilibrium constants), but also by different choices in the set of equations for the activity coefficients as well as the use of different values in the choice of parameters appearing in these equations (e.g., the Debye-Hückel A_ϕ parameter, various kinds of interaction coefficients). The equations for calculating the activity coefficients of aqueous species are discussed in Chapter 3. The equations for calculating the activity coefficients of end-member components of solid solution phases are discussed in Chapter 4. In speciation-solubility calculations, these latter equations and their supporting data normally affect only the saturation indices calculated for solid solutions. However, they do affect the computed aqueous speciation model if one of the defining model constraints assumes equilibrium with a solid solution end-member component.

In some modes, such as when the concentration of a species is adjusted to satisfy electrical balance or to satisfy an equilibrium constraint, the code actually computes part of what would normally be analytical data. In this mode, for example, the code can be used to calculate recipes for custom *pH* buffers. An example of this is included in Chapter 7. Calculations using such constraints can be somewhat dangerous, especially when used in combination. It is not hard to construct problems that have no physical solutions. In such cases, the code can of course compute no corresponding answers, but it does a generally good job of diagnosing the problems and informing the user of the nature of the problem.

In general, the code is highly flexible in that the roles of many parameters as inputs and outputs can be reversed. There are very few restrictions on the input combinations that may be defined by the code user. The main requirement is that the problem must have a realistic answer.

EQ3NR uses a highly efficient hybrid Newton-Raphson algorithm in which the activity coefficients of the aqueous species are held constant in a Newton-Raphson step and re-adjusted between such steps. The code features both user-controlled and automatic basis-switching, a procedure for rewriting reactions and redefining the set of basis species. These features are occasionally necessary to induce the iterative calculations to converge. The code creates its own starting estimates for Newton-Raphson iteration, and uses a first order algorithm in addition to possible automatic basis switching to optimize these before beginning Newton-Raphson iteration. The numerical methods used by the code are discussed in Chapter 9.

EQ3NR performs a number of tests on the model constraints to see if they make sense. It first checks the data and options read from the **input** file for inconsistent or incomplete combinations. It will write informative error messages and terminate any further action if it detects bad input. However, not all bad input can be detected at this stage. Further analysis takes place when the code chooses starting estimates for the master iteration variables. Finally, if Newton-Raphson iteration fails to converge, EQ3NR will analyze the results to generate crash diagnostics. Most of these will point to bad input, usually input that is bad in more subtle ways than those which would have been flagged earlier.

The code architecture is described in Chapter 10. Appendix A contains a glossary of the major code variables. The source code modules are listed and briefly described in Appendix B (for a similar treatment of EQLIB modules, see Appendix B of the EQ3/6 Package Overview and Installation Guide, Wolery, 1992). Appendix C contains a list of error messages generated by EQ3NR modules, along with related notes (see Appendix C of Wolery, 1992, for a similar list for EQLIB modules). Appendix D contains notes pertaining to known bugs and such.

EQ3NR and the other codes in the EQ3/6 software package are written in FORTRAN 77 and have been developed to run under UNIX operating systems on computers ranging from workstations to supercomputers, including Sun SPARCstations, VAXes (ULTRIX operating system), Alliants (CONCENTRIX operating system), and Crays (UNICOS operating system). They are fairly readily portable to VAX computers running the non-UNIX VMS operating system. They may be portable as well to 386 and 486 PCs. Platforms used at LLNL include Sun SPARCstations and an Alliant FX/80. For details concerning platforms, see the EQ3/6 Package Overview and Installation Guide (Wolery, 1992).

activity coefficients, and thermodynamic activities of all species present. This in turn permits evaluation of the saturation indices ($SI = log\ Q/K$, where Q is the activity product and K the equilibrium constant) and thermodynamic affinities ($A = -2.303\ RT\ log\ Q/K$, where R is the gas constant and T the absolute temperature) of various reactions, chiefly for the dissolution of minerals. However, these functions are also evaluated for certain reactions occurring internally in the aqueous solution and which are normally only assumed to be in equilibrium (the **input** file requires additional data to do this). In the case of aqueous redox reactions, the theoretical *Eh*, *pe*, oxygen fugacity, and redox affinity (*Ah*) are computed. Differences in the values of these corresponding parameters for two redox couples are measures of the degree of disequilibrium between them. The equilibrium fugacities of various gas species are also determined.

The results of these calculations depend on the supporting data read from the data file. The use of different data files may give different results. Different results may be obtained not only because of the use of different values of standard state thermodynamic data (e.g., equilibrium constants), but also by different choices in the set of equations for the activity coefficients as well as the use of different values in the choice of parameters appearing in these equations (e.g., the Debye-Hückel A_ϕ parameter, various kinds of interaction coefficients). The equations for calculating the activity coefficients of aqueous species are discussed in Chapter 3. The equations for calculating the activity coefficients of end-member components of solid solution phases are discussed in Chapter 4. In speciation-solubility calculations, these latter equations and their supporting data normally affect only the saturation indices calculated for solid solutions. However, they do affect the computed aqueous speciation model if one of the defining model constraints assumes equilibrium with a solid solution end-member component.

In some modes, such as when the concentration of a species is adjusted to satisfy electrical balance or to satisfy an equilibrium constraint, the code actually computes part of what would normally be analytical data. In this mode, for example, the code can be used to calculate recipes for custom *pH* buffers. An example of this is included in Chapter 7. Calculations using such constraints can be somewhat dangerous, especially when used in combination. It is not hard to construct problems that have no physical solutions. In such cases, the code can of course compute no corresponding answers, but it does a generally good job of diagnosing the problems and informing the user of the nature of the problem.

In general, the code is highly flexible in that the roles of many parameters as inputs and outputs can be reversed. There are very few restrictions on the input combinations that may be defined by the code user. The main requirement is that the problem must have a realistic answer.

EQ3NR uses a highly efficient hybrid Newton-Raphson algorithm in which the activity coefficients of the aqueous species are held constant in a Newton-Raphson step and re-adjusted between such steps. The code features both user-controlled and automatic basis-switching, a procedure for rewriting reactions and redefining the set of basis species. These features are occasionally necessary to induce the iterative calculations to converge. The code creates its own starting estimates for Newton-Raphson iteration, and uses a first order algorithm in addition to possible automatic basis switching to optimize these before beginning Newton-Raphson iteration. The numerical methods used by the code are discussed in Chapter 9.

EQ3NR performs a number of tests on the model constraints to see if they make sense. It first checks the data and options read from the **input** file for inconsistent or incomplete combinations. It will write informative error messages and terminate any further action if it detects bad input. However, not all bad input can be detected at this stage. Further analysis takes place when the code chooses starting estimates for the master iteration variables. Finally, if Newton-Raphson iteration fails to converge, EQ3NR will analyze the results to generate crash diagnostics. Most of these will point to bad input, usually input that is bad in more subtle ways than those which would have been flagged earlier.

The code architecture is described in Chapter 10. Appendix A contains a glossary of the major code variables. The source code modules are listed and briefly described in Appendix B (for a similar treatment of EQLIB modules, see Appendix B of the EQ3/6 Package Overview and Installation Guide, Wolery, 1992). Appendix C contains a list of error messages generated by EQ3NR modules, along with related notes (see Appendix C of Wolery, 1992, for a similar list for EQLIB modules). Appendix D contains notes pertaining to known bugs and such.

EQ3NR and the other codes in the EQ3/6 software package are written in FORTRAN 77 and have been developed to run under UNIX operating systems on computers ranging from workstations to supercomputers, including Sun SPARCstations, VAXes (ULTRIX operating system), Alliants (CONCENTRIX operating system), and Crays (UNICOS operating system). They are fairly readily portable to VAX computers running the non-UNIX VMS operating system. They may be portable as well to 386 and 486 PCs. Platforms used at LLNL include Sun SPARCstations and an Alliant FX/80. For details concerning platforms, see the EQ3/6 Package Overview and Installation Guide (Wolery, 1992).

2. Speciation-Solubility Modeling of Aqueous Systems

2.1. Introduction

EQ3NR is a speciation-solubility code for aqueous systems. As such, given sufficient data on a specific aqueous system, it computes a model of the solution which consists of two principal parts: the distribution of species in the solution and a set of saturation indices ($SI = log\ Q/K$) for various reactions of interest. The saturation indices are measures of the degree of disequilibrium of the corresponding reactions. They provide a means of searching for solubility controls on natural waters. For example, if a series of related fluids all have calcite *SI* values close to zero, it is probable that this mineral is present and partial equilibrium with it is maintained as the solutions evolve in composition.

EQ3NR is not a computerized geochemical model, but a code which is capable of evaluating geochemical models which are defined by the contents of a supporting data file (of which there are now five to choose from) and by other assumptions which the user sets on the EQ3NR **input** file. The supporting data files differ not only in terms of data values, but more importantly in terms of the identities of the components and chemical species represented and in terms of the general approaches to dealing with the problem of activity coefficients. Because of various limitations, some problems may require the use of only certain data files, while others can be treated using any of the available data files. The user must choose the best data file (or files) with which to run a particular problem. The user must also understand both the particular problem and the code capabilities and limitations well enough to construct an adequate **input** file.

Although speciation-solubility models are commonly used as a means of testing whether or not heterogeneous reactions are in a state of thermodynamic equilibrium, they often just assume that all reactions occurring in aqueous solution are in such a state. Such reactions most likely to be in disequilibrium are redox reactions or reactions for the formation or dissociation of large complexes that are more like small polymers, such as $(UO_2)_3(OH)_7^-$. Speciation-solubility models are better used when they are employed to test the degree of disequilibrium of these kinds of reactions than when they are forced to assume that such reactions are in equilibrium.

A speciation-solubility model can not by itself predict how aqueous solution composition will change in response to rock/water interactions. Nevertheless, this type of modeling can be a powerful tool for elucidating such interactions when it is applied to a family of related waters. Such a family might be a set of spring waters issuing from the same geologic formation, a sequence of ground water samples taken from along an underground flow path, or a sequence of water samples taken in the course of a rock/water interactions experiment in the laboratory. Jenne (1981) reviews several studies of this kind. Particularly interesting are Nordstrom and Jenne's (1977) study of fluorite solubility equilibria in geothermal waters and Nordstrom, Jenne, and Ball's (1979) study of controls on the concentration of iron in acid mine waters.

EQ3NR offers many options for the **input** file description of the composition of a given water. Consequently, the code can be used in a variety of ways. Many of the descriptive parameters of interest can be either model inputs or outputs. For example, the *pH* of a buffer solution can be calculated from the buffer recipe by adjusting the hydrogen ion concentration to satisfy charge balance. Alternatively, adjusting the concentration of a buffer component to satisfy the charge

balance is a means of computing the complete recipe for a buffer having a desired *pH*. Some of the possible model inputs are assumptions, as of equilibrium with specified minerals. The use of some types of model inputs also pose special problems, some of which occur in particular contexts. The worst of these pertain to *Eh*, alkalinity, and *pH* and will be discussed in some detail later in this chapter.

2.2. Units of Concentration

EQ3NR uses the molal scale as the principal unit of concentration for aqueous species. The molal concentration (molality) of a substance dissolved in water is defined as:

$$m_i = \frac{n_i}{w_w} \tag{1}$$

where n_i is the number of moles of the i-th solute species and w_w is the number of kilograms of solvent water. Other common measures of aqueous solute concentration are the molarity (moles of substance per liter of aqueous solution), the part-per-million or ppm by volume (mg/L, milligrams of substance per liter of solution), and the ppm by weight (mg/kg, milligrams of substance per kilogram of solution). The EQ3NR code accepts concentration parameters in any of these units (see Chapter 4), but converts non-molal concentrations to molalities before computing the aqueous speciation model. Whether or not it does this correctly depends on circumstance and data provided by the user.

The conversion equations in all three cases require a value for the total dissolved salts in mg/kg solution ($C_{TS,\,mg/kg}$). The density of the aqueous solution in g/ml ($\rho_{g/ml}$) is also required to convert molarities and mg/L concentrations to molalities. The total dissolved salts in mg/kg may be calculated from the total dissolved salts in mg/L ($C_{TS,\,mg/L}$) and the density according to:

$$C_{TS,\,mg/kg} = \frac{C_{TS,\,mg/L}}{\rho_{g/ml}} \tag{2}$$

EQ3NR expects values of $C_{TS,\,mg/kg}$ and $\rho_{g/ml}$ on the **input** file if such conversions are necessary (see Chapter 4). In place of $C_{TS,\,mg/kg}$, one may enter $C_{TS,\,mg/L}$ and $\rho_{g/ml}$, and $C_{TS,\,mg/kg}$ is calculated from the above equation. If such values are not provided, $C_{TS,\,mg/kg}$ is assigned a default value of zero and $\rho_{g/ml}$ is assigned a default value of unity. These values are generally adequate for dilute solutions at temperatures near 25°C. In the case of brines, these values are not adequate, and the user must provide actual values as part of the input in order to obtain accurate conversion. *The code provides no checks or warnings if these are not provided.*

The weight fraction of solvent water is given by:

$$N_w = \frac{1,000,000 - C_{TS,\,mg/kg}}{1,000,000} \tag{3}$$

Letting $C_{i,\,molar}$ be the molar concentration of the i-th solute species, the molality is given by

$$m_i = \frac{C_{i,molar}}{\rho_{g/ml} N_w} \tag{4}$$

Letting $C_{i,mg/L}$ be the concentration in mg/L, the conversion is:

$$m_i = \frac{0.001 C_{i,mg/L}}{\rho_{g/ml} M_w N_w} \tag{5}$$

where M_w is the molecular weight of the solvent, water ($M_w \approx 18.015$ g/mole). Letting $C_{i,mg/kg}$ be the concentration in mg/kg solution, the conversion is:

$$m_i = \frac{0.001 C_{i,mg/L}}{M_w N_w} \tag{6}$$

Some dissolved gas analyses are reported in units of ml (STP)/ml solution, where STP refers to standard temperature and pressure (0°C and 1 atm). The conversion equation is:

$$m_i = \frac{1000 C_{i,ml(STP)/ml}}{V_g M_w N_w} \tag{7}$$

where V_g is the molar volume of an ideal gas at STP ($V_g = 22{,}413.6$ ml/mole).

The concentration of solvent water is defined as its mole fraction:

$$x_w = \frac{n_w}{n_w + \sum_i n_i} \tag{8}$$

where n_w is the number of moles of water. The molality of the i-th solute species can also be written as:

$$m_i = \frac{\Omega n_i}{n_w} \tag{9}$$

where Ω is the number of moles of water comprising a mass of 1 kg ($\Omega \approx 55.51$; $\Omega w_w = n_w$). Substituting this relation into the one above it gives:

$$x_w = \frac{\Omega}{\Omega + \sum_i m_i} \tag{10}$$

EQ3NR uses this relation to calculate the mole fraction of water. This is done in a self-consistent manner in the iteration process. Thus, the user is not required to input a value.

A similar self-consistent treatment could be implemented to handle both $C_{TS,\ mg/kg}$ and the solution density. However, no such treatment exists in the current version of EQ3NR, nor any other such modeling code known to the present writers. Implementation of a self-consistent treatment of the solution density would require the addition of models for partial molar volumes to the code and incorporation of the corresponding equations in the iteration process. The theoretical and practical aspects of partial molar volumes in solutions extending to high concentration have been addressed for example by Millero (1977), Helgeson, Kirkham, and Flowers (1981), Kumar (1986), Connaughton, Millero, and Pitzer (1989), and Monnin (1989).

2.3. Input Constraints, Governing Equations, and Outputs

2.3.1. Overview

Aqueous speciation models can be constructed to satisfy a wide variety of combinations of possible input constraints and governing equations. The input constraints may include total (analytical) concentrations, an electrical balance requirement, free concentrations, activities, *pH*, *Eh*, *pe*, oxygen fugacity, phase equilibrium requirements, homogeneous equilibria, and run-specific values for equilibrium constants. The governing equations are the corresponding mathematical expressions. such as the mass balance equation and the charge balance equation.

The choice of governing equations in large part depends on which parameters are to be inputs to the model and which are to be outputs. This, in turn, is a function of what data on a given water are available, what form they are in, and what assumptions the modeler would like to use.

Chemical analysis provides mainly a set of values for the so-called total concentrations of dissolved components. The analytical value for an ion such as calcium is an example. It does not discriminate between the various calcium species in solution, but rather estimates the dissolved calcium contributed by all of them. This leads to a mass balance equation of the form:

$$m_{T,\,Ca^{2+}} = m_{Ca^{2+}} + m_{CaOH_{(aq)}} + m_{CaCO_{3\,(aq)}} + m_{CaHCO_3^+} + \ldots \qquad (11)$$

where $m_{T,\,Ca^{2+}}$ is the total or analytical concentration (on the molal scale) and m_i is the molality of any individual chemical species contributing to the mass balance. The summations must be weighted by the appropriate stoichiometric equivalences; e.g., in the case of F^-, one has:

$$m_{T,\,F^-} = m_{F^-} + m_{HF_{(aq)}} + 2m_{H_2F_{2\,(aq)}} + 2m_{HF_2^-} + 3m_{AlF_{3\,(aq)}} + \ldots \qquad (12)$$

The total concentration is the most common type of input parameter to an aqueous speciation model. The mass balance constraint, which corresponds to it, is therefore the most common governing equation. As we shall see, there are situations in which a total concentration is replaced by another type of input. In these cases, the mass balance constraint is replaced by a different governing equation, and the total concentration becomes something to be calculated (an output parameter).

From a purely mathematical point of view, there is no reason to discriminate among ion pairs (and ion-triplets, etc.) and complexes. For some investigators, the term "ion pair" implies a spe-

cies in which an anion is separated from a cation by an unbroken hydration sheath about the latter, whereas the term "complex" implies direct contact and perhaps some degree of covalent bonding. Other investigators use these terms interchangeably. It is a general assumption in cases of geochemical interest that the concentrations of ion-pairs and complexes are governed by thermodynamic equilibrium.

Each case of this equilibrium can be represented by a mass-action equation for the dissociation of the ion-pair or complex. An example will illustrate this. The calcium sulfate ion-pair dissociates according to the reaction:

$$CaSO_{4(aq)} = Ca^{2+} + SO_4^{2-} \tag{13}$$

where "=" is used as the sign for a reversible chemical reaction. The corresponding mass action equation is:

$$K_{CaSO_{4(aq)}} = \frac{a_{Ca^{2+}} a_{SO_4^{2-}}}{a_{CaSO_{4(aq)}}} \tag{14}$$

where K is the equilibrium constant and a_i represents the thermodynamic activity of each species. This may also be written in logarithmic form:

$$\log K_{CaSO_{4(aq)}} = \log a_{Ca^{2+}} + \log a_{SO_4^{2-}} - \log a_{CaSO_{4(aq)}} \tag{15}$$

The thermodynamic activity is related to the molal concentration by the relation:

$$a_i = m_i \gamma_i \tag{16}$$

where γ_i is the activity coefficient, a function of the composition of the aqueous solution. As the solution approaches infinite dilution, the value of γ_i for each species approaches unity. The set of equations for computing the activity coefficients of aqueous species is chosen by the user on the EQ3/6 **input** file (by means of the **iopg1** option switch). The requisite supporting data are on the EQ3/6 data file. The various formulations presently treated by EQ3/6 are discussed in Chapter 3.

The following subsections discuss the formulation of aqueous speciation problems in general terms. The rigorous mathematical development is presented in Chapter 9. How to implement these models in EQ3NR is the subject of Chapter 6, and examples are presented in Chapter 7.

2.3.2. Reference Formulation of the Aqueous Speciation Problem

In general terms, setting up an aqueous speciation model involves choosing n unknowns and n governing equations. The EQ3NR code offers a very wide range of options in this regard. In order to make sense of the different ways of setting up a model, we define a reference formulation for the aqueous speciation problem. This reference formulation serves as a springboard for discussing what goes into speciation models, what comes out, and what the options are. It is also used to compare how the aqueous speciation problem is formulated in EQ3NR (and other speciation-solubility codes in general) with how it is formulated in a reaction-path code like EQ6.

In the reference formulation, we assume that the activity coefficients are known parameters (the numerical treatment of these is discussed in Chapter 9). Note that the molal concentration of the solvent is fixed as the number of moles of water in a one kilogram mass of the pure substance.

We assume that there are ε_T chemical elements in the model. In order to further simplify the reference formulation, we assume that each element is present in only one oxidation state. Suppose that chemical analysis has given us ε_T - 2 total concentration values, each for a chemical species, each of which corresponds to a chemical element other than oxygen and hydrogen (e.g., Na^+ for Na, SO_4^{2-} for S). That gives ε_T - 2 mass balance equations as governing equations.

The charge balance equation plays the role that might have been played by a mass balance equation for hydrogen. The charge balance equation may be written in the general form:

$$\sum_{s=1}^{s_T} z_s m_s = 0 \tag{17}$$

where the summation is over all aqueous species, z_s is the electrical charge of a species, and m_s is its molal concentration. The hydrogen mass balance equation can not be used as a governing equation to calculate the pH from the corresponding analytical data. This is due to the impracticability if not impossibility of ever measuring the total concentration of hydrogen with sufficient accuracy when nearly all of it is contributed by the solvent. As a practical matter, even the charge balance equation can be used for this purpose only in limited circumstances.

One may associate the solvent, water, with a mass balance for oxygen. However, the mass of water in a speciation-solubility calculation is fixed at 1 kg, and the concentration of water is entirely determined by the concentrations of the other components in the solution. Therefore, no such mass balance is required.

To sum up, the reference formulation consists of ε_T - 2 mass balance equations/total concentrations (one pair for every element except oxygen and hydrogen) and the charge balance equation (to calculate pH). Each element is present in only one oxidation state. Activity coefficients are treated as known parameters.

Before proceeding, we contrast this framework (common to speciation-solubility codes in general) with that employed in the EQ6 code. In the corresponding problem in that code, we would be given ε_T masses, in moles, and the same number of mass balance equations, this time written in terms of masses instead of concentrations. There we have a mass balance equation for oxygen, and we must calculate the mass of the solvent, water. In the case where each element appears in only one oxidation state, as we have temporarily assumed here, the charge balance equation is a linear combination of the mass balance equations, and the governing equation associated with H^+ can be either a hydrogen mass balance equation or the charge balance equation. The speciation-solubility problem has one fewer unknown, hence one less governing equation, than the corresponding EQ6 problem.

In either the EQ3NR or EQ6 type formulation of the problem, we may formally associate one aqueous species with each balance equation; e.g., Na^+ with sodium balance, Al^{3+} with aluminum balance, and H^+ with charge balance. Suppose our model must consider n balance equations and k aqueous complexes (using the term to include ion-pairs). That gives k mass action relationships which are also governing equations. We now have $n + k$ equations in $n + k$ unknowns (the masses/concentrations/activities of the $n + k$ aqueous species).

The number of aqueous complexes is usually much greater than the number of balance equations. This is especially true when the number of balance equations becomes very large. A useful approach is to reduce the number of equations and unknowns by substituting the aqueous mass action equations into the balance equations (see Chapter 9). This leaves us with n equations (modified balance equations) in n unknowns (the concentrations or activities of the aqueous species that were chosen to formally correspond to the balance relationships).

This approach leads us to the concept of dealing with a set of master aqueous species. These may also be termed basis species. However, the concept does not arise purely from an attempt to reduce the number of iteration variables. The k aqueous complexes give us k linearly independent dissociation reactions and k linearly independent logarithmic mass action equations. An efficient way to write these reactions and equations is in terms of the associated complex (the species that dissociates) and such a set of master aqueous species. The dissociation reactions are then written as overall dissociation reactions but never as stepwise reactions; e.g., one has:

$$HgCl_3^- = Hg^{2+} + 3Cl^- \tag{18}$$

not:

$$HgCl_3^- = HgCl_2^o + Cl^- \tag{19}$$

We will also use this format to write dissolution reactions for minerals and gases and their associated heterogeneous mass action equations.

2.3.3. Alternative Constraints

The reference formulation of the aqueous speciation problem consists of:

(1) ε_T - 2 mass balance equations/total concentrations.

(2) the charge balance equation (to calculate pH).

We now discuss alternative constraints to the balance equations in the reference formulation. We discuss how to put oxidation-reduction problems into the formulation in the following subsection.

The alternative constraints are:

- Specifying log activity for a species (recall $pH = -\log a_{H^+}$).

- Log activity combination functions (e.g., *pHCl*; Section 2.3.4).
- Log mean activity of an ion and one of opposite charge (Section 2.3.4).
- Applying the charge balance constraint to a master species other than H^+.
- Phase equilibrium with a pure mineral.
- Phase equilibrium with an end member of a solid solution (the composition of the solid solution must be specified).
- Phase equilibrium with a gas (the fugacity of the gas must be specified).
- Equilibrium with other aqueous species, without falling under a mass balance constraint.
- Specifying the individual concentration of an aqueous basis species.

When a mass balance constraint is replaced by one of the above, we continue to reduce the number of unknowns to a master set as discussed above. The corresponding total concentrations become parameters to be calculated. We can calculate, for example, the total mass/concentration of hydrogen. This can be done with sufficient relative accuracy to permit the EQ6 code to use it as a constraint to solve for *pH*.

The log activity constraint. The first substitution that we discuss is most often applied to the hydrogen ion. In the course of chemical analysis, the *pH* of an aqueous solution is usually determined by means of a specific-ion electrode. This gives us the activity of the hydrogen ion from the relation:

$$pH = -\log a_{H^+} \tag{20}$$

The activities of many other species, including Na^+, Ca^{2+}, S^{2-}, F^-, and Cl^-, to name but a few, may also be measured by specific-ion electrodes.

EQ3NR will accept as an input the logarithm of the activity of a species. Note that this means that the code expects to see *-pH*, not *pH*, on the **input** file when this option is invoked. The new governing equation is just:

$$m_i = \frac{a_i}{\gamma_i} \tag{21}$$

The charge balance constraint. This can be applied to one of the major ions if a charge-balanced speciation model is desired. If EQ3NR does not use the charge balance equation as a constraint, it will calculate the charge imbalance. Otherwise, it will notify the user of the change in total concentration or *pH* that was required to generate a charge-balanced model.

We recommend routinely calculating *pH* from electrical balance only in cases of synthetic solutions for which the ionic totals are exact with respect to charge balance. Such solutions are most likely to be *pH* buffer solutions. In other circumstances, this practice is potentially dangerous be-

cause the result is affected by the error in every analytical value that is put in the model and also by every analytical value that should have been put in the model but was not. In general, apart from the case of *pH* buffer solutions, it is only safe to calculate *pH* this way if the *pH* is low (high concentrations of H^+) or high (high concentrations of OH^-).

Equilibrium constraint involving a non-aqueous species. A mass balance constraint may also be replaced by an equilibrium constraint involving a specified pure mineral, solid solution component species, or gas species. Suppose we wanted to know what concentration of dissolved calcium would be required for a water to be in equilibrium with calcite (the stable polymorph of $CaCO_{3(c)}$ at 25°C). The dissolution reaction may be written as:

$$Calcite + H^+ = Ca^{2+} + HCO_3^- \tag{22}$$

and the corresponding governing equation is then:

$$K_{Calcite} = \frac{a_{Ca^{2+}} a_{HCO_3^-}}{a_{Calcite} a_{H^+}} \tag{23}$$

Because calcite is a pure phase, its activity is fixed at unity.

If the required equilibrium involves an end-member component of a solid solution, the governing equation is slightly modified. Suppose we choose equilibrium with a calcite end-member of a high-magnesium calcite $(Ca,Mg)CO_{3(c)}$. The governing equation has the same form as above, but the activity of the calcite end-member is no longer unity. Instead, one has:

$$a_{Calcite} = \lambda_{Calcite} x_{Calcite} \tag{24}$$

where $\lambda_{Calcite}$ is the activity coefficient and $x_{Calcite}$ is the mole fraction of the calcite component. The mole fraction of the σ-th component of the ψ-th solid solution is given by:

$$x_{\sigma\psi} = \frac{n_{\sigma\psi}}{\sum_{\sigma'}^{\sigma_{T,\psi}} n_{\sigma'\psi}} \tag{25}$$

where $n_{\sigma'\psi}$ is the number of moles of the σ'-th component and $\sigma_{T,\psi}$ is the number of such components. The current version of EQ3NR deals only with solid solutions that are composed of end-member components. The activity coefficients ($\lambda_{\sigma\psi}$) may be computed from a variety of equations. The activity coefficient model for a given solid solution is specified on the EQ3/6 data file, which also contains the requisite supporting parameters. The formulations presently treated in EQ3/6 are discussed in Chapter 4.

Suppose we would like to know how much dissolved carbonate would be in solution if it were in equilibrium with $CO_{2(g)}$. The $CO_{2(g)}$ dissolution reaction may be written as:

$$CO_{2(g)} + H_2O_{(l)} = H^+ + HCO_3^- \quad (26)$$

The corresponding governing equation is:

$$K_{CO_{2(g)}} = \frac{a_{H^+} a_{HCO_3^-}}{f_{CO_2} a_w} \quad (27)$$

Here f_{CO_2} is the fugacity of CO_2. In order to use this option, the user must provide an input value for it to the speciation model.

Fugacity is a thermodynamic variable for gases that is akin to partial pressure in the same way the thermodynamic activity of an aqueous species is akin to the molal concentration. The formal relationship is given by:

$$f_g = \chi_g p_g \quad (28)$$

where p_g is the partial pressure and χ_g i the fugacity coefficient of the g-th gas. The fugacity coefficient is analogous to the activity coefficient. At low pressures, it approaches unity and hence the fugacity approaches the partial pressure.

Specifying heterogeneous equilibria as inputs to an aqueous speciation model can be a *bit* dangerous. First, the user must choose which phases, stable or metastable, are controlling solubility equilibria. If a choice is an extremely poor one, the equilibrium concentration of a species so constrained may be very large. Furthermore, the expressions for the logarithm of the ion activity products for all such relations must be a linearly independent set in the corresponding aqueous species. (A corollary to this is that one may not constrain more than one species by the same heterogeneous equilibrium.) Such linear dependence violates the "apparent" or "mineralogic" phase rule (Wolery, 1979). This is slightly more restrictive than the phase rule of thermodynamics. Sets of equilibria that satisfy the phase rule, but only because the temperature and pressure happen to fall on a univariant curve, do not satisfy the apparent phase rule.

Equilibrium constraint involving an aqueous species. It is possible to specify equilibrium with other species in a manner in which the species so constrained does not fall under any mass balance constraints. As an example, one might treat dissolved sulfide (represented by HS^-) in this manner, computing it on the basis of equilibrium with sulfate and oxygen gas. The reaction relating sulfide to sulfate is:

$$HS^- + 2O_{2(g)} = H^+ + SO_4^{2-} \quad (29)$$

The governing equation is the corresponding mass action equation:

$$K_{HS^-} = \frac{a_{H^+} a_{SO_4^{2-}}}{a_{HS^-} f_{O_2}^2} \quad (30)$$

The sulfide component (HS^- and related species such as $H_2S_{(aq)}$) does not count in the mass balance defined for sulfate. This option is similar to those involving specifying various heterogenous equilibria.

Direct specification of individual molality. EQ3NR allows input of the individual concentrations of master species. The governing equation in this case is just the identity:

$$m_i = m_i \tag{31}$$

It is largely appropriate only for master species that form no complexes, such as $O_{2(aq)}$ and other dissolved gases.

2.3.4. pH in Brines: pHCl and Related Functions as Alternative Constraints

Using standard methods (e.g., Bates, 1964), the *pH* is measured using an ion-specific electrode for the hydrogen ion in combination with a standard reference electrode (usually silver-silver chloride). The electrode pair (commonly marketed as a combination electrode) is calibrated when used by immersion in at least two standard solutions whose *pH* values bracket the expected sample values. This method is appropriate in dilute solutions, but not in brines. The problem is the presence of a liquid junction potential in the reference electrode at the interface between the standard or sample solution and an internal solution composed of concentrated potassium chloride. The idea behind the standard measurement is that for sufficiently dilute sample or standard solutions, the liquid junction potential will remain at an essentially constant value (which can be factored out in the calibration process). As proposed by Bates (1964), the method should be restricted to solutions of ionic strength no greater than 0.1 molal. However, it is routinely applied to more concentrated solutions, such as seawater (for which the ionic strength is nearly 0.7 molal).

The standard method fails when applied to brines because the liquid junction potential obtained with the sample is significantly changed from that obtained with the relatively dilute calibration buffers. The theory describing liquid junction potentials has been reviewed by Baes and Mesmer (1976). In general, the dependence of the liquid junction potential on the sample solution composition is complex and can not be solely related to the ionic strength. Thus, one can not simply make *pH* measurements in the usual way using concentrated calibration standards whose ionic strengths match those of the samples. Furthermore, the theory consists of an ideal and a non-ideal part. Taking only the ideal part and making some approximations leads to the Henderson equation. This has occasionally been put forth as a means of correcting *pH* values in concentrated solutions obtained by going through the mechanics of the standard method. This approach is highly dubious.

Recently Knauss, Wolery, and Jackson (1990, 1991) have proposed a method to quantify *pH* in concentrated solutions which avoids the liquid junction potential problem by eliminating the standard reference electrode. In this method, this electrode is replaced by another specific ion electrode. If this is a chloride electrode, what one measures is *pHCl*, which is the sum of *pH* and *pCl*. As an input to a speciation-solubility code, this is just as adequate as the *pH* as long as there is a separate measurement of dissolved chloride to also input. This maintains a system of *n* equations in *n* unknowns. The code is able to separate *pH* from *pCl* using an activity coefficient model

for the dissolved species and a chosen *pH* scale. The subject of aqueous species activity coefficients and *pH* scales is addressed in Chapter 3.

Knauss, Wolery, and Jackson (1990) used EQ3NR to compute the *pHCl* and related functions corresponding to different combinations of specific-ion electrodes of various test solutions, such as 0.01 molal HCl with varying concentrations of NaCl. Pitzer's equations were used to compute the activity coefficients in these solutions, using mostly the model of Harvie, Møller, and Weare (1984) and sometimes an alternative data set given by Pitzer (1979). They then measured the corresponding electrical potentials and plotted them against the computed *pHCl* or other function. In most cases, excellent Nernstian responses were obtained, in essence identical to those one would obtain examining the standard *pH* method. This indicated that such solutions could be defined as calibration buffers. Of critical importance to constraining the *pH* in concentrated solutions was the fact that no interference due to sodium was found in the case of the hydrogen ion electrode, even in solutions with very low hydrogen ion concentration and very high sodium ion concentration.

The only observed failures of the method involved cases in which a specific-ion electrode responded to an ion other than the one to which it was supposed to respond. The chloride electrode was found to respond to bromide, for example. In solutions containing both bromide and chloride, however, *pHBr* could be measured without interference by using a bromide electrode. Interferences of this type were no surprise and are in fact well known from the use of the specific ion electrodes in dilute solutions, where they are paired with a standard reference electrode.

The method appears to work, but should receive more study. There are no official recommendations or standards concerning this method, such as those which the National Institute of Standards and Technology (formerly the National Bureau of Standards) has promulgated in the case of the standard *pH* measurement technique. One must currently make up one's own calibration buffers, which ideally should closely resemble the samples. The method has been criticized by Mesmer (1991), who prefers not to obtain *pH* by a method which requires the use of a model for the activity coefficients in the solution. He proposes alternative approaches which involve measuring the concentration of the hydrogen ion. These in turn are criticized by Knauss, Wolery, and Jackson (1991).

Values of *pHCl* and related functions such as *pHBr* and *pH/Na* (= *pH - pNa*) can now be input to EQ3NR as alternative constraints. In the case of *pHCl*, the governing equation takes the form:

$$log m_{H^+} = -pHCl - log\gamma_{H^+} - log m_{Cl^-} - log\gamma_{Cl^-} \tag{32}$$

EQ3NR expects to receive input of this type in one of two general forms. The first is the activity combination parameter defined by:

$$\aleph_{ij} = |z_j| log a_i - \frac{z_i z_j}{|z_j|} log a_j \tag{33}$$

This is valid for i and j of any charge combination. Note that $\aleph_{H^+, Cl^-} = -pHCl$, so it is actually $-pHCl$ that is input to the code, not $pHCl$ (analogous to the input of $-pH$ instead of pH). The more general form of the governing equation is then:

$$log m_i = \frac{\aleph_{ij}}{|z_j|} - log\gamma_i + \frac{z_i}{z_j} log m_j + \frac{z_i}{z_j} log\gamma_j \tag{34}$$

The second general form is to input the mean log activity of the electrolyte composed of ions i and j:

$$log a_{\pm,ij} = \frac{|z_j| log a_i + |z_i| log a_j}{|z_i| + |z_j|} \tag{35}$$

This is not quite as general, because the two ions must have opposite signs of electrical charge. Note that $log\ a_{+,HCl} = -1/2\ pHCl$. The corresponding governing equation is:

$$log m_i = \frac{|z_i| + |z_j|}{|z_j|} log a_{\pm,ij} - log\gamma_i - \left|\frac{z_i}{z_j}\right| log m_j - \left|\frac{z_i}{z_j}\right| log\gamma_j \tag{36}$$

2.3.5. The Carbonate System: Dealing with Alkalinity

To model the carbonate system, EQ3NR expects as normal input an analytical value for total dissolved bicarbonate ($CO_{2(aq)} + HCO_3^- + CO_3^{2-}$, where these are taken in the sense of components, including any ion pairs or complexes of the corresponding species). The appropriate measurement can be made using ion chromatography or infrared detection of carbon dioxide released from an acidified sample. The results might be expressed on a data sheet as total dissolved CO_2 in mg/L. This must be converted to the equivalent concentration of HCO_3^- for input to EQ3NR, as it is defined on the supporting data files as the basis species corresponding to carbonate mass balance. This can be done by multiplying this quantity by the ratio of the molecular weight of HCO_3^- (61.016 g/mole) to that of $CO_{2(aq)}$ (44.009 g/mole) (the value of this ratio being 1.3864). A data sheet might also list a value for "total free CO_2." This represents only the $CO_{2(aq)}$ component. If this is the only available measure of dissolved bicarbonate, the total dissolved bicarbonate can be computed from this and the pH by inverting the relevant equations given later in this section.

Carbonate (in the form of CO_3^{2-} and HCO_3^-, including any ion pairs of these species) makes up nearly all of the alkalinity of many aqueous solutions. Strictly speaking, the alkalinity is a measure of the acid neutralizing capability of an aqueous solution. However, it is also commonly used as a measure of the carbonate system. In fact, alkalinity is only an indirect measure of this system, and its usage as such a measure entails a number of assumptions which are not always valid. In this context, it is also frequently misunderstood and misused. The worst consequence of this usage of alkalinity is that it leads people to think that a direct measurement of total dissolved bicarbonate (in the sense discussed in the above paragraph) is unnecessary. Indeed, it is common to find analytical data sheets on groundwater chemistry which lack such direct measurements.

The purpose of this section is to discuss these problems, and to suggest means by which the total dissolved bicarbonate (in the desired sense) can be estimated, in the case in which direct measurements are lacking. These means are not always entirely satisfactory, and are here suggested mainly for the benefit of those who must work with historical data.

EQ3NR formerly allowed titration alkalinity (A_t)to be input for bicarbonate instead of total concentration. This capability essentially matched that used in the PHREEQE code (Plummer, Parkhurst, and Thorstenson, 1980). The approach is to define an alkalinity balance equation, which is very similar to a mass balance equation. It may be written in the general form:

$$A_t = \sum_{s=1}^{s_T} \tau_s m_s \tag{37}$$

where τ_s is the alkalinity factor of the s-th species. This is the number of moles of hydrogen ion neutralized by one mole of species in the process of titrating the solution with an acid solution (usually of dilute sulfuric acid) to some end-point, usually in the range of pH 4.0 to 4.5 (See Standard Methods, 1976, p. 278-293; see also Plummer, Parkhurst, and Thorstenson, 1980, p. 17-18). Titration alkalinity defined in this manner is in units of equivalents per kilogram of solvent water, where "equivalent" means hydrogen ion neutralizing equivalent.

Titration alkalinity is usually not reported in these units, however. Standard Methods calls for reporting the titration alkalinity in terms of the stoichiometric equivalent of mg/L of $CaCO_3$. We will mark alkalinities in such units with an asterisk (*). Thus, the form of titration alkalinity usually reported must be converted according to:

$$A_t = \frac{A_t^*}{50,000\rho_{g/ml}} \tag{38}$$

The "50,000" in the above equation is actually the product of 1000 mg/g and the molecular weight of calcium carbonate (taken as 100 g/mole following Standard Methods), divided by the alkalinity factor of $CaCO_3$ (2.0 equivalents/mole). The molecular weight of $CaCO_3$ is more accurately 100.088 g/mole, but the 100 g/mole value is used by Standard Methods in the formula for calculating A_t^* from the titration data, so retaining it as above is actually more consistent with the titration measurement.

The titration alkalinity (A_t^*) is referred to in Standard Methods as T. This quantity may appear on analytical data sheets as "T" or "titration alkalinity" and in units marked "mg/L" or "mg/L $CaCO_3$." In this context, "mg/L" means "mg/L $CaCO_3$." Users of geochemical modeling codes sometimes mistakenly interpret "mg/L" to mean that the titration alkalinity is given in units of mg/L HCO_3^- (A_{t,HCO_3^-}). It is not unknown for analysts to report the titration alkalinity in such units as well, though this is not a standard practice. It can be obtained by multiplying A_t^* by the

molecular weight of HCO_3^- (61.016 g/mole) and the alkalinity factor of $CaCO_3$ (2.0 equivalents/mole) and dividing the result by the product of the molecular weight of $CaCO_3$ and the alkalinity factor of HCO_3^- (1.0 equivalents/mole). In simpler terms, one has:

$$A_{t,HCO_3^-} = 1.2192\, A_t^* \tag{39}$$

It is very important to note that the titration alkalinity expressed in mg/L HCO_3^- is not equal to the total dissolved bicarbonate (in the sense required for input to EQ3NR) expressed in the same units. Recall that $CO_{2(aq)}$ does not contribute to the titration alkalinity, while it does contribute to the desired total dissolved bicarbonate. Furthermore, CO_3^{2-} contributes differently to the titration alkalinity (by a factor of 2) than it does to the desired total dissolved bicarbonate.

In the determination of alkalinity, Standard Methods calls for two end points to be determined in the titration. One of these gives T, the titration alkalinity, the other P, the phenolphthalein alkalinity. The latter corresponds to an end point of pH 8.3. If the pH of the sample solution is already less than or equal to this, then $P = 0$. The phenolphthalein alkalinity may also appear on an analytical data sheet. Standard Methods calls for using P to partition the titration alkalinity into components due to bicarbonate, carbonate, and hydroxide; thus, one may write:

$$A_t^* = A^*_{HCO_3^-} + A^*_{CO_3^{2-}} + A^*_{OH^-} \tag{40}$$

This scheme implicitly assumes that no other components are present which contribute to the alkalinity. It also ignores ion pairing and complexing as it pertains to these species. Note that each of these component alkalinities is reported in units of equivalent mg/L $CaCO_3$.

These three components, bicarbonate alkalinity, carbonate alkalinity, and hydroxide alkalinity, are determined from T and P according to the partitioning formula given in Standard Methods (1976, Table 403:I, p. 281). At least one of these three always has a value of zero. Sometimes two are zero. They are supposed to be reported in units of mg/L $CaCO_3$. They are commonly found on analytical data sheets. Since they in essence determine the titration alkalinity, this quantity is sometimes omitted, and if it is desired, it must be computed from then using the above equation.

It is not unknown for analysts to report the bicarbonate alkalinity in units of mg/L HCO_3^-. Users have been known to confuse the bicarbonate alkalinity expressed in such units with the total dissolved bicarbonate (the desired input to EQ3NR), which may be expressed in the same units.

The concentration of the bicarbonate component can be computed from the bicarbonate alkalinity:

$$C_{HCO_3^-,mg/L} = 1.2192\, A^*_{HCO_3^-} \tag{41}$$

The numerical factor on the right hand side is the same as that appearing in eq (39). The molality of the bicarbonate component can be computed directly from the bicarbonate alkalinity:

$$m_{HCO_3^-} = \frac{A^*_{HCO_3^-}}{50,044} \quad (42)$$

The denominator on the right hand side is the product of 1000 mg/g, the molecular weight of $CaCO_3$, and the alkalinity factor of HCO_3^-, divided by the alkalinity factor of $CaCO_3$. The molecular weight of HCO_3^- is factored out in the derivation of this equation.

The concentration of the carbonate component can be similarly computed from the carbonate alkalinity:

$$C_{CO_3^{2-},mg/L} = 0.5996\, A^*_{CO_3^{2-}} \quad (43)$$

The numerical factor on the right hand side is the product of the molecular weight of CO_3^{2-} (60.008 g/mole) and the alkalinity factor of $CaCO_3$, divided by the product of the molecular weight of $CaCO_3$ and the alkalinity factor of CO_3^{2-} (also 2.0 equivalent/mole, so the alkalinity factors cancel out). The molality of the carbonate component can be computed directly from the carbonate alkalinity:

$$m_{CO_3^-} = \frac{A^*_{CO_3^-}}{100,088} \quad (44)$$

It is not unknown for analysts to report the total concentration of bicarbonate as:

$$\text{“}C_{T,HCO_3^-,mg/L}\text{”} = C_{HCO_3^-,mg/L} + 1.0168\, C_{CO_3^{2-},mg/L} \quad (45)$$

where the concentrations on the right hand side are obtained from alkalinities as above and the factor 1.0168 is the ratio of the molecular weight of HCO_3^- to that of CO_3^{2-} and is used to convert the units of carbonate concentration from mg/L CO_3^{2-} to the equivalent mg/L HCO_3^-. In terms of molalities, this is equivalent to taking:

$$\text{“}m_{T,HCO_3^-}\text{”} = m_{HCO_3^-} + m_{CO_3^{2-}} \quad (46)$$

This measure of total bicarbonate, whether reported in mg/L or molality, is not the measure of total bicarbonate which is to be input to EQ3NR, because it does not include the contribution from the component $CO_{2(aq)}$.

Above *pH* 8.3, the contribution of $CO_{2(aq)}$ to total bicarbonate is negligible (1% or less), and estimates based on the above formulations may be input to EQ3NR with negligible error. At lower *pH* values, the concentration of $CO_{2(aq)}$ can be estimated from the bicarbonate alkalinity and the *pH*. Standard Methods (1976, Figure 407:4, p. 297) gives a nomograph for this purpose. The nomograph also takes into account the dependency on the temperature and the ionic strength, using

the quantity "total filterable residue" as a proxy for the latter. If this procedure is followed, the total dissolved bicarbonate to be input to EQ3NR can be estimated as:

$$C_{T,\,mg/L,\,HCO_3^-} = 1.3864 C_{mg/L,\,CO_{2(aq)}} + C_{mg/L,\,HCO_3^-} + 1.0168 C_{mg/L,\,CO_3^{2-}} \quad (47)$$

where the factor 1.3864 is the ratio of the molecular weight of HCO_3^- to that of $CO_{2(aq)}$ (44.009 g/mole). In terms of molalities, this is equivalent to:

$$m_{T,\,HCO_3^-} = m_{CO_{2aq}} + m_{HCO_3^-} + m_{CO_3^{2-}} \quad (48)$$

As an alternative to the nomograph of Standard Methods, we note that the molality of $CO_{2(aq)}$ may be estimated from the molality of HCO_3^- and the pH by considering equilibrium for the reaction:

$$CO_{2\,(aq)} + H_2O_{\,(l)} = H^+ + HCO_3^- \quad (49)$$

Assuming that the activity of water differs negligibly from a value of unity, the following equation is obtained:

$$m_{CO_{2(aq)}} = 10^{\left(-\log K_{CO_{2(aq)}} + \log \gamma_{HCO_3^-} - \log \gamma_{CO_{2(aq)}} - pH\right)} m_{HCO_3^-} \quad (50)$$

The *log K* for reaction (49) has values of -6.5804, -6.3447, and -6.2684 at 0, 25, and 60°C (data taken from the **data0.sup.R10** data file). At 25°C, this reduces to:

$$m_{CO_{2\,(aq)}} = 10^{6.32 - pH} m_{HCO_3^-} \quad (51)$$

for a dilute solution of ionic strength 0.0024 molal (using the B-dot equation to calculate the activity coefficients; see Chapter 3). From this, one can see that at pH 4.33, the molality $CO_{2(aq)}$ is 100 times that of HCO_3^-. For seawater (ionic strength of 0.662 molal), the equation becomes:

$$m_{CO_{2\,(aq)}} = 10^{6.14 - pH} m_{HCO_3^-} \quad (52)$$

One of the points that may be deduced from these equations is that alkalinity is a poor way to measure the carbonate system in waters of relatively low pH, in which $CO_{2(aq)}$ dominates the total dissolved bicarbonate (defined in the sense desired for input to EQ3NR). The propagated uncertainty in such calculations can become large owing to a contribution from the uncertainty in pH measurement in addition to one from the uncertainty in the measurement in the titration alkalinity (which is interpreted as entirely bicarbonate alkalinity at such low pH). The propagated error is also affected to some degree by uncertainty in the values of the activity coefficients, though this is not likely to be of much significance in very dilute solutions. It is probably affected much more by contributions due to uncertainties regarding the contribution to the measured al-

kalinity of non-carbonate species. This is a potential major problem regarding the use of alkalinity in solutions of any *pH* value.

We have shown above how the total dissolved bicarbonate (in the sense of including aqueous carbon dioxide and carbonate) may be estimated from alkalinity measurements. These methods assume that only bicarbonate, carbonate, and hydroxide contribute to the measured alkalinity. Alkalinity can also be contributed by dissolved organic species such as acetate, by components such as borate, phosphate, silicate, and sulfide, and by some dissolved metals, such as iron and aluminum, in the form of hydroxy complexes. Of course, if one knows the concentrations of the relevant species, corrections may be attempted. Such corrections could take the form of subtracting the estimated contributions from the measured titration alkalinity. Alternatively, one can make the corrections in a speciation-solubility calculation, using an alkalinity balance equation. It requires assigning alkalinity factors to all the relevant species. Such an approach is available in the PHREEQE code (Plummer, Parkhurst, and Thorstenson, 1980) and previous versions of EQ3NR (Wolery, 1983). Either form of correction carries various uncertainties, however, and major problems arise when the corrections are large. In using previous versions of the EQ3NR code in this way, the code has occasionally terminated unsuccessfully because the magnitude of the corrections would have exceeded the value of the reported titration alkalinity.

Many waters of interest to geochemists have substantial amounts of alkalinity due to non-carbonate species. In oil field waters, the titration alkalinity is often heavily dominated (50-100%) by short chain aliphatic anions, chiefly acetate, propionate, butyrate, and valerate, in order of decreasing importance (Willey et al., 1975; Carothers and Kharaka, 1978). Organic anions are also present in significant concentrations in the water in and around many landfills and other geologic waste disposal sites. These may be both products of the decomposition of organic wastes and original components of the disposed waste. Waters at disposal sites may also be rich in other components which contribute to alkalinity, such as sulfide, ammonia, phosphate, silicate, and metal hydroxy complexes. Many natural waters of interest are also high in sulfide, and a few are high in borate.

The titration alkalinity input option was removed from EQ3NR for the following reasons:

- To avoid undue propagation of errors inherent in the method, which can be severe in certain cases.
- To avoid possible errors by both analysts and code users concerning the nature, interpretation, and usage of analytical data.
- To avoid the problem of having to assign alkalinity factors to new species added to the supporting data files.
- To encourage the practice of obtaining direct analytical measures of total dissolved bicarbonate.

For cases in which the code user must deal with historical data which include alkalinity measurements but no direct measures of the carbonate system, the following procedure is recommended:

- Using the reported alkalinity and *pH* values, estimate the total dissolved bicarbonate (total dissolved carbon dioxide expressed as bicarbonate) using the methods presented above; make rough corrections if possible for contributions of organics, sulfide, etc.

- Compute a model of the solution.

- Feed the model of the solution to the EQ6 code; simulate the titration process as described by Standard Methods (1976) and compute the corresponding value of the titration alkalinity (an example of this use of EQ6 is given in the EQ6 Theoretical Manual and User's Guide, Wolery and Daveler, 1992).

- Compare the computed value of the titration alkalinity with the reported value; if they match, stop; if not, adjust the estimate of the total dissolved bicarbonate accordingly and repeat the process until a reasonable match if obtained.

This procedure may not always work. For example, if the contribution of bicarbonate/carbonate to the alkalinity is very small compared to that of organics, sulfide, etc., then the available data really offer no constraint on the bicarbonate/carbonate system. In such cases, the user would be wise to recognize the futility of the situation.

2.3.6. Redox Constraints

2.3.6.1. There is No "System" *Eh* in Most Real Systems

The high degree of emphasis on trying to understand the geochemistry of natural waters in terms of pure equilibrium thermodynamics has misled many people into believing that the redox state of real aqueous systems can be characterized by a single parameter, usually the *Eh* (a redox potential, given in volts). The related parameter *pe*, the negative of the logarithm of the hypothetical electron, is similarly incapable of describing the overall redox state of a real aqueous system.

The concept of there being such a thing as a "system" *Eh* or a "system" *pe* is based on the assumption that all redox reactions in an aqueous system are in a state of thermodynamic equilibrium. This assumption is a poor one for most real systems (Morris and Stumm, 1967; Jenne, 1981; Thorstenson, 1983; Hostetler, 1984; Lindberg and Runnells, 1984). In the rush to interpret geochemical data by means of *Eh-pH* and *pe*-activity diagrams, this point is often forgotten or simply ignored. This has had the unfortunate consequence of legitimizing these variables as all-encompassing redox descriptors in the minds of many students.

This misconception has no doubt been reinforced by the use of *Eh* (and sometimes *pe*) as inputs to speciation-solubility codes. Some of these codes require the assumption of a system *Eh*. Most of the better known codes, EQ3NR, WATEQ2 (Ball, Jenne, and Nordstrom, 1979), and PHREEQE (Parkhurst, Plummer, and Thorstenson, 1980) permit the use of such an input but do not require it. With sufficient analytical data, the degree of disequilibrium among various redox couples may be calculated, and the existence of a system *Eh* thus tested. Often, however, the available analytical data are insufficient to do this, and one is forced to assume a system *Eh*.

Redox disequilibrium in natural aqueous systems is created by solar irradiation, radioactive decay, fluid mixing, and transfer of redox components from one phase to another. It is maintained

primarily by the strength of covalent bonds, a major factor in the redox disequilibrium of the light elements such as carbon, hydrogen, oxygen, nitrogen, and sulfur. Biological activity literally feeds on redox disequilibrium (e.g., photosynthesis, if one counts the initial disequilibrium due to solar radiation, and chemosynthesis) and catalyzes an overall approach toward redox equilibrium.

Several well known examples of redox disequilibrium in natural aqueous systems can be cited. One is the coexistence of dissolved oxygen and organic carbon in nearly all natural waters, implying disequilibrium between the $O_{2(aq)}/H_2O_{(l)}$ couple and organic/HCO_3^- couples. Another is the disequilibrium between $CH_{4(aq)}/HCO_3^-$ and HS^-/SO_4^{2-} in many marine sediments (Thorstenson, 1970). A third example is the disequilibrium of $N_{2(aq)}/NO_3^-$ with $O_{2(aq)}/H_2O_{(l)}$ in marine surface waters (Berner, 1971, p. 119).

As shown below, each such couple can be treated as having have its own redox state. This can be expressed in a variety of ways, including a couple-specific *Eh* or *pe*. These can be calculated using the Nernst equation in conjunction with chemical analyses that are specific with respect to the two oxidation states represented in any redox couple. In the following section, we will discuss the details of this concept.

2.3.6.2. Background: Redox Couples and Half-Reactions

Oxidation-reduction in aqueous systems is commonly treated in terms of redox couples and their associated half-reactions. Common couples in aqueous solution include $O_{2(aq)}/H_2O_{(l)}$, $H_{2(aq)}/H_2O_{(l)}$, Fe^{2+}/Fe^{3+}, HS^-/SO_4^{2-}, SO_3^{2-}/SO_4^{2-}, $S_2O_3^{2-}/SO_4^{2-}$, NH_4^+/NO_3^-, $N_{2(aq)}/NO_3^-$, $CH_{4(aq)}/HCO_3^-$, and a host of organic/HCO_3^- couples. The half-reaction is illustrated in the case of the very important couple $O_{2(aq)}/H_2O_{(l)}$:

$$2H_2O_{(l)} = O_{2(aq)} + 4H^+ + 4e^- \quad (53)$$

Another very important half-reaction corresponds to the so-called hydrogen electrode:

$$H_{2(g)} = 2H^+ + 2e^- \quad (54)$$

Multiplying this half-reaction by two and subtracting it from the first yields the following complete redox reaction (which has no electrons among the reactants or products):

$$2H_2O_{(l)} = O_{2(aq)} + 2H_{2(g)} \quad (55)$$

The thermodynamic convention used to describe the state of electrical potentials of half-reactions in terms of Eh values is to take the electrical potential of the standard hydrogen electrode as zero at all temperatures and pressures. This is consistent with the following additional thermodynamic conventions, where $\Delta G^o_{f,i}$ is the standard state Gibbs energy of the i-th species:

- $\Delta G^o_{f,H_{2(g)}} = 0$ at all temperatures and pressures (the standard state fugacity is 1 bar)

- $\Delta G^{o}_{f,H^+} = 0$ at all temperatures and pressures
- $\Delta G^{o}_{f,e^-} = \Delta G_{f,e^-} = 0$ at all temperatures and pressures

The Gibbs energy (ΔG) is related to the electrical potential (E) by the Nernst equation (cf. Garrels and Christ, 1965):

$$\Delta G = +nFE \tag{56}$$

where n is the number of electrons in the half-reaction and F is the Faraday constant.

An alternative treatment almost equivalent to that above is to write the half-reactions as reduction reactions, so that the electron appears on the left hand side. One then reverses the sign of the right hand side of the Nernst equation as written above. This development is equivalent to the one above, except that the signs of the Gibbs energies and corresponding equilibrium constants and activity products are reversed (Stumm and Morgan, 1981, Chapter 8).

The standard thermodynamic relation describing the Gibbs energy of reaction (ΔG_r),of the r-th reaction is:

$$\Delta G_r = \Delta G^{o}_r + 2.303RT \log Q_r \tag{57}$$

where ΔG^{o}_r is the corresponding standard state Gibbs energy of reaction and Q_r is the corresponding activity product. Applying this to half-reactions and using the positive convention version of the Nernst equation, one obtains:

$$E = E^{o} + \frac{2.303RT}{nF} \log Q_{+,\,1/2} \tag{58}$$

where E^{o} is the standard state electrical potential, and $Q_{+,\,1/2}$ is the activity product of the half-reaction. If one uses the negative convention version of the Nernst equation, one obtains instead:

$$E = E^{o} - \frac{2.303RT}{nF} \log Q_{-,\,1/2} \tag{59}$$

where $Q_{-,\,1/2}$ is the activity product of the reverse half-reaction. These relations are equivalent because $\log Q_{-,\,1/2} = -\log Q_{+,\,1/2}$. Because the Gibbs energy of the hypothetical electron is always zero, whether it is in the standard state or not, its thermodynamic activity is fixed at unity and it need not explicitly appear in the activity product expressions for half-reactions.

2.3.6.3. Background: *Eh*, *pe*, *Ah*, and Equilibrium Oxygen Fugacity

One can write a modified Nernst equation for any redox couple. In the case of the ferrous-ferric couple, the corresponding half-reaction is:

$$Fe^{2+} = Fe^{3+} + e^{-} \tag{60}$$

The corresponding form of the Nernst equation is:

$$E_{Fe^{2+}/Fe^{3+}} = E^{o}_{Fe^{2+}/Fe^{3+}} + \frac{2.303RT}{F} log\left(\frac{a_{Fe^{3+}}}{a_{Fe^{2+}}}\right) \quad (61)$$

Under the thermodynamic conventions adopted above, the potential E on the left hand side of each of the above equations can be taken as the Eh for this specific couple. This may or may not equate to the results of an "Eh" measurement.

If the two redox couples are in equilibrium with each other, they must have the same Eh. Conversely, if they have the same Eh, they must be in equilibrium. Any difference in couple-specific Eh values is a measure (in volts) of the degree of disequilibrium. This can be shown by relating the Gibbs energy of a combined, complete reaction to the differences in potentials. If the first half- reaction has Eh_1 and n_1 electrons appear in it, and the second half- reaction has Eh_2 and n_2 electrons, one can construct a complete reaction by multiplying the second half-reaction by - n_1/n_2 and adding the result to the first half-reaction. Then n_1 electrons are transferred in the complete reaction. The Gibbs energy of this reaction is then given by:

$$\Delta G_r = n_1 F (Eh_1 - Eh_2) \quad (62)$$

The condition of zero Gibbs energy of reaction (thermodynamic equilibrium) is met if and only if $Eh_1 = Eh_2$.

The redox parameter pe, popularized by Truesdell (1968) and Stumm and Morgan (1981), is defined to be analogous to pH:

$$pe = -log a_{e^-} \quad (63)$$

where e^- is the hypothetical aqueous electron. It should not be confused with real aqueous electrons, which are extremely scarce in nature. Their thermodynamic properties are not the same. In fact, the hypothetical electron used to define pe is not the same as the one used to define Eh. The Eh conventions require the activity of the hypothetical electron to always be unity. That convention would fix pe at a value of zero.

The relation between pe and Eh is:

$$pe = \left(\frac{F}{2.303RT}\right) Eh \quad (64)$$

(Thorstenson, 1970; Stumm and Morgan, 1981). One may derive that this requires the thermodynamic convention

- $\Delta G_{f,e^-} = \frac{1}{2}\Delta G_{H_{2(g)}} - \Delta G_{H^+}$ at all temperatures and pressures

This in turn requires that

- $\Delta G_{f,e^-} = 0$ at all temperatures and pressures

whereas the Eh convention for the hypothetical electron was

- $\Delta G^{o}_{f,e^-} = \Delta G_{f,e^-} = 0$ at all temperatures and pressures

It should be clear that *pe* is not a perfect analog to *pH*, because *pH* is defined with respect to H^+, a real aqueous species, whereas *pe* is defined with respect to a hypothetical species. Each redox couple can have its own *pe*, just as it can have its own *Eh*, the two being related by the equation given above. It follows from the previous development that thermodynamic equilibrium between two redox couples is synonymous with each having the same value of *pe*.

The state of an aqueous redox couple can also be expressed in terms of chemical affinity by the redox affinity, *Ah* (Wolery, 1983). This is a special case of the thermodynamic affinity function (its application to half-reactions). It is related to *Eh* by the relation

$$Ah = FEh \tag{65}$$

The driving force for any kind of complete chemical reaction (meaning to exclude half-reactions) can be expressed by the thermodynamic affinity (A_r), which is related to the equilibrium constant K_r and the activity product Q_r by the equation:

$$A_r = -2.303RTlog\left(\frac{Q_r}{K_r}\right) \tag{66}$$

If n_1 electrons appear in one half-reaction and n_2 in another, the two half-reactions can be combined into a complete redox reaction in the manner discussed previously. The thermodynamic affinity of the complete reaction, in which n_1 electrons are transferred, is then related to the *Ah* values (Ah_1 and Ah_2, respectively) of the two half reactions by the equation:

$$A_r = n_1 (Ah_1 - Ah_2) \tag{67}$$

Thermodynamic equilibrium ($A = 0$) among two redox couples is the case if and only if both couples have the same value of *Ah*.

Alternatively, the state of a redox couple may be expressed in terms of an equilibrium oxygen fugacity (a couple-specific oxygen fugacity). Fugacities are properties of gas species. Gas species do not exist in aqueous solution because, by definition, all species in aqueous solution are aqueous species. Therefore, we can only talk about oxygen fugacities in aqueous solution by reference to hypothetical equilibria with a gas phase. Putting it another way, $O_{2(g)}$ makes a perfectly good hypothetical aqueous species, much like the hypothetical aqueous electron.

Consider the half-reaction:

$$2H_2O_{(l)} = O_{2(g)} + 4H^+ + 4e^- \tag{68}$$

where we now take $O_{2(g)}$ to be a hypothetical aqueous species with the thermodynamic properties of the real gas species. We can calculate an equilibrium oxygen fugacity for any half-reaction by coupling it with this one to form a complete redox reaction.

Let K_{Eh} be the equilibrium constant for the $O_{2(g)}/H_2O_{(l)}$ half- reaction given above. *Eh* and oxygen fugacity are then related by the equation:

$$logf_{O_2} = (\frac{4F}{2.303RT})Eh - 4loga_{H^+} + 2loga_w + logK_{Eh} \quad (69)$$

This equation can be used to relate the equilibrium oxygen fugacity of any redox couple with its own *Eh*. Two redox couples are in thermodynamic equilibrium with each other if and only if they have the same equilibrium oxygen fugacity.

2.3.6.4. Redox Options: Testing versus Assuming Equilibrium

A commonly used approach in aqueous speciation modeling is to input a total concentration for a dissolved element that occurs in more than one oxidation state and partition it according to a given *Eh*, *pe*, or oxygen fugacity. This, however, requires us to assume that all redox couples in the system are in a state of thermodynamic equilibrium. The EQ3NR code offers this option.

If we constrain the thermodynamic activities of all the aqueous species appearing in a couple's half-reaction without resorting to an input *Eh*, *Ah*, *pe*, or oxygen fugacity, the equations presented above give us a means to calculate its individual redox state expressed as any of the following:

- *Eh* (in terms of an electrical potential).
- *Ah* (in terms of a chemical potential).
- *pe*.
- oxygen fugacity.

Analytical techniques do not generally discriminate between a simple species and its ion-pairs and complexes. However, there are techniques in many cases to discriminate between different oxidation states. To calculate the *Eh* of the ferrous-ferric couple, for example, we must have analytical data for both Fe^{2+} and Fe^{3+} (see for example Nordstrom, Jenne, and Ball, 1979). If these data are both total concentrations (e.g, total Fe^{2+}, total Fe^{3+}), we simply have two mass balance equations for iron in the aqueous speciation model instead of one.

This is the preferred approach for treating oxidation-reduction in aqueous speciation modeling (see Nordstrom et al., 1979). One may then test whether or not various redox couples are in equilibrium with each other. EQ3NR can treat any redox couple in this fashion. Alternative constraints discussed in the previous subsection could substitute for one or both total concentrations/ mass balances in the usual way. The code will use a redox default to partition an element that appears in more than one oxidation state if insufficient data are input to calculate a couple-specific parameter. The redox default may be an input *Eh*, a *pe*, or log oxygen fugacity. Alternatively, it may be defined by a redox couple for which sufficient data are input to calculate couple-specific parameters. By constraining one or more of the species in the corresponding half-reaction by a heterogeneous equilibrium constraint, it is possible to constrain the default redox state by a heterogeneous equilibrium.

2.3.7. Measures of Mineral Saturation.

EQ3NR employs two measures of the saturation state of an aqueous solution with respect to minerals. The first is the saturation index defined as:

$$SI = log\frac{Q}{K} \tag{70}$$

where it is understood that Q is the activity product and K the equilibrium constant for a dissolution reaction. In the case of the dissolution of calcite, for example, if the reaction is written as:

$$Calcite + H^+ = Ca^{2+} + HCO_3^- \tag{71}$$

the ion activity product is then defined as:

$$Q_{Calcite} = \frac{a_{Ca^{2+}} a_{HCO_3^-}}{a_{Calcite} a_{H^+}} \tag{72}$$

The second measure of the saturation state is the thermodynamic affinity of the precipitation reaction. The affinity of a reaction (no matter how it is written) is related to its activity product and equilibrium constant by:

$$A = -2.303RTlog\frac{K}{Q} = 2.303RTlog\frac{Q}{K} \tag{73}$$

Because *log Q/K* reverses sign when the reaction is reversed, the affinity to precipitate is related to the saturation index by:

$$A_{\_} = 2.303RT\,SI \tag{74}$$

Following these conventions, both SI and $A_{\_}$ are positive for supersaturated minerals, zero for saturated ones, and negative for undersaturated minerals.

In the case of solid solution minerals with end-member components, the saturation index of the σ-th end member is related to that of the corresponding pure phase ϕ by:

$$SI_{\sigma\psi} = SI_{\phi} - log a_{\sigma\psi} = SI_{\phi} - log x_{\sigma\psi} - log\lambda_{\sigma\psi} \tag{75}$$

where $a_{\sigma\psi}$ is the thermodynamic activity of the end-member, $x_{\sigma\psi}$ is its mole fraction, and $\lambda_{\sigma\psi}$ is its mole fraction activity coefficient. Consideration of an overall dissolution reaction of a solid solution of given composition suggests that the saturation index of the ψ-th solid solution should be defined by:

$$SI_{\psi} = \sum_{\sigma=1}^{\sigma_{T,\psi}} x_{\sigma\psi} SI_{\sigma\psi} \tag{76}$$

Affinity functions can be defined analogously.

The problem of defining the saturation state of a solid solution for which no composition is given is not so straightforward, because the result is composition-dependent. One way to approach this would be to find the compositions that maximize the *SI*. This is the method presently employed in EQ3/6 (see Bourcier, 1985, 1989).

2.4. Use and Misuse of Speciation-Solubility Codes

There is significant potential to misuse any speciation-solubility code. No such code should be used as a "black box". As Jenne (1981, p. 36) puts it, "... each application should be viewed as a partial validation." The geochemical model of each new scenario (e.g., a set of waters in a compositional range not previously studied) may have a different set of important aqueous species, and hence provide a test of some thermodynamic data that have not previously been exercised. Also, reactions controlled by equilibrium in one situation may be in disequilibrium in another, and vice versa, especially heterogeneous and aqueous redox reactions.

Geochemical modeling with aqueous speciation-solubility codes must actively address three questions. First, are all the significant species in the model? Second, are all the important thermodynamic data sufficiently correct? Do they make sense when compared with the model outputs when working with a set of wa[illegible]amples? Do they make sense in comparison with other knowledge about an aqueous syste[illegible]ch as data on the identities of minerals with which the water is in contact? Third, would disequilibrium constraints be more appropriate than equilibrium constraints for some reactions (especially aqueous redox reactions)? Users should keep in mind the admonition of Nordstrom et al. (1979) that "... no model is better than the assumptions on which it is based."

If no thermodynamic data are available for species known or suspected to be important in a given application, then such data should be estimated by empirical or semi-empirical methods. EQ3NR has an **input** file option which permits the user to temporarily modify equilibrium constants at run time (see Chapter 6). This makes it convenient to conduct sensitivity studies of the uncertainty in such estimated values. Langmuir (1979) summarizes approaches for estimating thermodynamic properties of aqueous species and reactions. Tardy and Garrels (1974), Wolery (1978), and Helgeson et al. (1978) discuss methods for estimating the thermodynamic properties of minerals.

A common problem faced by novices at speciation-solubility modeling is that their models come out grossly supersaturated with nearly every aluminum and ferric iron bearing mineral in the data base. This often occurs because analysis is made of inadequately filtered samples, which commonly contain colloidal particles of these two components. These particles then "inflate" the corresponding chemical analyses. Busenberg (1978) showed that large quantities of a colloidal aluminum phase occurred in the size range 0.1-0.45 μm during a set of feldspar dissolution experiments. Laxen and Chandler (1982) did more detailed studies of iron particulate size distribution in fresh waters. Their work shows that a filter finer than 0.1 μm is necessary to effectively remove these particulates from the chemical analysis.

The modeler should be aware that many solubility-controlling phases, especially at low temperature, are metastable (e.g., amorphous $Fe(OH)_3$ may control the level of dissolved iron, not the

more stable hematite, Fe_2O_3). In addition, the stability of some controlling phases may be somewhat variable due to such factors as crystallinity (i.e., crystal size), order/disorder, ionic substitution, or, in the case of fresh precipitates, aging. Helgeson et al. (1978) discuss many of these effects.

One approach that may be helpful to users is to estimate the amount of aluminum or iron that would be in solution under the assumption of appropriate solubility equilibria. For example, one might constrain dissolved aluminum to satisfy equilibrium with gibbsite ($Al(OH)_{3(c)}$) or constrain iron to satisfy equilibrium with amorphous $Fe(OH)_3$ or a nontronite (ferric-rich smectite) clay. However, this is not a substitute for analysis of carefully filtered samples.

The state of available analytical data on water compositions is often a limiting factor in the usage of EQ3NR or any other speciation-solubility code. In general, the data must be both accurate and sufficiently complete. Inaccurate data often result when methods suitable for analyzing drinking water are applied to waters very dissimilar to this medium. This can take the form of both positive and negative interferences. Some analytical parameters (e.g., *pH*, alkalinity, dissolved sulfide) must be measured immediately upon sampling to avoid changes due to mineral precipitation, ingassing, or outgassing. Water samples should be inspected after transportation and storage for the formation of precipitates. Quite often, analytical data are incomplete for geochemical modeling purposes. This may have the effect of completely inhibiting modeling work, or it may result in modeling with assumptions that could have been avoided if the right hard data had been available. In general, analytical work is most useful to modeling if there is interplay between the modeler and the analyst.

Internal consistency can provide useful tests of the quality of aqueous speciation models (see Merino, 1979). One such test is to compare the calculated electrical imbalance with the cation/anion subtotals for charge equivalents. EQ3NR makes these calculations, which are a meaningful test if electrical balance is not used as an input constraint. Merino (1979) also recommends the technique of comparing measured and independently calculated values of titration alkalinity. In essence, his recommendation corresponds to the currently recommended procedure for dealing with alkalinity described earlier in this chapter.

3. Activity Coefficients of Aqueous Species

3.1. Introduction

The thermodynamic activities (a_i) of aqueous solute species are usually defined on the basis of molalities. Thus, they can be described by the product of their molal concentrations (m_i) and their molal activity coefficients (γ_i):

$$a_i = m_i \gamma_i \tag{77}$$

The thermodynamic activity of the water (a_w) is always defined on a mole fraction basis. Thus, it can be described analogously by product of the mole fraction of water (x_w) and its mole fraction activity coefficient (λ_w):

$$a_w = x_w \lambda_w \tag{78}$$

It is also possible to describe the thermodynamic activities of aqueous solutes on a mole fraction basis. However, such mole fraction-based activities ($a_i^{(x)}$) are not the same as the more familiar molality-based activities ($a_i^{(m)}$), as they are defined with respect to different choices of standard states. Mole fraction based activities and activity coefficients (λ_i), are occasionally applied to aqueous nonelectrolyte species, such as ethanol in water. In geochemistry, the aqueous solutions of interest almost always contain electrolytes, so mole-fraction based activities and activity coefficients of solute species are little more than theoretical curiosities. In EQ3/6, only molality-based activities and activity coefficients are used for such species, so a_i always implies $a_i^{(m)}$. Because of the nature of molality, it is not possible to define the activity and activity coefficient of water on a molal basis; thus, a_w always means $a_w^{(x)}$.

Solution thermodynamics is a construct designed to approximate reality in terms of deviations from some defined ideal behavior. The complex dependency of the activities on solution composition is thus dealt with by shifting the problem to one of describing the activity coefficients. The usual treatment of aqueous solutions is one which simultaneously employs quantities derived from, and therefore belonging to, two distinct models of ideality (Wolery, 1990). All solute activity coefficients are based on molality and have unit value in the corresponding model of ideality, called molality-based ideality. The activity and activity coefficient of water are not constant in an ideal solution of this type, though they do approach unit value at infinite dilution. These solvent properties are derived from mole fraction-based ideality, in which the mole fraction activity coefficients of all species components in solution have unit value. In an ideal solution of this type, the molal activity coefficients of the solutes are not unity, though they approach it at infinite dilution (see Wolery, 1990).

Any geochemical modeling code which treats aqueous solutions must provide one or more models by which to compute the activity coefficients of the solute species and the solvent. In many codes, what is computed is the set of γ_i plus a_w. As many of the older such codes were constructed to deal only with dilute solutions in which the activity of water is no less than about 0.98, some of these just take the activity of water to be unity. With the advent of activity coefficient models

of practical usage in concentrated solutions (mostly based on Pitzer's 1973, 1975 equations), there has been a movement away from this particular and severe approximation. Nevertheless, it is generally the activity of water, rather than the activity coefficient of water, which is evaluated from the model equations. This is what was previously done in EQ3/6. However, EQ3/6 now evaluates the set of γ_i plus λ_w. This is done to avoid possible computational singularities that may arise, for example if heterogeneous equilibria happen to fix the activity of water (e.g., when a solution is saturated with both gypsum and anhydrite).

Good models for activity coefficients must be accurate. A prerequisite for general accuracy is thermodynamic consistency. The activity coefficient of each aqueous species is not independent of that of any of the others. Each is related to a corresponding partial derivative of the excess Gibbs energy of the solution (G^{EX}). The excess Gibbs energy is the difference between the complete Gibbs energy and the ideal Gibbs energy. Because there are two models of ideality, hence two models for the ideal Gibbs energy, there are two forms of the excess Gibbs energy, G^{EXm} (molality-based) and G^{EXx} (mole fraction-based). The consequences of this are discussed by Wolery (1990). In version 7.0 of EQ3/6, all activity coefficient models are based on ideality defined in terms of molality. Thus, the excess Gibbs energy of concern is G^{EXm}. The activity of water, which is based on mole-fraction ideality, is imported into this structure as discussed by Wolery (1990). The relevant differential equations are:

$$ln\,\gamma_i = \frac{1}{RT}\frac{\partial G^{EXm}}{\partial n_i} \tag{79}$$

$$ln\,a_w = -\frac{\Sigma m}{\Omega} + \frac{1}{RT}\frac{\partial G^{EXm}}{\partial n_w} \tag{80}$$

where R is the gas constant, T the absolute temperature, Ω the number of moles of solvent water comprising a mass of 1 kg ($\Omega \approx 55.51$),and:

$$\Sigma m = \sum_i m_i \tag{81}$$

the sum of molalities of all solute species. Given an expression for the excess Gibbs energy, such equations give a guaranteed route to thermodynamically consistent results (Pitzer, 1984; Wolery, 1990). Equations that are derived by other routes may be tested for consistency using other relations, such as the following forms of the cross-differentiation rule (Wolery, 1990):

$$\frac{\partial\, ln\,\gamma_j}{\partial m_i} = \frac{\partial\, ln\gamma_i}{\partial m_j} \tag{82}$$

$$\frac{\partial\, ln\,a_w}{\partial n_i} = \frac{\partial\, ln\,\gamma_i}{\partial n_w} - \frac{1}{n_w} \tag{83}$$

In general, such equations are most easily used to prove that a set of model equations is not thermodynamically consistent. The issue of sufficiency in proving consistency using these and related equations (Gibbs-Duhem equations and sum rules) is addressed by Wolery (1990).

The activity coefficients in reality are complex functions of the composition of the aqueous solution. In electrolyte solutions, the activity coefficients are influenced mainly by electrical interactions. Much of their behavior can be correlated in terms of the ionic strength, defined by:

$$I = \frac{1}{2}\sum_i m_i z_i^2 \tag{84}$$

where the summation is over all aqueous solute species and z_i is the electrical charge. However, the use of the ionic strength as a means of correlating and predicting activity coefficients has been taken to unrealistic extremes (e.g., in the mean salt method of Garrels and Christ, 1965, p. 58-60). In general, model equations which express the dependence of activity coefficients on solution composition only in terms of the ionic strength are restricted in applicability to dilute solutions.

The three basic options for computing the activity coefficients of aqueous species in EQ3/6 are models based respectively on the Davies (1962) equation, the "B-dot" equation of Helgeson (1969), and Pitzer's (1973, 1975, 1979, 1987) equations. The first two models, owing to limitations on accuracy, are only useful in dilute solutions (up to ionic strengths of 1 molal at most). The third basic model is useful in highly concentrated as well as dilute solutions, but is limited in terms of the components that can be treated.

With regard to temperature and pressure dependence, all of the following models are parameterized along the 1 atm/steam saturation curve. This corresponds to the way in which the temperature and pressure dependence of standard state thermodynamic data are also presently treated in the software. The pressure is thus a function of the temperature rather than an independent variable, being fixed at 1.013 bar from 0-100°C and the pressure for steam/liquid water equilibrium from 100-300°C. However, some of the data files have more limited temperature ranges.

3.2. The Davies Equation

The first activity coefficient model in EQ3/6 is based on the Davies (1962) equation:

$$\log \gamma_i = -A_{\gamma,\,10} z_i^2 \left(\frac{\sqrt{I}}{1+\sqrt{I}} + 0.2I \right) \tag{85}$$

(the constant 0.2 is sometimes also taken as 0.3). This is a simple extended Debye-Hückel model (it reduces to a simple Debye-Hückel model if the "0.2I" part is removed). The Davies equation is frequently used in geochemical modeling (e.g., Parkhurst, Plummer, and Thorstenson, 1980; Stumm and Morgan, 1981). Note that it expresses all dependence on the solution composition through the ionic strength. Also, the activity coefficient is given in terms of the base ten logarithm, instead of the natural logarithm. The Debye-Hückel A_γ parameter bears the additional label "10" to ensure consistency with this. The Davies equation is normally only used for temperatures close to 25°C. It is only accurate up to ionic strengths of a few tenths molal in most

solutions. In some solutions, inaccuracy, defined as the condition of model results differing from experimental measurements by more than the experimental error, is apparent at even lower concentrations.

In EQ3/6, the Davies equation option is selected by setting the option flag **iopg1** = -1. A supporting data file consistent with the use of a simple extended Debye-Hückel model must also be supplied (e.g., **data1** = **data1.com**, **data1.sup**, or **data1.nea**). If **iopg1** = -1 and the supporting data file is not of the appropriate type, the software terminates with an error message.

The Davies equation has one great strength: the only species-specific parameter required is the electrical charge. This equation may therefore readily be applied to a wide spectrum of species, both those whose existence is well-established and those whose existence is only hypothetical.

The Davies equation predicts a unit activity coefficient for all neutral solute species. This is known to be inaccurate. In general, the activity coefficients of neutral species that are non-polar (such as $O_{2(aq)}$, $H_{2(aq)}$, and $N_{2(aq)}$) increase with increasing ionic strength (the "salting out effect," so named in reference to the corresponding decreasing solubilities of such species as the salt concentration is increased; cf. Garrels and Christ, 1965, p. 67-70). In addition, Reardon and Langmuir (1976) have shown that the activity coefficients of two polar neutral species (the ion pairs $CaSO_{4(aq)}$ and $MgSO_{4(aq)}$) decrease with increasing ionic strength, presumably as a consequence of dipole-ion interactions.

The Davies equation is thermodynamically consistent. It is easy to show, for example, that it satisfies the solute-solute form of the cross-differentiation equation.

Most computer codes using the Davies equation set the activity of water to one of the following: unity, the mole fraction of water, or a limiting expression for the mole fraction of water. Usage of any of these violates thermodynamic consistency, but this is probably not of great significance as the inconsistency is numerically not significant at the relatively low concentrations at which the Davies equation itself is accurate. For usage in EQ3/6, we have used standard thermodynamic relations to derive the following expression:

$$\log a_w = \frac{1}{\Omega}\left(-\frac{\Sigma m}{2.303} + \frac{2}{3}A_{\gamma,10}I^{\frac{3}{2}}\sigma(\sqrt{I}) - 2(0.2)A_{\gamma,10}I^2\right) \tag{86}$$

where "2.303" is a symbol for and approximation of $ln\ 10$ (warning: this is not in general a sufficiently accurate approximation) and:

$$\sigma(x) = \frac{3}{x^3}\left(1 + x - \frac{1}{1+x} - 2\,ln(1+x)\right) \tag{87}$$

This result is thermodynamically consistent with the Davies equation.

3.3. The B-dot Equation

The second model for activity coefficients available in EQ3/6 is based on the B-dot equation of Helgeson (1969) for electrically charged species:

$$\log\gamma_i = -\frac{A_\gamma z_i^2 \sqrt{I}}{1 + \mathring{a}_i B_\gamma \sqrt{I}} + \dot{B}I \qquad (88)$$

Here $\mathring{a}_i$ is the hard core diameter of the species, B_γ is the Debye-Hückel B parameter, and $\dot{B}$ is the characteristic B-dot parameter. Like the Davies equation, this is a simple extended Debye-Hückel model, the extension being the "$\dot{B}I$" term. The Debye-Hückel part of this equation is equivalent to that of the Davies equation if the product "$\mathring{a}_i B_\gamma$" has a value of unity. In the extended part, these equations differ in that the Davies equation has a coefficient in place of $\dot{B}$ which depends on the electrical charge of the species in question.

In EQ3/6, the B-dot equation option is selected by setting the option flag **iopg1** = 0. A supporting data file consistent with the use of a simple extended Debye-Hückel model must also be supplied (e.g., **data1** = **data1.com**, **data1.sup**, or **data1.nea**). Note that these data files support the use of the Davies equation as well (the $\dot{B}$ data on these files is simply ignored in that case). If **iopg1** = 0 and the supporting data file is not of the appropriate type, the software terminates with an error message.

The B-dot equation has about the same level of accuracy as the Davies equation, and almost as much universality (one needs to know $\mathring{a}_i$ in addition to z_i). However, it fails to satisfy the solute-solute form of the cross-differentiation rule. The first term is consistent with this rule only if all hard core diameters have the same value. The second is consistent only if all ions share the same value of the square of the electrical charge. However, the numerical significance of the inconsistency is small in the range of low concentrations in which this equation can be applied with useful accuracy. On the positive side, the B-dot equation has been developed (Helgeson, 1969) to span a wide range of temperature (up to 300°C).

For electrically neutral solute species, the B-dot equation reduces to:

$$\log\gamma_i = \dot{B}I \qquad (89)$$

As $\dot{B}$ has positive values at all temperatures in the range of application, the equation predicts a salting out effect. However, by tradition (Helgeson et al., 1970), the B-dot equation itself is not used in the case of neutral solute species. The practice, as suggested by Garrels and Thompson (1962) and reiterated by Helgeson (1969), is to assign the value of the activity coefficient of aqueous CO_2 in otherwise pure sodium chloride solutions of the same ionic strength. This function was represented in previous versions of EQ3/6 by a power series in the ionic strength:

$$\log\gamma_i = k_1 I + k_2 I^2 + k_3 I^3 + k_4 I^4 \qquad (90)$$

The first term on the right hand side dominates the others. The first coefficient is positive, so the activity coefficient of CO_2 increases with increasing ionic strength (consistent with the "salting out" effect). As it was applied in EQ3/6, the coefficients for the power series themselves were represented as similar power series in temperature, and this model was fit to data taken from Ta-

ble 2 of Helgeson (1969). These data (including extrapolations made by Helgeson) covered the range 25-300°C and 0-3 molal *NaCl*.

The high order power series in eq (90) was unfortunately very unstable when extrapolated outside the range of the data to which it was fit. EQ3NR and EQ6 would occasionally run into an unrecoverable problem attempting to evaluate this model for high ionic strength values generated in the process of attempting to find a numerical solution (not necessarily because the solutions in question really had high ionic strength). To eliminate this problem, the high order power series has been replaced by a new expression after Drummond (1981, p. 19):

$$ln\gamma_i = (C + FT + \frac{G}{T})I - (E + HT)(\frac{I}{I+1}) \tag{91}$$

where T is the absolute temperature and C = -1.0312, F = 0.0012806, G = 255.9, E = 0.4445, and H = -0.001606. Note that this is presented in terms of the natural logarithm. Conversion is accomplished by using the relation:

$$log x = \frac{ln\, x}{2.303} \tag{92}$$

This expression is both much simpler (considering the dependencies on both temperature and ionic strength) and is more stable. However, in deriving it, the ionic strength was taken to be equivalent to the sodium chloride molality. In the original model (based on Helgeson, 1969), the ionic strength was based on correcting the sodium chloride molality for ion pairing. This correction is numerically insignificant at low temperature. It does become significant at high temperature. However, neither this expression nor the power series formulation it replaced is thermodynamically consistent with the B-dot equation itself, as can be shown by applying the solute-solute cross-differentiation rule.

The more recent previous versions of EQ3/6 only applied the "CO_2" approximation to species that are essentially nonpolar (e.g., $O_{2(aq)}$, $H_{2(aq)}$, $N_{2(aq)}$), for which salting-out would be expected. In the case of polar neutral aqueous species, the activity coefficients were set to unity (following the recommendation of Garrels and Christ, 1965, p. 70); i.e., one has:

$$log\gamma_i = 0 \tag{93}$$

This practice is still followed in the present version of the code.

EQ3/6 formerly complemented the B-dot equation with an approximation for the activity of water that was based on assigning values in pure sodium chloride solutions of the same "stoichiometric" ionic strength (Helgeson et al., 1970). This approximation was fairly complex and was, of course, not thermodynamically consistent with the B-dot equation itself. In order to simplify the data requirements, as well as avoid the need to employ a second ionic strength function, this formulation has been replaced by a new one which depends on the $\dot{B}$ parameter and is quasi-consistent with the B-dot equation:

$$\log a_w = \frac{1}{\Omega}\left(-\frac{\Sigma m}{2.303} + \frac{2}{3}A_\gamma 10^{I^{\frac{3}{2}}}\sigma(\mathring{a}B_\gamma\sqrt{I}) - BI^2\right) \quad (94)$$

The solute hard core diameter (å) is assigned a fixed value of 4.0Å (a reasonable value). This equation is consistent with the B-dot equation if all solute species are ions, have the same fixed value of the hard core diameter, and have the same value of the square of the electrical charge.

3.4. Scaling of Individual Ionic Activity Coefficients: pH Scales

Before proceeding to a discussion of Pitzer's (1973, 1975) equations, we will address the problem of scaling associated with the activity coefficients of individual ions. It is not possible to observe (measure) any of the thermodynamic functions of such species, because any real solution must be electrically balanced. Thus, the activity coefficients of aqueous ions can only be measured in electrically neutral combinations. These are usually expressed as the mean activity coefficients of neutral electrolytes. The mean activity coefficient of neutral electrolyte *MX* (*M* denoting the cation, *X* the anion) is given by:

$$\log\gamma_{\pm,MX} = \frac{\nu_M \log\gamma_M + \nu_X \log\gamma_X}{\nu_{MX}} \quad (95)$$

where ν_M is the number of moles of cation produced by dissociation of one mole of the electrolyte, ν_X is the number of moles of anion produced, and:

$$\nu_{MX} = \nu_M + \nu_X \quad (96)$$

Electrical neutrality requires that:

$$z_M\nu_M = -z_X\nu_X \quad (97)$$

Although the activity coefficients of ions can not be individually observed, the corresponding molal concentrations can be. The corresponding products, the thermodynamic activities of the ions, are not individually observable, precisely because of the problem with the activity coefficients. Thus, the problem of obtaining individual activity coefficients of the ions and the problem of obtaining individual activities of the same species is really the same problem.

Individual ionic activity coefficients can be defined on a conventional basis by introducing some arbitrary choice. This is can be made by adopting some expression for the activity coefficient of a single ion. The activity coefficients of all other ions then follow via electroneutrality relations. The activities for all the ions are then also determined (cf. Bates and Alfenaar, 1969). Because this applies to the hydrogen ion, such an arbitrary choice then determines the *pH*. Such conventions are usually made precisely for this purpose, and they are generally known as *pH* scales.The NBS *pH* scale, which is the basis of nearly all modern conventional *pH* measurement, is based on the Bates-Guggenheim equation (Bates, 1964):

$$log\gamma_{Cl^-} = \frac{-A_{\gamma,10}\sqrt{I}}{1+1.5\sqrt{I}} \tag{98}$$

This scale is significant not only to the measurement of *pH*, but of corresponding quantities (e.g., *pCl*, *pBr*, *pNa*) obtained using other specific-ion electrodes (cf. Bates and Alfenaar, 1969; Bates, 1973; Bates and Robinson, 1974).

The Bates-Guggenheim equation, like the Davies equation and the B-dot equation, is an extended Debye-Hückel formula. However, if one applies the Davies equation or the B-dot equation to the chloride ion, the result is not precisely identical. The difference approaches zero as the ionic strength approaches zero, and is not very significant quantitatively in the low range of ionic strength in which either the Davies equation or the B-dot equation has useful accuracy. Nevertheless, the use of either of these equations in uncorrected form introduces an inconsistency with measured *pH* values, as use of the Davies equation for example would interpret the *pH* as being on an implied "Davies" scale.

Activity coefficients (and activities) of ions can be moved from one scale to another. The general relation for converting from scale (1) to scale (2) is (Knauss, Wolery, and Jackson, 1991):

$$log\gamma_i^{(2)} = log\gamma_i^{(1)} = \frac{z_i}{z_j}\left(log\gamma_j^{(2)} - log\gamma_j^{(1)}\right) \tag{99}$$

For example, if we evaluate the Davies equation for all ions, we may take the results as being on scale (1). To convert these to the NBS scale (here scale (2)), we take the *j*-th ion to be the chloride ion and evaluate the Bates-Guggenheim equation. We then apply the scale conversion equation to every other ion *i*.

In EQ3/6, activity coefficients are first calculated from the "raw" single-ion equations. They are then immediately rescaled, unless no rescaling is to be done. Thus, rescaling occurs during the iteration process; it is not deferred until convergence has been achieved. The user has control over rescaling via the option switch **iopg2**. If **iopg2** = 0, all single-ion activity coefficients and activities are put on the NBS scale. If **iopg2** = -1, no rescaling is performed. If **iopg2** = 1, all single-ion activity coefficients and activities are put on a scale which is defined by the relation:

$$log\gamma_{H^+} = 0 \tag{100}$$

This has the effect of making the activity and the molality of the hydrogen ion numerically equal. This may have some advantages in comparing with experimental measurements of the hydrogen ion molality. Such measurement techniques have recently been discussed by Mesmer (1991).

The problem of scaling the activity coefficients of ions is more acute in concentrated solutions, and the need to discriminate among different scales in geochemical modeling codes has only been addressed as such codes have been written or modified to treat such solutions (e.g., Harvie, Møller, and Weare, 1984; Plummer et al., 1988).

3.5. Pitzer's Equations

3.5.1. Introduction

Pitzer (1973, 1975) proposed a set of semi-empirical equations to describe activity coefficients in aqueous electrolytes. These equations have proven to be highly successful as a means of dealing with the thermodynamics of concentrated solutions (e.g., Pitzer and Kim, 1974). Models based on these equations have been developed to describe not only solution properties, but also equilibrium between such solutions and salt minerals (e.g., Harvie and Weare, 1980; Harvie, Møller, and Weare, 1984). The utility of these models in geochemical studies has been well established. For example, such models have been shown to account for the mineral sequences produced by evaporation of seawater (Harvie et al., 1980), the process of trona deposition in Lake Magadi, Kenya (Monnin and Schott, 1984), and the formation of the borate-rich evaporite deposits at Searles Lake, California (Felmy and Weare, 1986).

Pitzer's equations are based on a semi-theoretical (see Pitzer, 1973) interpretation of ionic interactions, and are written in terms of interaction coefficients (and parameters from which such coefficients are calculated). There are two main categories of such coefficients, "primitive" ones which appear in the original theoretical equations, but most of which are only observable in certain combinations, and others which are "observable" by virtue of corresponding to observable combinations of the primitive coefficients or by virtue of certain arbitrary conventions. Only the observable coefficients are reported in the literature.

There is a very extensive literature dealing with Pitzer's equations and their application in both interpretation of experimental data and calculational modeling. A complete review is beyond the scope of the present manual. Discussion here will be limited to the equations themselves, how to use them in EQ3/6, and certain salient points that are necessary in order to use them in an informed manner. Readers who wish to pursue the subject further are referred to reviews given by Pitzer (1979, 1987, 1992). Jackson (1988) has addressed the verification of the addition of Pitzer's equations to EQ3/6.

In EQ3/6, the Pitzer's equations option is selected by setting the option flag **iopg1** = 1. A supporting data file consistent with this option must also be supplied (e.g., **data1** = **data1.hmw** or **data1.pit**). If **iopg1** = 1 and the supporting data file is not of the appropriate type, the software terminates with an error message.

Pitzer's equations are based on the following virial expansion for the excess Gibbs energy:

$$G^{EXm} = RT\left(w_w f(I) + \left(\frac{1}{w_w}\right)\sum_{ij}\lambda_{ij}(I)\,n_i n_j + \left(\frac{1}{w_w^2}\right)\sum_{ijk}\mu_{ijk}n_i n_j n_k\right) \qquad (191)$$

where w_w is the number of kilograms of solvent water, $f(I)$ is a Debye-Hückel function describing the long-range electrical interactions to first order, the subscripts i, j, and k denote aqueous solute species, and n_i is the number of moles of the i-th solute species. The equation also contains two kinds of interaction or virial coefficients: the λ_{ij} are second order interaction coefficients, and the μ_{ijk} are third order interaction coefficients. A key element in the success of Pitzer's equations is

the treatment of the second order interaction coefficients as functions of ionic strength. As will be discussed later in more detail, the λ_{ij} consist of both theoretically defined and empirical parts, while the μ_{ijk} are completely empirical. As the term is used here, "empirical" means that at least some of the parameter values required to evaluate a quantity must be obtained by fitting experimental data.

The sums in the interaction coefficient terms are actually double and triple sums. As the number of components in a system increases, the number of interaction coefficients of the type shown above becomes very large. It turns out there are many more of the λ_{ij} and μ_{ijk} than can actually be observed, other than in combination. For example, in the equation for the excess Gibbs energy, it is quickly obvious that λ_{12} and λ_{21} can only be seen in the combination $(\lambda_{12} + \lambda_{21})$, and a similar situation holds in the case of the μ_{ijk}. This leads to the first simplification in dealing with these coefficients, which is that those with the same subscripts (regardless of order or permutation) are required to be equal (Pitzer, 1973). This is not the end of the story, as other constraints (mostly related to electrical neutrality) force even more simplifications (which will be addressed later).

A set of thermodynamically consistent equations for the activity coefficients follows by application of the partial differential equations given previously. In the case of solute species, this leads to:

$$ln\ \gamma_i = \left(\frac{z_i^2}{2}\right) f'(I) + 2\sum_j \lambda_{ij}(I)\, m_j + \sum_{jk}\left(\left(\frac{z_i^2}{2}\right)\lambda'_{jk}(I) + 3\mu_{ijk}\right) m_j m_k \qquad (102)$$

where $f(I)$ is the derivative df/dI and $\lambda'_{ij}(I)$ is similarly $d\lambda_{ij}/dI$. For water, the corresponding result is:

$$ln\ a_w = -\frac{\Sigma m}{\Omega} - \frac{1}{\Omega}\left(If'(I) - f(I)\right)$$

$$+ \frac{1}{\Omega}\left(\sum_{ij}\left(\lambda_{ij}(I) + \lambda'_{ij}(I)\right) m_i m_j + 2\sum_{ijk}\mu_{ijk} m_i m_j m_k\right) \qquad (103)$$

The activity of water is closely related to the osmotic coefficient (ϕ):

$$ln\ a_w = -\left(\frac{\Sigma m}{\Omega}\right)\phi \qquad (104)$$

The thermodynamic properties of water are often discussed in the physical chemistry literature in terms of the osmotic coefficient instead of the activity of water (or the mole fraction activity coefficient of water).

The Debye-Hückel model used in Pitzer's equations is not the usual Debye-Hückel-charging formulation exemplified in the Davies or B-dot equations, but a different one derived by Pitzer (1973) and called the Debye-Hückel-osmotic model. The relevant equations are:

$$f(I) = -\left(\frac{4A_\phi I}{b}\right) ln\ (1 + b\sqrt{I}) \tag{105}$$

$$f'(I) = -2A_\phi \left(\frac{2}{b} ln\ (1 + b\sqrt{I}) + \frac{\sqrt{I}}{(1 + b\sqrt{I})}\right) \tag{106}$$

The Debye-Hückel parameter A_ϕ is related to the more familiar $A_{\gamma,10}$ by:

$$A_\phi = \frac{2.303 A_{\gamma, 10}}{3} \tag{107}$$

The parameter b is assigned a constant value of 1.2 (Pitzer, 1973). Theoretically, this is the product $\mathring{a}B_\gamma$; thus the hard core diameter at 25°C is effectively fixed at a value of about 3.65Å (and somewhat different values at other temperatures). Differences in the hard core diameters of various ions in solution are not explicitly accounted for (this is the case also in the Davies equation). However, the interaction coefficient terms of the equation effectively compensate for this. A very important feature of the Debye-Hückel-osmotic model is that it, like the Debye-Hückel-charging model, is consistent with the Debye-Hückel limiting law:

$$log\gamma_i \rightarrow -A_{\gamma, 10} z_i^2 \sqrt{I} \quad \text{as} \quad I \rightarrow 0 \tag{108}$$

3.5.2. Solutions of Electrolytes

In a pure solution of aqueous neutral electrolyte MX, the following combinations of interaction coefficients are observable:

$$B_{MX}(I) = \lambda_{MX}(I) + \left|\frac{z_X}{2z_M}\right| \lambda_{MM}(I) + \left|\frac{z_M}{2z_X}\right| \lambda_{XX}(I) \tag{109}$$

$$C^\phi_{MX} = 3\left(\left|\frac{z_X}{z_M}\right|^{\frac{1}{2}} \mu_{MMX} + \left|\frac{z_M}{z_X}\right|^{\frac{1}{2}} \mu_{MXX}\right) \tag{110}$$

For example, the osmotic coefficient for such a solution can be written in the form (Pitzer, 1973):

$$\phi - 1 = \frac{|z_M z_X|}{2} (If'(I) - f(I)) + \left(\frac{2\nu_M \nu_X}{\nu_{MX}}\right) B^\phi_{MX}(I)\, m_{MX} + \left(\frac{2(\nu_M \nu_X)^{\frac{3}{2}}}{\nu_{MX}}\right) C^\phi_{MX} m^2_{MX} \tag{111}$$

Appearing in this equation is B^ϕ_{MX}, which is given by:

$$B^{\phi}_{MX}(I) = B_{MX}(I) + IB'_{MX}(I) \tag{112}$$

Here $B'_{MX}(I)$ is the derivative of $B_{MX}(I)$ with respect to the ionic strength.

The ionic strength dependence of B^{ϕ}_{MX} was defined by Pitzer (1973) to take the following form:

$$B^{\phi}_{MX}(I) = \beta^{(0)}_{MX} + \beta^{(1)}_{MX} e^{-\alpha\sqrt{I}} \tag{113}$$

where α was assigned a constant value of 2.0. $\beta^{(0)}_{MX}$ and $\beta^{(1)}_{MX}$, along with C^{ϕ}_{MX}, are parameters whose values are determined by fitting experimental data, such as for the osmotic coefficient. Corresponding to the above equation is:

$$B_{MX}(I) = \beta^{(0)}_{MX} + \beta^{(1)}_{MX} g(\alpha\sqrt{I}) \tag{114}$$

where:

$$g(x) = \left(\frac{2}{x^2}\right)(1 - (1+x)e^{-x}) \tag{115}$$

Pitzer and Mayorga (1974) proposed a description for B^{ϕ}_{MX} in the case of 2:2 electrolytes that is based on an additional fitting parameter:

$$B^{\phi}_{MX}(I) = \beta^{(0)}_{MX} + \beta^{(1)}_{MX} e^{-\alpha_1\sqrt{I}} + \beta^{(2)}_{MX} e^{-\alpha_2\sqrt{I}} \tag{116}$$

Here α_1 is assigned a value of 1.4 and α_2 one of 12.0 and $\beta^{(2)}_{MX}$ is the additional fitting parameter. Corresponding to this is:

$$B_{MX}(I) = \beta^{(0)}_{MX} + \beta^{(1)}_{MX} g(\alpha_1\sqrt{I}) + \beta^{(2)}_{MX} g(\alpha_2\sqrt{I}) \tag{117}$$

We consider first the exponential function in eqs (113) and (117). This is shown in Figure 2 for the three commonly used values of α. At zero ionic strength, this function has a value of unity. Thus, $B^{\phi}_{MX} = \beta^{(0)}_{MX} + \beta^{(1)}_{MX}$ or $B^{\phi}_{MX} = \beta^{(0)}_{MX} + \beta^{(1)}_{MX} + \beta^{(2)}_{MX}$. The magnitude of each term containing $\beta^{(1)}_{MX}$ or $\beta^{(2)}_{MX}$ decreases exponentially as the ionic strength increases, approaching zero as the ionic strength approaches infinity (a limit which is not of physical interest). Most of the decay takes place in the very low ionic strength range. Thus, the terms in $\beta^{(1)}_{MX}$ and $\beta^{(2)}_{MX}$ are important parts of the model, even in dilute solutions.

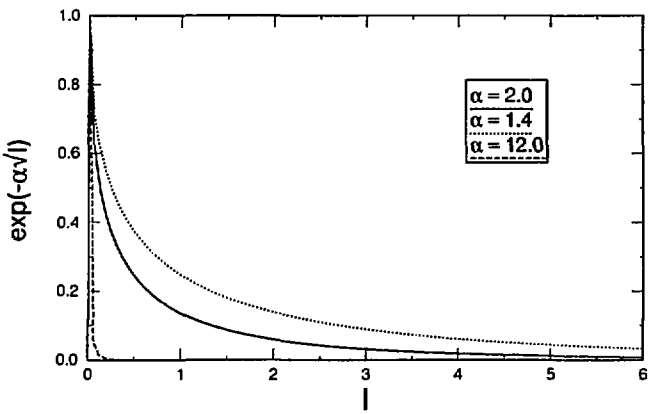

Figure 2. Behavior of the exponential function governing the ionic strength dependence of second-order interactions among cations and anions.

The function $g(x)$ is shown in Figure 3 for the three commonly used α values. It resembles the above exponential function, though it does not decay quite so rapidly. This function may be expanded as follows:

$$g(x) = 1 - 2\left(\frac{2x}{3!} - \frac{3x}{4!} + \frac{4x^2}{5!} - \frac{5x^3}{6!} + \ldots\right) \tag{118}$$

This shows that $g(x) = 1$ at $x = 0$ ($I = 0$). Thus, at zero ionic strength, $B_{MX} = \beta^{(0)}_{MX} + \beta^{(1)}_{MX}$ or $B_{MX} = \beta^{(0)}_{MX} + \beta^{(1)}_{MX} + \beta^{(2)}_{MX}$. It can be shown that $g(x)$ approaches zero as x (and I) approach infinity.

The development thus far shows that there are two major categories of interaction coefficients. The λ_{ij} and the μ_{ijk} in terms of which the theoretical equations were originally derived are what we will call the primitive interaction coefficients. The observable combinations of these, such as $\beta^{(0)}_{MX}$, $\beta^{(1)}_{MX}$, $\beta^{(2)}_{MX}$, and C^{ϕ}_{MX}, are what we will call the observable interaction coefficients. This latter kind of interaction coefficient represents the model data that are reported for the various systems for which Pitzer's equations have been fit to experimental data.

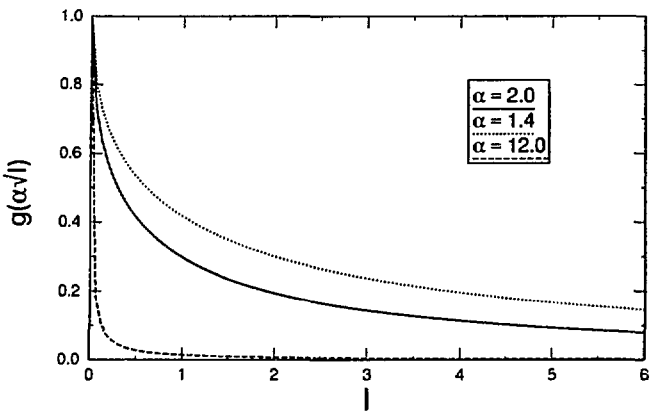

Figure 3. Behavior of the g(x) function governing the ionic strength dependence of second-order interactions among cations and anions.

It is possible to rewrite the equations for $ln\ \gamma_i$ and $ln\ a_w$ in complex mixtures in terms of the observable interaction coefficients. An example of such equations was suggested by Pitzer (1979) and adopted with changes in notation by Harvie, Møller, and Weare (1984). These equations are much more complex than the original form written in terms of the primitive interaction coefficients. They have been incorporated into computer codes, such as that of Harvie, Møller, and Weare (1984), PHRQPITZ (Plummer et al., 1988), and SOLMINEQ.88 (Perkins et al., 1990). As noted in the previous section, there is no unique way to construct equations for single-ion activity coefficients. Furthermore, direct usage of such equations constitutes implicit adoption of a corresponding *pH* scale. In the case of the single-ion activity coefficient equation suggested by Pitzer, this could be termed the "Pitzer" scale.

The equations for $ln\ \gamma_i$ and $ln\ a_w$ which are evaluated in EQ3/6 are those written in terms of the primitive interaction coefficients. The set of these which is used is not the generalized theoretical set, which is not obtainable for the reasons discussed previously, but a practical set that is obtained by mapping the set of reported observable interaction coefficients using a set of equations that contain arbitrary conventions. These mapping equations imply a *pH* scale. We will show that the conventions chosen here match those suggested by Pitzer (1979), so this implied *pH* scale is identical to his.

The basic guides to choosing such mapping conventions are pleasing symmetries and the desirability of minimizing the number of conventional primitive interaction coefficients with non-zero values. In the case of the second order coefficients, both of these considerations suggest the following definitions:

$$\lambda_{MM}(I) = 0 \tag{119}$$

$$\lambda_{XX}(I) = 0 \tag{120}$$

$$\lambda_{MX}(I) = B_{MX}(I) \tag{121}$$

Analogous to the formulas used to describe B_{MX}, one may write:

$$\lambda_{MX}(I) = \lambda_{MX}^{(0)} + \lambda_{MX}^{(1)} g(\alpha\sqrt{I}) \tag{122}$$

or:

$$\lambda_{MX}(I) = \lambda_{MX}^{(0)} + \lambda_{MX}^{(1)} g(\alpha_1\sqrt{I}) + \lambda_{MX}^{(2)} g(\alpha_2\sqrt{I}) \tag{123}$$

From the principle of corresponding terms, it follows that the corresponding mapping equations are:

$$\lambda_{MM}^{(n)} = 0 \quad \text{for } n = 0, 2 \tag{124}$$

$$\lambda_{XX}^{(n)} = 0 \quad \text{for } n = 0, 2 \tag{125}$$

$$\lambda_{MX}^{(n)} = \beta_{MX}^{(n)} \quad \text{for } n = 0, 2 \tag{126}$$

Evaluation of the equations for $ln\ \gamma_i$ and $ln\ a_w$ also requires the ionic strength derivatives of the λ_{ij} coefficients. These are given by:

$$\lambda'_{MX}(I) = \lambda_{MX}^{(1)} g'(x) \left(\frac{\alpha}{2\sqrt{I}}\right) \tag{127}$$

or:

$$\lambda'_{MX}(I) = \lambda_{MX}^{(1)} g'(x) \left(\frac{\alpha_1}{2\sqrt{I}}\right) + \lambda_{MX}^{(2)} g'(x) \left(\frac{\alpha_2}{2\sqrt{I}}\right) \tag{128}$$

where $g'(x)$ is the derivative of $g(x)$ (with respect to x, not I), given by

$$g'(x) = -\left(\frac{4}{x^3}\right)\left(1 - e^{-x}\left(1 + x + \frac{x^2}{2}\right)\right) \tag{129}$$

The principle of pleasing symmetry suggests the following mapping equations for dealing with the C_{MX}^{ϕ} parameter:

$$\mu_{MMX} = \frac{1}{6}\left|\frac{z_M}{z_X}\right|^{\frac{1}{2}} C^{\phi}_{MX} \tag{130}$$

$$\mu_{MXX} = \frac{1}{6}\left|\frac{z_X}{z_M}\right|^{\frac{1}{2}} C^{\phi}_{MX} \tag{131}$$

The two μ coefficients are then related by:

$$\frac{\mu_{MMX}}{z_M} = \frac{\mu_{MXX}}{|z_X|} \tag{132}$$

These are in fact the mapping equations used in EQ3/6. However, the principle of minimizing the number of conventional primitive interaction coefficients would suggest instead mapping relations such as:

$$\mu_{MMX} = \frac{1}{3}\left|\frac{z_M}{z_X}\right|^{\frac{1}{2}} C^{\phi}_{MX} \tag{133}$$

$$\mu_{MXX} = 0 \tag{134}$$

Note that with this set of mapping relations, a different *pH* scale would be implied.

In mixtures of aqueous electrolytes with a common ion, two additional observable combinations of interaction coefficients appear (Pitzer, 1973; Pitzer and Kim, 1974):

$$\theta_{MM'}(I) = \lambda_{MM'}(I) - \left(\frac{z_{M'}}{2z_M}\right)\lambda_{MM}(I) - \left(\frac{z_M}{2z_{M'}}\right)\lambda_{M'M'}(I) \tag{135}$$

and:

$$\Psi_{MM'X} = 6\mu_{MM'X} - \left(\frac{3z_{M'}}{z_M}\right)\mu_{MMX} - \left(\frac{3z_M}{z_{M'}}\right)\mu_{M'M'X} \tag{136}$$

Here M and M' are two cations and X is the anion, or M and M' are two anions and X is the cation. From previously adopted mapping conventions, it immediately follows that the corresponding mappings are given by:

$$\lambda_{MM'}(I) = \theta_{MM'}(I) \tag{137}$$

$$\mu_{MM'X} = \frac{1}{6}\left(\Psi_{MM'X} + \left(\frac{3z_{M'}}{z_M}\right)\mu_{MMX} + \left(\frac{3z_M}{z_{M'}}\right)\mu_{M'M'X}\right) \tag{138}$$

In the original formulation of Pitzer's equations (Pitzer, 1973), the $\theta_{MM'}$ coefficient is treated as a constant. It was later modified by Pitzer (1975) to take the following form:

$$\theta_{MM'}(I) = {}^{S}\theta_{MM'} + {}^{E}\theta_{MM'}(I) \tag{139}$$

$\theta_{MM'}(I)$ corresponds to the Φ_{ij} of Harvie, Møller, and Weare (1984). The first term is a constant and accounts for short-range effects (this is the θ_{ij} of Harvie, Møller, and Weare). The second term, which is the newer part, is entirely theoretical in nature and accounts for higher-order electrostatic effects. Only the ${}^{S}\theta_{MM'}$ part is obtained by fitting. Corresponding to this is the equation:

$$\lambda_{MM'}(I) = {}^{S}\lambda_{MM'} + {}^{E}\lambda_{MM'}(I) \tag{140}$$

The relevant mapping relation is then:

$${}^{S}\lambda_{MM'} = {}^{S}\theta_{MM'} \tag{141}$$

The ${}^{E}\lambda_{MM'}(I)$ part is obtainable directly from theory (Pitzer, 1975):

$${}^{E}\theta_{MM'} = \left(\frac{z_M z_{M'}}{4I}\right)\left(J(x_{MM'}) - \frac{J(x_{MM})}{2} - \frac{J(x_{M'M'})}{2}\right) \tag{142}$$

where:

$$J(x) = \frac{1}{x}\int_0^{\infty}\left(1 + q + \frac{q^2}{2} - e^q\right)y^2 dy \tag{143}$$

in which:

$$q = -\left(\frac{x}{y}\right)e^{-y} \tag{144}$$

and:

$$x_{ij} = 6z_i z_j A^{\phi}\sqrt{I} \tag{145}$$

The derivative of ${}^{E}\lambda_{MM'}(I)$ is given by:

$$E_{\lambda'_{MM'}}(I) = -\left(\frac{E_{\lambda}(I)}{I}\right)$$

$$+\left(\frac{z_M z_{M'}}{8I^2}\right)(x_{MM'}J'(x_{MM'}) - \frac{x_{MM}J'(x_{MM})}{2} - \frac{x_{M'M'}J'(x_{M'M'})}{2}) \tag{146}$$

Expansion of $J(x)$ gives (Pitzer, 1975):

$$J(x) = -(\frac{x^2}{6})(ln\,x + 0.419711) + ... \tag{147}$$

Application of L'Hospital's rule shows that $J(x)$ goes to zero as x goes to zero (hence also as the ionic strength goes to zero). $J(x)$ is a monotonically increasing function. So is $J'(x)$, which approaches a limiting value of 0.25 as x goes to infinity. The function $J(x)$ and its derivative are approximated in EQ3/6 by a Chebyshev polynomial method suggested by Harvie and Weare (1980). This method is described in detail by Harvie (1981, Appendix B, in which $J(x)$ is referred to as $J_0(x)$); this method is also described in the review by Pitzer (1987, p. 131-132).

Pitzer (1979) showed that substitution of the observable interaction coefficients into the single-ion activity coefficient equation gives the following result for cation M:

$$ln\,\gamma_M = z_M^2 f^{\gamma} + 2\sum_a m_a [B_{Ma} + (\Sigma mz)\,C_{Ma}] +$$

$$2\sum_c m_c \theta_{Mc} + \sum_c \sum_a m_c m_a [z_M^2 B'_{ca} + z_M C_{ca} + \Psi_{Mca}] +$$

$$\frac{1}{2}\sum_a \sum_{a'} m_a m_{a'} [z_M^2 \theta'_{aa'} + \Psi_{Maa'}] + \frac{z_M^2}{2}\sum_c \sum_{c'} m_c m_{c'} \theta'_{cc'} +$$

$$z_M \{\sum_c \frac{m_c \lambda_{cc}}{z_c} - \sum_a \frac{m_a \lambda_{aa}}{|z_a|} + \frac{3}{2}\sum_c \sum_a m_c m_a \left(\frac{\mu_{cca}}{z_c} - \frac{\mu_{caa}}{|z_a|}\right)\} \tag{148}$$

Here a denotes anions, c denotes cations, and:

$$f^{\gamma} = \frac{f'}{2} \tag{149}$$

$$C_{MX} = \frac{C^{\phi}_{MX}}{2\sqrt{|z_M z_X|}} \tag{150}$$

$$\Sigma mz = \sum_c m_c m_c \tag{151}$$

(The single-ion equation for an anion is analogous). As pointed out by Pitzer, the unobservability of single-ion activity coefficients in his model lies entirely in the last term (the fourth line) of the equation and involves the primitive interaction coefficients λ_{cc}, λ_{aa}, μ_{cca}, and μ_{caa}. His suggested conventional single-ion activity coefficient equation is obtained by omitting this part. This requires the affected primitive interaction coefficients to be treated exactly as in the previously adopted mapping equations. This approach could in fact have been used to derive them.

In theory, the relevant data required to evaluate Pitzer's equations for complex mixtures of relatively strong aqueous electrolytes can all be obtained from measurements of the properties of pure aqueous electrolytes (giving the observable interaction coefficients $\beta^{(0)}_{MX}$, $\beta^{(1)}_{MX}$, $\beta^{(2)}_{MX}$, and C^{ϕ}_{MX}) and mixtures of two aqueous electrolytes having a common ion ($^{S}\theta_{MM'}$ and $\Psi_{MM'X}$). There is one peculiarity in this fitting scheme in that $^{S}\theta_{MM'}$ is obtainable from more than one mixture of two electrolytes having a common ion, because this parameter does not in theory depend on that ion. Thus, the value adopted may have to be arrived at by simultaneously considering the experimental data for a suite of such mixtures.

3.5.3. Solutions of Electrolytes and Nonelectrolytes

In general, it is necessary to consider the case of solutions containing nonelectrolyte solute species in addition to ionic species. Examples of such uncharged species include molecular species such as $O_{2(aq)}$, $CO_{2(aq)}$, $CH_{4(aq)}$, $H_2S_{(aq)}$, $C_2H_5OH_{(aq)}$, and $SiO_{2(aq)}$; strongly bound complexes, such as $HgCl_{3(aq)}$ and $UO_2CO_{3(aq)}$; and weakly bound ion pairs such as $CaCO_{3(aq)}$ and $CaSO_{4(aq)}$. The theoretical treatment of these kinds of uncharged species is basically the same. There are practical differences, however, in fitting the models to experimental data. This is simplest for the case of molecular neutral species. In the case of complexes or ion pairs, the models are complicated by the addition of corresponding mass action equations.

The treatment of solutions of electrolytes using Pitzer's equations is quite standardized. In such solutions, there is one generally accepted relation for describing single-ion activity coefficients, though it may be expressed in various equivalent forms. Thus, in such solutions there is only one implied "Pitzer" pH scale. Also, the set of parameters to be obtained by regressing experimental measurements is well established. Unfortunately, this is not the case for the treatment of solutions containing both electrolytes and nonelectrolytes.

Harvie, Møller, and Weare (1984) used Pitzer's equations to construct a model of all of the major components of seawater at 25°C. They modified the equations for electrolyte systems to include some provision for neutral species-ion interactions. Additional modification was made by Felmy and Weare (1986), who extended the Harvie, Møller, and Weare model to include borate as a component. The Felmy and Weare equation for the activity of water (obtained from their equation for the osmotic coefficient) is:

$$\ln a_w = -\frac{\Sigma m}{\Omega} - \frac{2}{\Omega}\left(\frac{If'-f}{2} + \sum_c\sum_a m_c m_a (B^{\phi}_{ca} + ZC_{ca})\right.$$

$$+ \sum_c \sum_{c'>c} m_c m_{c'} (\Phi^{\phi}_{cc'} + \sum_a m_a \Psi_{aa'c})$$

$$+ \sum_a \sum_{a'>a} m_a m_{a'} (\Phi^{\phi}_{aa'} + \sum_c m_c \Psi_{cc'a})$$

$$\left. + \sum_n\sum_c m_n m_c \lambda_{nc} + \sum_n\sum_a m_n m_a \lambda_{na} + \sum_n\sum_c\sum_a m_n m_c m_a \zeta_{nca}\right) \tag{152}$$

In this equation, c denotes a cation and a an anion, and the following definitions are introduced:

$$Z = \sum_i |z_i| m_i \tag{153}$$

$$\Phi^{\phi}_{ij} = {}^{S}\theta_{ij} + {}^{E}\theta_{ij}(I) + I\,{}^{E}\theta'_{ij}(I) \tag{154}$$

The first three lines are equivalent to the mixture formulation given by Pitzer (1979). The fourth line (last three terms) is the new part. Here n denotes a neutral species, λ_{nc} and λ_{na} are second order interaction coefficients describing neutral species-ion interactions, and ζ_{nca} is an observable third order coefficient. These new interaction coefficients are treated as constants. The terms in λ_{nc} and λ_{na} were introduced by Harvie, Møller, and Weare (1984) to treat the species $CO_{2(aq)}$. The term in ζ_{nca} was put in by Felmy and Weare (1986) and is a third order interaction coefficient. It was necessary to include it in the equations to account for interactions involving the species $B(OH)_{3(aq)}$.

The corresponding single-ion equation for cation M takes the following form:

$$\ln \gamma_M = z_M^2 F + \sum_a m_a [2B_{Ma} + ZC_{Ma}]$$

$$+ \sum_c m_c (2\Phi_{Mc} + \sum_a m_a \Psi_{Mca})$$

$$+ \sum_a \sum_{a'>a} m_a m_{a'} \Psi_{Maa'} + |z_M| \sum_c\sum_a m_c m_a C_{ca}$$

$$+ 2\sum_n m_n \lambda_{nM} + \sum_n\sum_a m_n m_a \zeta_{naM} \tag{155}$$

Here Φ_{ij} is the θ_{ij} of the earlier notation and:

$$F = \frac{f'}{2} + \sum_c \sum_a m_c m_a B'_{ca} + \sum_c \sum_{c'>c} m_c m_{c'} \Phi'_{cc'}$$

$$+ \sum_a \sum_{a'>a} m_a m_{a'} \Phi'_{aa'} \qquad (156)$$

The first three lines are equivalent to Pitzer's suggested single-ion activity coefficient equation. The fourth line (last two terms) is the new part. The corresponding equation for anions is analogous. The corresponding equation for the N-th neutral species is:

$$ln\,\gamma_N = 2\sum_c m_c \lambda_{Nc} + 2\sum_a m_a \lambda_{Na} + \sum_c \sum_a m_c m_a \zeta_{Nca} \qquad (157)$$

To deal with the fact that the λ_{nc} and λ_{na} are only observable in combination, Harvie, Møller, and Weare (1984) adopted the convention that:

$$\lambda_{N,H^+} = 0 \qquad (158)$$

These equations were presented for the modeling of specific systems, and are not completely general. They are missing some terms describing interactions involving neutral species. A set of complete equations is given by Clegg and Brimblecombe (1990). Their equation for the activity coefficient of a neutral solute species is:

$$ln\,\gamma_N = 2\sum_n m_n \lambda_{Nn} + 2\sum_c m_c \lambda_{Nc} + 2\sum_a m_a \lambda_{Na}$$

$$+ 6\sum_n \sum_c m_n m_c \mu_{Nnc} + 6\sum_n \sum_a m_n m_a \mu_{Nna}$$

$$+ 3\sum_c m_c^2 \mu_{Ncc} + 3\sum_a m_a^2 \mu_{Naa} + 6\sum_c \sum_a m_c m_a \mu_{Nca}$$

$$+ 6\sum_c \sum_{c'>c} m_c m_{c'} \mu_{Ncc'} + 6\sum_a \sum_{a'>a} m_a m_{a'} \mu_{Naa'}$$

$$+ 3\sum_n m_n^2 \mu_{Nnn} + 6\sum_{n \neq N} m_N m_n \mu_{NNn} + 6\sum_{n \neq N} \sum_{n' \neq N} m_n m_{n'} \mu_{Nnn'} \qquad (159)$$

This is a complete and general representation of the activity coefficient of a neutral species in terms of all possible second order and third order primitive interaction coefficients. The first line of this equation contains the same terms in λ_{nc} and λ_{na} as appear in the Felmy-Weare equation. This line is augmented by an addition term which describes second order interactions among neutral species (and which was also pointed out by Pitzer, 1987). The third line in this equation is

equivalent to the term in ζ_{Nca} that appears in the Felmy-Weare equation. Clegg and Brimblecombe (1990) have pointed out that this observable interaction coefficient is related to the corresponding primitive interaction coefficients by the relation:

$$\zeta_{NMX} = 6\mu_{NMX} + \frac{3|z_X|}{z_M}\mu_{NMM} + \frac{3z_M}{|z_X|}\mu_{NXX} \tag{160}$$

The second, fourth, and fifth lines consist of terms not found in the Felmy-Weare equation.

In a solution of a pure aqueous nonelectrolyte, the activity coefficient of the neutral species takes the form:

$$ln\,\gamma_N = 2m_N\lambda_{NN} + 3m_N^2\mu_{NNN} \tag{161}$$

This activity coefficient is directly observable. Hence the two interaction coefficients on the right hand side are also observable. In a study of the solubility of aqueous ammonia, Clegg and Brimblecombe (1989) found that the term including μ_{NNN} was significant only for concentrations greater than 25 molal (a solution containing more ammonia than water). They therefore dropped this term and reported model results only in terms of λ_{NN}. Similarly, Barta and Bradley (1985) found no need for a μ_{NNN} term to explain the data for pure solutions of $CO_{2(aq)}$, $H_2S_{(aq)}$, and $CH_{4(aq)}$, and no such term was apparently required by Felmy and Weare (1986) to explain the data for $B(OH)_{3(aq)}$. Pitzer and Silvester (1976) report a significant μ_{NNN} term for undissociated phosphoric acid. This result now appears somewhat anomalous and has not been explained. The bulk of the available data, however, suggest that the μ_{NNN} term is generally insignificant in most systems of geochemical interest and can be ignored without loss of accuracy.

This result suggests that in more complex solutions, terms in $\lambda_{NN'}$, $\mu_{NNN'}$, $\mu_{N'N'N}$, and $\mu_{NN'N''}$ can also often be ignored. While there may be solutions in which the full complement of these terms are significant, one could argue that they must be so concentrated in nonelectrolyte components that they have little relevance to the study of surface waters and shallow crustal fluids (though some deep crustal fluids are rich in CO_2). Furthermore, one could argue that to address such solutions, it would be more appropriate to use a formalism based on a different kind of expansion than the one used in the present treatment (see Pabalan and Pitzer, 1990).

In an aqueous solution consisting of one nonelectrolyte and one electrolyte, the activity coefficient of the neutral species takes the form:

$$ln\,\gamma_N = 2m_N\lambda_{NN} + 2(m_M\lambda_{NM} + m_X\lambda_{NX})$$
$$+ 6m_N(m_M\mu_{NNM} + m_X\mu_{NNX}) + m_M m_X\zeta_{NMX} + 3m_N^2\mu_{NNN} \tag{162}$$

Three new terms appear. The resemblance of the term in λ_{NM} and λ_{NX} to a traditional Setchenow term has been pointed out by various workers (e.g., Felmy and Weare, 1986; Pitzer, 1987). Work reported by Clegg and Brimblecombe (1989, 1990) for a number of such systems containing am-

monia showed that the most important of the three new terms were the second (λ_{NM}, λ_{NX}) term and the third (μ_{NNM}, μ_{NNX}) term. They defined these using the following conventions:

$$\lambda_{N,Cl^-} = 0 \tag{163}$$

$$\mu_{N,N,Cl^-} = 0 \tag{164}$$

Note that the first of these conventions conflicts with the corresponding convention adopted by Harvie, Møller, and Weare (1984), though it matches that proposed by Pitzer and Silvester (1976) in a study of the dissociation of phosphoric acid, a weak electrolyte. Clegg and Brimblecombe found that in one system, the use of the fourth (ζ_{NMX}) term was also required, though the contribution was relatively small. No use was required of the last (μ_{NNN}) term, as was shown by fitting the data for pure aqueous ammonia.

There seems to be some disagreement in the literature regarding the above picture of the relative significance of the (μ_{NNM}, μ_{NNX}) term versus that of the ζ_{NMX} term, although the seemingly contradictory results involve nonelectrolytes other than ammonia. We have noted above that Felmy and Weare (1986) used a ζ_{NMX} term to explain the behavior of boric acid-electrolyte mixtures. It is not clear if they considered the possibility of a (μ_{NNM}, μ_{NNX}) term. Pitzer and Silvester (1976) found no apparent need to include a (μ_{NNM}, μ_{NNX}) term or a ζ_{NMX} term to explain the thermodynamics of phosphoric acid dissociation in electrolyte solutions. The data on aqueous silica in electrolyte solutions of Chen and Marshall (1981), discussed by Pitzer (1987), require a ζ_{NMX} term, but no (μ_{NNM}, μ_{NNX}) term. A similar result was obtained by Barta and Bradley (1985) for mixtures of electrolytes with $CO_{2(aq)}$, $H_2S_{(aq)}$, and $CH_{4(aq)}$. Simonson et al. (1987) interpret data for mixtures of boric acid with sodium borate and sodium chloride and of boric acid with potassium borate and potassium chloride exclusively in terms of the first (λ_{NN}) and second (λ_{NM}, λ_{NX}) terms, using neither of the third order terms for nonelectrolyte-electrolyte interactions.

The (μ_{NNM}, μ_{NNX}) term can only be observed (and hence is only significant) when the concentrations of both the nonelectrolyte and the electrolyte are sufficiently high. In contrast, evaluating the ζ_{NMX} term requires data for high concentrations of the electrolyte, but low concentrations of the nonelectrolyte will suffice. Some nonelectrolytes, such as aqueous silica, are limited to low concentrations by solubility constraints. Thus, the results of Chen and Marshall (1981) noted by Pitzer (1987) are not surprising. In the case of more soluble nonelectrolytes, the range of the available experimental data could preclude the evaluation of the (μ_{NNM}, μ_{NNX}) term. This may be why Pitzer and Silvester (1976) reported no need for such a term to describe the data for mixtures of electrolytes with phosphoric acid and why Barta and Bradley (1985) found no need for such a term for similar mixtures of electrolytes with $CO_{2(aq)}$, $H_2S_{(aq)}$, and $CH_{4(aq)}$. The data analyzed by Felmy and Weare (1986) correspond to boric acid concentrations of about one molal, which may not be high to observe this term (or require its use). In the case of Simonson et al. (1987), who also looked at mixtures of electrolytes and boric acid, the need for no third order terms describing nonelectrolyte-electrolyte interactions is clearly due to the fact that the concentrations of boric acid were kept low to avoid the formation of polyborate species.

The equations for solutions containing nonelectrolytes can be considerably simplified if the model parameters are restricted to those pertaining to solutions of pure aqueous nonelectrolytes and mixtures of one nonelectrolyte and one electrolyte. This is analogous to the usual restriction in treating electrolyte solutions, in which the parameters are restricted to those pertaining to solutions of two electrolytes with a common ion. Furthermore, it seems appropriate as well to drop the terms in μ_{NNN}. The equation for the activity coefficient of a neutral electrolyte in electrolyte-nonelectrolyte mixtures then becomes:

$$\ln \gamma_N = 2m_N\lambda_{NN} + 2\left(\sum_c m_c\lambda_{Nc} + \sum_a m_a\lambda_{Na}\right)$$

$$+ 6m_N\left(\sum_c m_c\mu_{NNc} + \sum_a m_a\mu_{NNa}\right) + \sum_c\sum_a m_c m_a\zeta_{Nca} \qquad (165)$$

The reduction in complexity is substantial. In the context of using Pitzer's equations in geochemical modeling codes, this level of complexity is probably quite adequate for dealing with nonelectrolytes in a wide range of application.

If a higher level of complexity is required, the next step is probably to add in terms in μ_{NNN} and $\lambda_{NN'}$. The first of these has been discussed previously and is obtained from data on pure aqueous nonelectrolytes. The second must be obtained from mixtures of two aqueous electrolytes (one could argue that this is also analogous to the treatment of electrolytes). This higher level of complexity may suffice to deal with at least some CO_2-rich deep crustal fluids and perhaps other fluids of interest in chemical engineering. However, an even higher level of complexity would probably be best addressed by a formalism based on an alternate expansion, as noted earlier.

The observability and mapping issues pertaining to the remaining parameters may be dealt with as follows. In the case of $\lambda_{NN'}$ no mapping relation is required because this parameter is directly observable. The same is true of μ_{NNN} and $\lambda_{NN'}$, if the higher level of complexity is required.

The λ_{NM} and λ_{NX}, and μ_{NNM} and μ_{NNX}, are only observable in combinations, but can be dealt with by adopting the following respective conventions:

$$\lambda_{N,J} = 0 \qquad (166)$$

$$\mu_{N,N,J} = 0 \qquad (167)$$

where J is a reference ion ($J = H^+$ as suggested by Felmy and Weare, 1986; $J = Cl^-$ as suggested by Pitzer and Silvester, 1976, and Clegg and Brimblecombe, 1989, 1990). In any data file used to support code calculations, the choice of reference ion must be consistent. This may require the recalculation of some published data.

The ζ_{NMX} parameter is observable and can be mapped into primitive form by adopting the following conventions:

$$\mu_{NMM} = 0 \qquad (168)$$

$$\mu_{NXX} = 0 \tag{169}$$

$$\mu_{NMX} = \frac{\zeta_{NMX}}{6} \tag{170}$$

These relations are analogous to those defined for the C^{ϕ}_{MX} parameter.

The above conventions correspond well with the current literature on the subject. However, the treatment of the λ_{NM} and λ_{NX}, and μ_{NNM} and μ_{NNX}, though valid and functional, still stands out in that it is not analogous to, or a natural extension of, the conventions which have been universally adopted in the treatment of electrolyte solutions. The logical extension, of course, is to define observable interaction coefficients to represent the primitive coefficients which can only be observed in combination, and to then follow Pitzer (1979) in determining exactly which parts of the theoretical equations constitute the non-observable part. The conventions would then be defined so as to make these parts have zero value.

The suggested process can be shown to be consistent with the above mapping conventions for all the other coefficients treated above, including ζ_{NMX}. However, the process which worked so nicely for electrolytes fails to work for λ_{NM}-λ_{NX}, and μ_{NNM}-μ_{NNX}. We will demonstrate this for the case of the λ_{NM}-λ_{NX}. Application of the above equation to the case of an aqueous mixture of a neutral species (N)and a neutral electrolyte (MX) immediately shows that the corresponding observable combination of primitive interaction coefficients is given by:

$$L_{NMX} = |z_X|\lambda_{NM} + z_M\lambda_{NX} \tag{171}$$

In such a system, the activity coefficient of the neutral species can be written as:

$$ln\,\gamma_N = \frac{2}{z_M + |z_X|}\nu_{MX}L_{NMX}m_{MX} \tag{172}$$

In the manner of Pitzer (1979), one can show that the relevant term in the single-ion activity coefficient for cation M expands in the following manner:

$$2\sum_n m_n\lambda_{nM} = 2\sum_n m_n\frac{L_{nMX'}}{|z_{X'}|} - 2\sum_n m_n\frac{z_M}{|z_{X'}|}\lambda_{nX'} \tag{173}$$

where X' is some reference anion. When X' is Cl^-, we have the convention proposed by Pitzer and Silvester (1976) and followed by Clegg and Brimblecombe (1989, 1990). The first term on the right hand side is the relevant observable part; the second term is the non-observable part. Following the logic of Pitzer (1979), we could set the second term to zero. This would have the effect of defining the following mapping relations:

$$\lambda_{NX'} = 0 \tag{174}$$

$$\lambda_{NM} = \frac{L_{NMX'}}{|z_{X'}|} \tag{175}$$

Although this makes the relevant non-observable part vanish in the single-ion activity coefficient equation for all cations, it forces the complementary part in the corresponding equation for anions to not vanish, as we will now show. The relevant part of the anion equation gives the following analogous result:

$$2\sum_n m_n \lambda_{nX} = 2\sum_n m_n \frac{L_{nM'X}}{z_{M'}} - 2\sum_n m_n \frac{|z_X|}{z_{M'}} \lambda_{nM'} \tag{176}$$

where M' is some reference cation. As before, the second term on the right hand side is the non-observable part. Using the above mapping equation for λ_{NM}, this can be transformed to:

$$2\sum_n m_n \lambda_{nX} = 2\sum_n m_n \frac{L_{nM'X}}{z_{M'}} - 2\sum_n m_n \frac{z_X}{z_{M'} z_{X'}} L_{nM'X'} \tag{177}$$

Thus, under the conventions defined above, the non-observable part of the single-ion activity coefficient equation for anions does not vanish.

There are alternatives, but none are particularly outstanding. For example, one could reverse the situation and make analogous conventions so that the non-observable part of the anion equation vanishes, but then the non-observable part of the cation equation would not vanish. When M' is H^+, we have the convention proposed by Felmy and Weare (1986). One could also try a symmetrical mapping, based on the following relation:

$$\lambda_{NM} = \frac{z_M}{|z_X|} \lambda_{NX} \tag{178}$$

This would lead to the following mapping relations:

$$\lambda_{NM} = \frac{z_M}{(z_M + |z_X|)} L_{NMX} \tag{179}$$

$$\lambda_{NX} = \frac{|z_X|}{(z_M + |z_X|)} L_{NMX} \tag{180}$$

Unfortunately, this would lead to a non-vanishing non-observable part in the equations for both cations and anions.

3.5.4. Temperature and Pressure Dependence

Pitzer's equations were originally developed and applied to conditions of 25°C and atmospheric pressure (e.g., Pitzer and Kim, 1974). The formalism was subsequently applied both to activity coefficients under other conditions and also to related thermodynamic properties which reflect

the temperature and pressure dependence of the activity coefficients (see the review by Pitzer, 1987).

The first effort to extend the Pitzer formalism to high temperature was a detailed study of the properties of aqueous sodium chloride (Silvester and Pitzer, 1977). In this study, the data were fit to a complex temperature function with up to 21 parameters per observable interaction coefficient and which appears not to have been applied to any other system. In general, the early efforts concerning the temperature dependence of the activity coefficients focused mainly on estimating the first derivatives of the observable interaction coefficient parameters with respect to temperature (e.g., Silvester and Pitzer, 1978). The results of the more detailed study of sodium chloride by Silvester and Pitzer (1977; see their Figures 4, 5, and 6) suggest that these first derivatives provide an extrapolation that is reasonably accurate up to about 100°C.

In more recent work, the temperature dependence has been expressed in various studies by a variety of different temperature functions, most of which require only 5-7 parameters per observable interaction coefficient. Pabalan and Pitzer (1987) used such equations to develop a model for the system *Na-K-Mg-Cl-SO_4-OH-H_2O* which appears to be generally valid up to about 200°C. Pabalan and Pitzer (1988) used equations of this type to built a model for the system *Na-Cl-SO_4-OH-H_2O* that extends to 300°C. Greenberg and Møller (1989), using an elaborate compound temperature function, have constructed a model for the *Na-K-Ca-Cl-SO_4-H_2O* system that is valid from 0-250°C. More recently, Spencer, Møller, and Weare (1990) have used a more compact equation to develop a model for the *Na-K-Ca-Mg-Cl-SO_4-H_2O* system at temperatures in the range -60 to 25°C.

The pressure dependence of activity coefficients has also been looked at in the context of the Pitzer formalism. For descri ptions of recent work, see Kumar (1986), Connaughton, Millero, and Pitzer (1989), and Monnin (1989).

3.5.5. Practical Aspects

In practice, the matter of obtaining values for the observable interaction coefficients is more complicated. Not all models based on Pitzer's equations are mutually consistent. Mixing reported data can lead to inconsistencies. For the most part, differences in reported values for the same coefficient are functions of the exact data chosen for use in the fitting process, not just whose data, but what kind or kinds of data as well. Some older reported values for the mixture parameters (e.g., Pitzer, 1979) are based on fits not employing the ${}^{E}\theta_{MM'}$ formalism, which has become firmly entrenched in more recent work.

Some differences in the values of reported Pitzer parameters are due to minor differences in the values used for the A^{Φ} Debye-Hückel parameter (e.g., 0.39 versus 0.392; see Plummer et al., 1988, p. 3, or Plummer and Parkhurst, 1990). The general problem of minor discrepancies in this and other limiting law slope parameters has been looked at in some detail by Ananthaswamy and Atkinson (1984). Recently, Archer (1990) has also looked at this problem and proposed a method for adjusting reported Pitzer coefficients for minor changes in Debye-Hückel parameters without resorting to refitting the original experimental data.

There has also been some occasional modification of the basic activity coefficient equations themselves. For example, in treating the activity coefficients of alkali sulfate salts at high temperature, Holmes and Mesmer (1986a, 1986b) changed the recommended value of the α parameter from 2.0 to 1.4. Also Kodytek and Dolejs (1986) have proposed a more widespread usage of the $\beta^{(2)}_{MX}$ parameter, based on the empirical grounds that better fits can be obtained for some systems. The usage of this parameter was originally restricted to the treatment of 2:2 electrolytes (Pitzer and Mayorga, 1973).

The formal treatment of speciation in the solutions (assumptions of which species are present) can also lead to different models. Association phenomena were first recognized in the Pitzer formalism in order to deal with phosphoric acid (Pitzer and Silvester, 1976) and sulfuric acid (Pitzer, Roy, and Silvester, 1977). In general, ion pairs have been treated formally as non-existent. An exception is in the model of Harvie, Møller, and Weare (1984), who employ three ion pair species: $CaCO_{3(aq)}$, $MgCO_{3(aq)}$, and $MgOH^+$.

Components which form strong complexes have received relatively little attention in the Pitzer formalism, presumably because of the much greater experimental data requirements necessary to evaluate the greater number of parameters associated with the greater number of species. However, Millero and Byrne (1984) have used Pitzer's equations to develop a model of activity coefficients and the formation of lead chloro complexes in some concentrated electrolyte solutions. Huang (1989) has also recently looked at some examples of complex formation in the context of the Pitzer formalism. However, because strong complexing can not be represented even mathematically by the interaction coefficient formalism without taking explicit account of the associated chemical equilibria, and because such models are more difficult to develop, the practical application of the Pitzer formalism remains limited mostly to systems of relatively strong electrolytes, molecular nonelectrolytes, and a few weak nonelectrolytes.

3.5.6. Pitzer's Equations in EQ3/6: Current Status

The present treatment of Pitzer's equations in EQ3/6 is somewhat limited, particularly in regard to some of the advances that have been made with these equations in the past few years. These limitations have to do with the state of the existing data files which support the use of Pitzer's equations, the treatment of the temperature dependence of the interaction coefficients, and the treatment of neutral solute species.

The **hmw** data file is an implementation of the model of Harvie, Møller, and Weare (1984). This model is restricted to 25°C. The **pit** data file is based mostly on the data summarized by Pitzer (1979). These data include the first order temperature derivatives of the interaction coefficients. The nominal temperature range of this data file is 0-100°C. These data are not based on the currently universally accepted $^E\theta$ formalism introduced by Pitzer (1975).

EQ3/6 uses or ignores the $^E\theta$ formalism, depending on the value of a flag parameter on the data file. The temperature dependence, if any, is handled by using first and second order temperature derivatives of the interaction coefficients, which are expected for use at temperatures other than 25°C. The code permits a $\beta^{(2)}_{MX}$ parameter to be specified on the data file for any electrolyte. The

α parameters are also provided on the data file for each electrolyte. Thus, non-standard values can be employed if desired.

The temperature dependence is presently limited to a representation in terms of a second-order Taylor's series in temperature. This requires the presence on the supporting data file of first and second temperature derivatives (see the EQPT User's Guide, Daveler and Wolery, 1992). No provision has yet been made for the more sophisticated representations proposed for example by Pabalan and Pitzer (1987) or Spencer, Møller, and Weare (1990).

EQ3/6 is presently quite limited in terms of the treatment of nonelectrolyte components by means of Pitzer's equations. This limitation is expressed in the structure of the data files and the mapping relations presently built into the EQPT data file preprocessor. These are presently set up to deal only with electrolyte parameters. However, it is possible to enter λ_{NN}, $\lambda_{NN'}$, λ_{NM}, and λ_{NX} parameters as though they were $\beta^{(0)}_{MX}$ parameters. The λ_{NM} and λ_{NX} parameters that are part of the model of Harvie, Møller, and Weare (1984) are included on the **hmw** data file in this manner.

The present version of EQPT can not handle the ζ_{MNX} interaction coefficient; however.

The means of storing and representing interaction coefficient data in EQ3/6 deserves some comment. There is a natural tendency to represent λ_{ij} by a two-dimensional array, and μ_{ijk} by a three-dimensional array. However, arrays of this type would be sparse (for example, $\lambda_{ij} = 0$ for many i, j). and many of the entries would be duplicates of others ($\lambda_{ij} = \lambda_{ji}$, etc.). Therefore, the λ_{ij} are represented instead by three parallel one-dimensional arrays. The first contains the λ_{ij} values themselves, the second contains indices identifying the i-th species, and the third identifies the j-th species. The treatment is analogous for μ_{ijk}, which only requires an additional array to identify the k-th species. These arrays are constructed from data listed on the **data0** data files. Coefficients which must be zero by virtue of the mapping relations or other conventions are not included in the constructed arrays. Also, the storage scheme treats for example λ_{ij} and λ_{ji} as one coefficient, not two.

4. Activity Coefficients of Solid Solution Components

4.1. Introduction

The thermodynamic activities (a_i) of solid solution components are always defined on the basis of mole fractions. Thus, they can be described by the product of their mole fractions (x_i) and their rational (mole fraction) activity coefficients (λ_i):

$$a_i = x_i \lambda_i \tag{181}$$

The same treatment is typically applied to all components in non-aqueous liquid phases. It is also applied to water in aqueous solutions (cf. Chapter 3).

Mole fraction ideality is the reference ideality when dealing with solid solutions. Therefore, the corresponding excess Gibbs energy is G^{EXx} (see Chapter 3). The relevant differential equation linking this with the mole fraction activity coefficients is:

$$ln\ \lambda_i = \frac{1}{RT} \frac{\partial G^{EXx}}{\partial n_i} \tag{182}$$

where R is the gas constant and T the absolute temperature. Given an expression for the excess Gibbs energy, this equation gives a guaranteed route to thermodynamically consistent results (cf. Wolery, 1990).

Problems involving the thermodynamic consistency of activity coefficients in non-aqueous phases seem to be uncommon. However, consistency may be tested using various relations, such as the following form of the cross-differentiation rule (cf. Wolery, 1990):

$$\frac{\partial\ ln\ \lambda_j}{\partial n_i} = \frac{\partial\ ln\ \lambda_i}{\partial n_j} \tag{183}$$

The issue of sufficiency in proving consistency using this and related equations (Gibbs-Duhem equations and sum rules) is addressed by Wolery (1990).

In most speciation-solubility calculations, the activity coefficients of solid solution components only affect the corresponding calculated saturation indices; they do not change the model of the aqueous solution itself (i.e., the speciation). However, if an equilibrium relation involving a solid solution phase is used as a constraint in defining a speciation-solubility problem, all of the model results may be affected by the choice of activity coefficient model. The results may similarly affected when such a constraint is used in mass transfer calculations, including reaction path calculations.

Mixing tends to stabilize a solid solution relative to its end-member components. Thus, an aqueous solution may be supersaturated with respect to a solid solution, yet undersaturated with respect to each of the pure end members. Consequently, a solid solution may form in a system in which some or none of the pure end members would form. This effect is true in the ideal case, in

which the activity coefficient has a value of unity. If the activity coefficient is less than unity, this stabilizing effect is increased. If it is greater than unity, it is lessened.

In general, the activity coefficient of a solid solution component depends on the composition of the solid solution. This is in turn is normally expressed in terms of the mole fractions of the components. In order to use an equilibrium constraint involving a solid solution component in a speciation-solubility calculation, the user must provide this composition in order to allow calculation of the activity coefficient of the component involved. In mass transfer calculations involving solid solutions in equilibrium with an aqueous solution, the solid solution composition is itself a subset of the unknowns to be calculated. In speciation-solubility and other kinds of equilibrium calculations, it is necessary to calculate a saturation index for a solid solution which is not presumed to be in equilibrium with the aqueous phase. This presents a problem, because no composition is defined. This is solved in EQ3/6 by finding the composition which maximizes the computed saturation index (Bourcier, 1985, 1989).

In liquid solutions, the solutes may mix over the whole volume of the solution. This type of mixing is sometimes referred to as molecular mixing. It is commonly applied to non-aqueous liquids, such as a solution composed of hydrocarbons. In the ideal case, the activity coefficient of each component is unity. In aqueous solutions and other solutions involving a solvent with a high dielectric constant, this concept is modified to account for ionic dissociation. The concept of molecular mixing has been applied many times to solid solutions (cf. the examples presented later in this chapter), and is predicated on the use of end-member components (for example, calcite [$CaCO_3$] and magnesite [$MgCO_3$] in magnesian calcite [$(Ca,Mg)CO_3$]). The activity coefficients of these components in the non-ideal case are then described using interaction coefficients more or less resembling those used in Pitzer's equations to describe the activity coefficients of aqueous species.

In crystalline solids, mixing tends rather strongly to occur over well-defined sites in the crystal structure (see for example Wood and Fraser, 1977, or Nordstrom and Munoz, 1985). Some ions may mix over more than one kind of site. Vacancies may be involved in the mixing process. They may be created or destroyed by substitutions of one ion for another of different electrical charge. Mixing which takes account of such effects is referred to as site mixing. In site-mixing models, the concept of ideality is modified from that appropriate to molecular mixing, though still based on the mole fractions of components. It is possible to utilize as the components species such as ions, vacancies, and framework moieties instead of end members. However, the more common practice is to continue using end-member components. This is followed in the present version of EQ3/6. However, the activity coefficient of an end-member component in an ideal site mixing model may have a value other than unity. A site-mixing model will appear to be ideal in this sense only if there is only one site, an ion substitutes for others of the same charge type, and vacancies are not present on the site. Site mixing then effectively reduces to molecular mixing.

Nearly all of the site-mixing models that have been proposed for the various solid solutions are ideal in the site-mixing sense (see for example Viani and Bruton, 1992). The only parameters of such models are site-mixing parameters. It is possible to consider site-mixing models which are non-ideal even in the site-mixing sense. These would be described by both site-mixing parameters and interaction coefficients. No models of this type are presently treated in EQ3/6.

In EQ3/6, all solid solution models are defined on the supporting data file (see Chapters 3 and 4 of the EQPT User's Guide, Daveler and Wolery, 1992). At present (through the R16 set of data files), only the **com** file contains any solid solutions. All of these are treated with ideal site-mixing models (the exception being olivine, which is treated according to a binary regular solution model). The actual types of models used on the data file are defined by the **jsol** flag array. The corresponding parameters (site-mixing parameters, interaction coefficients, and parameters used to compute interaction coefficients) are stored on the data file in the **cpx** array. The elements of this array are represented below as $p_{k\psi}$. In EQ3NR and EQ6, solid solutions are presently ignored unless the option switch **iopt4** is set to a value greater than or equal to 1.

The various models presently treated in EQ3/6 are discussed in the following sections. To avoid confusion, we will write the activity, mole fraction, activity coefficient, and related parameters of a solid solution component with a "$\sigma\psi$" subscript in place of "*i*." Here σ denotes the component itself (takes the place of "*i*"), and ψ the solid solution (in order to be completely explicit about which solid solution is being addressed).

4.2. Ideal Solution, with One Optional Site-Mixing Parameter

The first activity coefficient model for solid solutions in EQ3/6 is for an ideal solution in either the molecular-mixing sense or a limited site-mixing model in which mixing is confined to one site and vacancies are ignored. The former is a special case of the latter. This model corresponds to **jsol** = 1 and is characterized by the equation (Wood and Fraser, 1977; Viani and Bruton, 1992):

$$a_{\sigma\psi} = x_{\sigma\psi}^{N_\psi} \tag{184}$$

where N_ψ is the site mixing parameter. This formulation is equivalent to:

$$log\lambda_{\sigma\psi} = (N_\psi - 1)\, log x_{\sigma\psi} \tag{185}$$

If $N_\psi = 1$, the above model is mathematically equivalent to an ideal molecular-mixing model ($log\lambda_{\sigma\psi} = 0$).

The N_ψ parameter is stoichiometric in nature. In essence, it is the number of formula units of the site on which mixing occurs per formula unit of the solid solution framework. In principle, the formula for all the end-member components of a solid solution can be written so as to yield $N_\psi = 1$, hence $log\lambda_{\sigma\psi} = 0$.

In the case of heterovalent single-site solid solutions such as clays and zeolites, vacancies are involved. In order to simplify the solution model and preserve the simple relationship defined by eqs (184) and (185), Viani and Bruton (1992) have chosen to treat such solid solutions according to a model in which the mixing entities are ions or ion-vacancy complexes. Thus, two sodium ion entities might mix with a calcium ion-vacancy entity.

The N_ψ parameter is obtained from the parameters read from the data file according to:

$$N_\psi = p_{7\psi} \tag{186}$$

4.3. Third-Order Maclaurin Model for a Binary Solution

The third-order Maclaurin model for a binary solution corresponds to **jsol** = 2. It is taken from Helgeson et al. (1970). The activity coefficients of the two end-member components are given by:

$$log\lambda_{1\psi} = \frac{1}{2.303RT}\left[-(\frac{W_{2\psi}}{2})x_2^2-(\frac{W_{3\psi}}{3})x_2^3\right] \tag{187}$$

$$log\lambda_{2\psi} = \frac{1}{2.303RT}\left[-(\frac{W_{2\psi}+W_{3\psi}}{2})x_1^2+(\frac{W_{3\psi}}{3})x_1^3+(W_{1\psi}+\frac{W_{2\psi}}{2}+\frac{W_{3\psi}}{6})\right] \tag{188}$$

Here $W_{1\psi}$, $W_{2\psi}$, and $W_{3\psi}$ are interaction coefficients. There are no site-mixing parameters.

The formulation represented by eqs (187) and (188) is highly unsymmetrical. In order to satisfy the condition that $log\lambda_{2\psi} \rightarrow 0$ as $x_1 \rightarrow 0$, the interaction coefficients are required to satisfy the relation:

$$W_{1\psi} = -\frac{W_{2\psi}}{2}-\frac{W_{3\psi}}{6} \tag{189}$$

The interaction coefficients are obtained from the parameters read from the data file according to:

$$W_{1\psi} = p_{1\psi} \tag{190}$$

$$W_{2\psi} = p_{2\psi} \tag{191}$$

$$W_{3\psi} = p_{3\psi} \tag{192}$$

However, $W_{1\psi}$ is actually recalculated using eq (189).

4.4. Regular Solution Model for a Binary Solution

The regular solution model for a binary solution corresponds to **jsol** = 3. It is also called a parabolic Maclaurin model. For a discussion of this model, see Saxena (1973, p. 11-12). The activity coefficients of the two end-member components are given by:

$$log\lambda_{1\psi} = \frac{1}{2.303RT}W_\psi x_2^2 \tag{193}$$

$$log\lambda_{2\psi} = \frac{1}{2.303RT}W_\psi x_1^2 \tag{194}$$

Here W_ψ is the single interaction coefficient. There are no site-mixing parameters. This formulation is symmetrical.

The interaction coefficients are obtained from the parameters read from the data file according to:

$$W_\psi = p_{1\psi} + p_{2\psi}T + p_{3\psi}P \tag{195}$$

Thus, the interaction coefficient in this model can be treated as a function of temperature and pressure. On the **com** data file in the R10 and R16 sets, there is a regular solution model for the solid solution olivine. The $p_{2\psi}$ and $p_{3\psi}$ parameters are set to zero, so the interaction coefficient is actually treated as a constant. A non-unit site-mixing parameter is also given in the $p_{7\psi}$ parameter, but this is not used.

4.5. Cubic Maclaurin Model for a Binary Solution

The cubic Maclaurin model for a binary solution corresponds to **jsol** = 4. For a discussion of this model, see Saxena (1973, p. 16). The activity coefficients of the two end-member components are given by:

$$log\lambda_{1\psi} = \frac{1}{2.303RT}\left[(2W_{2\psi} - W_{1\psi})x_2^2 + 2(W_{1\psi} - W_{2\psi})x_2^3\right] \tag{196}$$

$$log\lambda_{2\psi} = \frac{1}{2.303RT}\left[(2W_{1\psi} - W_{2\psi})x_1^2 + 2(W_{2\psi} - W_{1\psi})x_1^3\right] \tag{197}$$

Here $W_{1\psi}$ and $W_{2\psi}$ are interaction coefficients. There are no site-mixing parameters. This formulation is asymmetrical.

The interaction coefficients are obtained from the parameters read from the data file according to:

$$W_{1\psi} = p_{1\psi} + p_{2\psi}T + p_{3\psi}P \tag{198}$$

$$W_{2\psi} = p_{4\psi} + p_{5\psi}T + p_{6\psi}P \tag{199}$$

4.6. Guggenheim Polynomial Model for a Binary Solution

The Guggenheim polynomial model for a binary solution corresponds to **jsol** = 5. For a discussion of this model, see Saxena (1973, p. 14-15). The activity coefficients of the two end-member components are given by:

$$log\lambda_{1\psi} = \frac{1}{2.303RT}\left[(W_{1\psi} + 3W_{2\psi} + 5W_{3\psi})x_2^2 + (-4W_{2\psi} - 16W_{3\psi})x_2^3 + 12W_{3\psi}x_2^4\right] \tag{200}$$

$$log\lambda_{2\psi} = \frac{1}{2.303RT}\left[(W_{1\psi} - 3W_{2\psi} + 5W_{3\psi})x_1^2 + (4W_{2\psi} - 16W_{3\psi})x_1^3 + 12W_{3\psi}x_1^4\right] \tag{201}$$

Here $W_{1\psi}$, $W_{2\psi}$, and $W_{3\psi}$ are interaction coefficients. There are no site-mixing parameters. This formulation is asymmetrical.

The interaction coefficients are obtained from the parameters read from the data file according to:

$$W_{1\psi} = p_{1\psi} + p_{2\psi}T + p_{3\psi}T^2 \tag{202}$$

$$W_{2\psi} = p_{4\psi} + p_{5\psi}T + p_{6\psi}T^2 \tag{203}$$

$$W_{3\psi} = p_{7\psi} + p_{8\psi}T + p_{9\psi}T^2 \tag{204}$$

The full form of this model can be used in the present version of EQ3/6, although the parameters $p_{k\psi}$ for $k \geq 7$ are now intended to be reserved for site-mixing parameters.

4.7. Regular Solution Model for a Ternary Solution

The regular solution model for a ternary solution corresponds to **jsol** = 6. For a discussion of this model, see Prigogine and Defay (1954, p. 257). The activity coefficients of the three end-member components are given by:

$$log\lambda_{1\psi} = \frac{1}{2.303RT}\left[W_{12\psi}x_2^2 + W_{13\psi}x_3^2 + (W_{12\psi} - W_{23\psi} + W_{13\psi})x_2x_3\right] \tag{205}$$

$$log\lambda_{2\psi} = \frac{1}{2.303RT}\left[W_{12\psi}x_1^2 + W_{23\psi}x_3^2 + (W_{12\psi} - W_{13\psi} + W_{23\psi})x_1x_3\right] \tag{206}$$

$$log\lambda_{3\psi} = \frac{1}{2.303RT}\left[W_{13\psi}x_1^2 + W_{23\psi}x_2^2 + (W_{13\psi} - W_{12\psi} + W_{23\psi})x_1x_2\right] \tag{207}$$

Here $W_{1\psi}$, $W_{2\psi}$, and $W_{3\psi}$ are interaction coefficients. There are no site-mixing parameters. This formulation is symmetrical.

The interaction coefficients are obtained from the parameters read from the data file according to:

$$W_{12\psi} = p_{1\psi} \tag{208}$$

$$W_{13\psi} = p_{2\psi} \tag{209}$$

$$W_{23\psi} = p_{3\psi} \tag{210}$$

5. Basis Species: Key Concepts

5.1. Basis Species

In the EQ3/6 system, there is a set of master or *basis* species. In Chapter 2, this concept was introduced via the notion that one such species is associated with each chemical element and its associated mass balance (e.g., Na^+ for Na). If oxidation-reduction is considered, one additional species such as $O_{2(g)}$ for e^- must be added, which is associated with charge balance. Every remaining species (aqueous, mineral, or gas) is formally associated with a reaction which destroys it. For example, an aqueous complex is paired with its dissociation reaction, and a mineral with its dissolution reaction. The basis species are used as a set of generalized "building blocks" in writing chemical reactions. The reactions are then written in terms of only the single associated species and the set of basis aqueous species.

We will call a basis set as defined above a *strict basis* set. It is the minimal basis set required for chemical modeling. The number of species in this set, in the general case including a redox species, is given by:

$$s_B = \varepsilon_T + 1 \tag{211}$$

where ε_T is the number of chemical elements in the system of interest.The redox species itself will be denoted as the s_B-th species. In the case of systems in which there is no oxidation-reduction, we will simply treat the redox species as being inactive.

A speciation-solubility problem concerning an aqueous solution deals only with mass balances involving species in one (aqueous) phase. Therefore, the basis set in EQ3NR consists entirely of aqueous species. These are defined (at least initially) on the supporting data file. One is $H_2O_{(l)}$, the solvent. The redox species used in EQ3/6 is $O_{2(g)}$, which is treated in this context as a fictive aqueous species; the conventional e^- used by some other modeling codes is another example of such. The other basis species are simple species likely to dominate their respective mass balance relationships, at least in many instances.

Basis species are usually chosen as mono-elemental species such as Na^+ and Ca^{2+}. Some are also comprised of oxygen and/or hydrogen (e.g., SO_4^{2-} and $B(OH)_{3(aq)}$). No basis species on a supporting data file is permitted to be comprised of more than one chemical element other than oxygen or hydrogen. The purpose of this restriction is to avoid certain problems that would otherwise arise in defining the total concentrations of the basis species. Such problems do not arise in the case of dealing with elemental oxygen and hydrogen because no meaningful analytical values exist for the total concentrations of the associated basis species, $H_2O_{(l)}$ and H^+or of these elements themselves. The concentration of water as measured by its mole fraction is implicitly fixed by the concentrations of the solute components. The concentration of the hydrogen ion is analytically determined via the pH or some other approach not involving a total concentration.

Using a strict basis set, all mass balance relationships can be defined in terms of chemical elements and the coefficients describing the elemental compositions of all species. The charge bal-

ance relationship can be defined in terms of the electrical charges of the species. All non-basis species appearing in these balance equations are related to the basis species via the associated chemical reactions. The concentrations of these non-basis species are then determined by the concentrations of the basis species through the associated mass action equations, assuming that the activity coefficients appearing in these equations are known. Thus, if the concentrations of the basis species are known, they may be used to span (compute) the complete speciation of the system. In mathematics, a set with such properties is usually called a basis, which is actually where the term *basis set* in the present context is derived.

In thermodynamic modeling, one deals in an algebraic sense with n equations in n unknowns. The use of a basis set which is strict requires assuming that the concentration of every non-basis species appearing in a balance equation satisfies a corresponding mass action equation. This has the effect of requiring the modeled system to be in a complete state of internal chemical equilibrium. There is simply no mechanism in this construction to deal with even one simple reaction in a state of disequilibrium. The concept of internal equilibrium as used here refers to a system which excludes any non-basis species that do not appear in the balance equations. Thus, an aqueous solution may be in a state of internal equilibrium, but still supersaturated with respect to calcite. The mineral in this context is a non-basis species, but it does not appear in the balance equations which describe the aqueous solution. The system consisting of the same aqueous solution plus the mineral, however, is not in a state of internal equilibrium.

In EQ3NR, the modeled system consists exclusively of the aqueous solution. Systems including other phases are treated in EQ6. Nevertheless, it is apparent from the above example that in order to model systems with some internal disequilibrium, one must expand the basis set beyond the confines of the strict basis. In the example given above, we would make the mineral a basis species. However, since the associated reaction is presumed to be in disequilibrium, the associated mass action equation is not used as a governing constraint. In order to maintain a balance of n equations in n unknowns, it is necessary to introduce a mass new balance equation for the new basis species. In the present example, this is just a statement of how much of the mineral is present in the system. Note that this is a new kind of mass balance relation not related to a chemical element.

The same principle holds in modeling an aqueous solution. One might wish to compute a model in which Fe^{3+} is not in equilibrium with Fe^{2+}. If Fe^{2+} is already in the strict basis set, one must add Fe^{3+} to the basis set (or vice versa). In this case, the situation is more complicated, as the new basis species may have its own ion pairs and complexes appearing in the associated mass balance. This is a simple concept. However, it requires rethinking the description of mass balances, as the number of mass balance equations now exceeds the number of chemical elements. As we will show, a better concept is to associate the mass balance relations with corresponding basis species, not with the chemical elements.

We now show how to develop this more generalized concept for defining mass balance relations. Consider the following reaction:

$$HgCl_3^- = Hg^{2+} + 3Cl^- \tag{212}$$

This is represented in the software by paired arrays of reaction coefficients (floating point numbers) and names of the corresponding species (character variables). If you ask the question, how many chlorides is the non-basis species on the left hand side equivalent to, a human being will invariably answer the question by looking at the subscript "3" in the chemical formula of the species. In the software, this is equivalent to looking up the elemental composition of the species in the appropriate data array. Thus, this mechanism produces the required coefficient for evaluating the contribution of this species to a mass balance relation based on a chemical element. This is really the answer to the question, to how many chlorines (not chlorides) is this species equivalent. This is not what is presently desired, and a different approach is required.

The original question can be more accurately answered by looking at the reaction coefficients. Since reactions must satisfy mass and charge balance to be valid, the required information must be available there. A human being would probably answer the question by looking at the coefficient on the right hand side of the reaction. However, the coefficient of the non-basis species on the left hand side must also be considered to obtain the correct answer in the general case. To emphasize this, we note that the reaction can also be written as:

$$2HgCl_3^- = 2Hg^{2+} + 6Cl^- \tag{213}$$

Although a reaction whose coefficients have not been reduced to the lowest common denominator is unlikely to be written in any of the EQ3/6 data files, it is not prohibited. Also, some reactions written with a unit coefficient for the associated non-basis species require fractional coefficients. An example is:

$$Fe^{3+} + \frac{1}{2}H_2O_{(l)} = Fe^{2+} + H^+ + \frac{1}{4}O_{2(g)} \tag{214}$$

One might reasonably wish to avoid the fractions and write instead:

$$4Fe^{3+} + 2H_2O_{(l)} = 4Fe^{2+} + 4H^+ + O_{2(g)} \tag{215}$$

Furthermore, certain actions taken by the code as it executes, such as basis switching, may cause a reaction to be rewritten, and there is no general restriction requiring the new reaction to have a unit coefficient for the associated non-basis species.

In the software, the coefficients of products are defined as positive numbers and those of reactants as negative ones. For the basis species, these coefficients are symbolized by $b_{s'r}$, where s' denotes a basis species and r the reaction. The non-basis species associated with the r-th aqueous reaction is denoted by s'', and its reaction coefficient is symbolized by $b_{s''r}$. Thus, the factor giving the stoichiometric equivalence of such a species to the s'-th basis species is given by:

$$u_{s''s'} = -\frac{b_{s'r}}{b_{s''r}} \tag{216}$$

In a speciation-solubility problem, the mass balance equation for the s'-th basis species is then:

$$m_{T,s'} = m_{s'} + \sum_{r=1}^{r_T} u_{s''s'} m_{s''} \tag{217}$$

where r is the reaction associated with the s''-th species ($r = s'' - s_B$; see Section 9.2) and r_T is the number of reactions for the dissociation of non-basis aqueous species. Considerable care must be used in the application of such a formulation. Mathematically, it is quite rigorous. Physically, however, there are some potential problems. The quantity on the left hand side may or may not correspond to something that can be obtained by chemical analysis and therefore have physical as well as mathematical meaning. The formulation can be applied to any basis species. In the case of $O_{2(g)}$ or e^-, these have no physical meaning, as these are only hypothetical aqueous species in the first place. In the case of H^+, the total concentration has no physical significance. Its value is uniquely established only because one normally chooses to put this species in the basis set instead of OH^-. In the case of $H_2O_{(l)}$, the computed total concentration is also technically non-physical and depends on which of H^+ or OH^- is chosen as a basis species.

It was pointed out earlier that basis species on the EQ3/6 data files are restricted in composition to no more than one chemical element other that oxygen and hydrogen. This is done to protect the physical meaning of the total concentrations of basis species other than $H_2O_{(l)}$, H^+, and $O_{2(g)}$, for which there is no possibility of physical meaning, anyway. To illustrate the problem, consider the following three reactions in which $HgCl_3^-$ is used as a basis species in place of Hg^{2+}:

$$Hg^{2+} + 3Cl^- = HgCl_3^- \tag{218}$$

$$HgBr_3^- + 3Cl^- = HgCl_3^- + 3Br^- \tag{219}$$

Consider the mass balance of chloride and the contribution to it from Hg^{2+}. Applying eq (216) to the first reaction above gives a stoichiometric coefficie. t of -3. The same result is obtained for the bromide complex in the second reaction. The chloride complex itself has a stoichiometric coefficient of zero.

In EQ3NR and EQ6, the chloride complex in the above example is likely to strongly dominate the mass balance of dissolved mercury, giving an incentive to consider switching it into the basis in place of the mercuric ion. The codes deal with this situation by continuing to define the stoichiometric factors appearing in the mass balance relations in terms the reactions as they were written prior to basis switching, modified only for stoichiometric factors relating the new basis species to the old ones.

5.2. Organization and Treatment of Basis Species

The set of basis species on an EQ3/6 data file is divided into two parts: the strict basis and the *auxiliary* basis. The species in the strict basis set correspond one-to-one with the chemical elements, except for $O_{2(g)}$, which is used as a hypothetical aqueous species, and which corresponds to charge balance. These species appear first in the overall list of aqueous species. The solvent,

$H_2O_{(l)}$, is the first aqueous species.The hypothetical aqueous species $O_{2(g)}$ is the last. The species in the strict basis set are not associated with any reactions, as are all other species.

The auxiliary basis species follow the strict basis species. For the most part, they represent chemical elements in different oxidation states. However, they may also include any species which do not readily equilibrate with other basis species according to the associated reactions. Auxiliary basis species are used like strict basis species as "building blocks" in writing reactions for various species on the data file. In EQ3NR, an auxiliary basis species may be treated as either a basis species or a non-basis species. The choice is up to the user in each case. By default, an auxiliary basis species is eliminated from the active basis set (except for $O_{2(aq)}$ and $H_{2(aq)}$, which are special cases). Any reactions for other species written in terms of this species will be rewritten to reflect this. However, an auxiliary basis species is treated as an active basis species if the user provides an appropriate matching input on the **input** file, such as a total concentration. The non-basis aqueous species follow the auxiliary basis species. In the present version of EQ3/6, a species defined as a non-basis species on the data file can not be treated as an active basis species unless it is switched with an existing member of this set. This prevents defining an additional mass balance relation for this species. If it is desired to use such a species in the active basis set for the purpose of defining an additional such relation, it is necessary to modify the data file, moving the species into the auxiliary basis set.

An input model constraint, such as a total concentration, is required for each master aqueous species in order to perform a speciation-solubility calculation. However, as discussed in Chapter 6, there are limitations on the constraints that can be placed on a given basis species, depending on whether it is in the strict basis or the auxiliary basis. The user needs to be keenly aware of which species are in which set. Users should consult either the relevant **data0** data file or the **slist** (species list) file written by EQPT when it processes this file.

The user may specify selected examples of basis-switching on the **input** file. This provides a means of changing the set of basis species at run time. For example, a strict basis species may be exchanged with an auxiliary basis species. This puts the latter in the strict basis, the former in the auxiliary set. A basis species may also be switched with a non-basis species. A basis switch causes reactions to be re-written in terms of the new basis set. Judicious basis switching can improve the code numerics, sometimes making the difference in whether the iteration converges or not. As a general rule, a basis species should not make up an insignificant part of the associated mass balance when a total concentration is used as the input constraint. The compositional restrictions on basis species on the EQ3/6 data file do not apply to basis switching made when running EQ3NR or EQ6.

A basis switch involving a non-basis species causes the corresponding input constraints (such as those represented by the **csp** and **uphase** parameters; see Chapter 6), to be reassigned to the species brought into the basis set. For example, if $Al(OH)_4^-$ is to be switched into the basis set in place of Al^{3+}, for which **jflag** = 16 (log activity constant) and **csp** = -5.0 (the desired value), the model actually specified is one in which the log activity of $Al(OH)_4^-$ is -5.0. In this case, specifying a basis switch actually changes the definition of the problem. However, in a switch involving a basis species constrained to satisfy a specified total concentration, the total concentration is

recomputed by a stoichiometric adjustment to match the species moved into the basis. In this case, the problem itself is not really changed; it is merely expressed in different terms.

Setting the **jflag** = 30 for an auxiliary basis species causes the species to be eliminated from the active basis set. As noted above, this is the default condition for most such species. Elimination from the active basis set causes reactions originally written in terms of the eliminated species to be rewritten. For example, consider the following reaction:

$$FeSO_4^+ = Fe^{3+} + SO_4^{2-} \tag{220}$$

Elimination of Fe^{3+} from the basis set causes this to be rewritten as:

$$FeSO_4^+ + \frac{1}{2}H_2O_{(l)} = Fe^{2+} + H^+ + \frac{1}{4}O_{2(g)} + SO_4^{2-} \tag{221}$$

Thus, $FeSO_4^+$ now appears to be, and is treated as, a complex of Fe^{2+}. Elimination thus has the effect of combining mass balances. In this case, Fe^{3+} and its complexes are folded into the mass balance for Fe^{2+}. Note that the reaction for $FeSO_4^+$ on the data file must be the first, not the second, of the two above reactions. Otherwise, $FeSO_4^+$ would have been considered a complex of Fe^{2+} from the start. If one did not eliminate Fe^{3+} from the active basis set, $FeSO_4^+$ would be incorrectly folded in to the mass balance for Fe^{2+}, and missing from that of Fe^{3+}.

In the majority of cases, there are only one or two auxiliary basis species for a given chemical element. When there is more than one such auxiliary basis species, the species can be "chained." The first such species must be related through its associated reaction to the corresponding strict basis species. This can also be done in the case of the second, third, etc., such basis species. However, the second such auxiliary basis species could be directly related to the first such auxiliary basis species. The third could be directly related to the first or second, etc. Of course, a non-basis species can be directly related to any basis species.

Chaining is not significant in the case of most chemical elements. Carbon, however, is an exception. A large number of organic species are now present on the **com** and **sup** data files because of the inclusion of such in SUPCRT92 from the work of Schock and Helgeson (1990). Several of these are treated as auxiliary basis species, the majority as non-basis species of which all are directly related to one of the organic species in the auxiliary basis. The basic problem with organics in a geochemical modeling code is that they may often be treated as complexes of HCO_3^- when this is not what is desired. In the "R7" versions of the **com** and **sup** data files, each of the several organic species in the auxiliary basis set is directly related to HCO_3^-. In order to model organic-free systems without the unexpected appearance of organics in the model, it is necessary to enter on the **input** file a zero concentration for each organic species in the auxiliary basis set. In future versions of the data files, one such species (probably 'acetic acid(aq)') may be set up as a sort of master organic species. Only this organic species will be directly related to HCO_3^-. All other organic species in the auxiliary basis set will be directly related to this master organic spe-

cies. Then all organics can be kept out of a computed model by entering on the **input** file a zero concentration for just the master organic species.

5.3. Stoichiometric Conversions of Analytical Data

The analytical data used to define speciation-solubility problems in EQ3NR pertain to the basis species on the supporting data file employed in a given run. To use the code correctly, one must know what the species are, and it is often necessary to correct the analytical data one is provided in order to provide a stoichiometric match.

For example, the river water test case of Nordstrom et al. (1979) includes the following data:

- *Si* 8.52 mg/L.
- *B* 0.050 mg/L.
- PO_4^{3-} 0.210 mg/L.

The species on the **data0.com** data file which respectively correspond to these components are $SiO_{2(aq)}$, $B(OH)_{3(aq)}$, and HPO_4^{2-}. The problem is that, for example, 8.52 mg/L of *Si* is not equivalent to 8.52 mg/L of $SiO_{2(aq)}$. A stoichiometric conversion, of the sort common in analytical chemistry, must be done. This is illustrated in the present case by:

$$C_{mg/L,\,SiO_{2(aq)}} = \frac{M_{w,\,SiO_{2(aq)}}}{M_{w,\,Si}} C_{mg/L,\,Si} \tag{222}$$

where $M_{w,i}$ is the molecular weight of the component labeled *i*. The atomic weight of *Si* is 28.086 g/mole and that of *O* is 15.999g /mole. The molecular weight of $SiO_{2(aq)}$ is therefore 60.0840 g/mole. The ratio of the molecular weights is therefore 2.1393, and the 8.52 mg/L of *Si* is therefore equivalent to 18.23 mg/L of $SiO_{2(aq)}$. By following this method, it can be shown that the 0.050 mg/L of *B* is equivalent to 0.286 mg/L of $B(OH)_{3(aq)}$, and that the 0.210 mg/L of PO_4^{3-} is equivalent to 0.212 mg/l of HPO_4^{2-}. Corrections are analogous if the data are concentrations in mg/kg of solution.

The situation is much simpler if the analytical data are reported as molalities or molarities, as no conversion is generally necessary. For example, 0.0001 molal *Si* is equivalent to 0.0001 molal $SiO_{2(aq)}$.

The code user must make any necessary stoichiometric conversions before entering the data on the **input** file. EQ3NR contains no provisions for direct input of data corresponding to dissolved components other than the basis species appearing on the data file used, so it is not possible for it to make these conversions for the user.

6. The EQ3NR Input File: Setting up the Problem

6.1. Input File Characteristics and Contents

We examine in this chapter the EQ3NR **input** file. This file is the means by which the user specifies a problem to be solved by the code. The user must supply a compatible **data1** file, and the results obtained may differ if the problem is run with more than one such data file. Some examples of EQ3NR **input** files are presented in this chapter, and the reader will find more examples, along with the corresponding **output** files, in Chapter 7. Still more examples are given in Appendix F of the EQ6 Theoretical Manual and User's Guide (Wolery and Daveler, 1992).

The EQ3NR **input** file presently exists in one of two formats. The first is of the type used in previous versions of the code (e.g., Wolery, 1983). This is a compact form more suitable for use by experienced users. We will refer to this as the "W" format. A newer optional form (Daveler and Bourcier, unpublished) has a menu-style format that is much less compact, but which may be easier to use for less experienced users. We will refer to this as the "D" format. The present manual will focus more on the "W" format, as this naturally leads into a discussion of the relevant code variables. Some examples will be given of problems presented in both formats.

The recommended way of creating a new **input** file is to use an old one as a template. This works best if the old one is as close as possible to the desired new one. To this end, a representative selection of sample input files is included in the distribution package for EQ3/6 (Wolery and Daveler, 1992a). The distribution package also includes some software which converts input files in "W" format to "D" format. Using this template approach reduces the necessary knowledge a user must have concerning **input** file formats, and minimizes the need to consult the related documentation.

The "W" format **input** file is read by the code module **readx.f**. The source code for this module is extensively documented internally by of comment lines. On-line access to the source code of this module may helpful to users, serving as a kind of on-line documentation. The "D" format **input** file is read by the module **rdninp.f**, which calls a number of other modules in carrying out this function (the EQ3NR modules **rdtyp1.f**, **rdtyp2.f**, **rd3tds.f**, **rdtyp4.f**, **rdtyp5.f**, and **rdtyp9.f**, and the EQLIB modules **rdtyp0.f**, **rdtyp6.f**, **rdtyp7.f**, and **rdtyp8.f**). The source codes of these modules may also be helpful to users as on-line documentation.

An **input** file of either format contains a title field to provide space for internal documentation. In addition, an **input** file may contain remarks in comment lines. These are marked by an asterisk in column one and are analogous to comment lines in FORTRAN source code. They may appear anywhere in the **input** file.

Regardless of which format of **input** files is used, EQ3NR writes an "instant echo" of the **input** file on the **output** file. That is to say, after the code has read a line or closely related group of lines, it echoes their contents. This is particularly helpful in identifying the causes of read format errors, which most commonly occur when a line is missing or out of the proper sequence. This feature also provides a record of the **input** file used for a given run. However, it does not include comment lines. If necessary, a lost **input** file may be recovered (less comment lines) by extracting the echo from the **output** file.

The user is cautioned that the number of lines in an EQ3NR **input** file in either format is variable. Whether or not some potential lines appear in a given file deperds upon the contents of other lines. Some items, such as the constraints applied to the basis species, appear on successive lines until a terminator appears. Certain options, if invoked, require corresponding additional lines of input. The file structure has been chosen so that the size of the **input** file reflects the size of the problem.

There are no species index numbers for users to bother with. Internally, EQ3NR sets up its own internal indexing schemes at run time. Users deal with species in terms of their names, which are 24-byte character variables. These are much easier to remember and are instantly recognizable. However, the names must match perfectly with those on the data file used or EQ3NR will not recognize them. In such a case, the code will write an error message and terminate activity on the current problem input.

EQ3NR input problems may be stacked on the **input** file so that the code will read one problem, solve it, read another, solve it, and so on, in one job. In most cases, if an error is caught in one problem input, the code will proceed to the next problem input, if any.

A short summary of the contents of the EQ3NR **input** file in "W" format is given in on the following pages. Parameters for which default values are recommended are marked with an asterisk. To take the default for a given parameter, leave the corresponding input field blank. Following the short summary is a discussion of the **input** file parameters themselves. The user need not enter values for all of these. Some may not appear on specific examples of **input** files, depending on other parameters. This summary is followed by an example of an EQ3NR input file in both "W" and "D" formats (more examples are given in Chapter 7).

Short Summary of the EQ3NR **input** file ("W" format):

Parameters	Format
Do **n** from 1, ending with the string 'endit.' in column 1:	
utitl(n)	(a80)
End do	
tempc	(12x,e12.5)
rho, tdspkg, tdspl (enter only one of **tdspkg, tdspl**)	(3(12x,e12.5))
fep, uredox (enter either **fep** or **uredox**)	(12x,e12.5,12x,a24)
tolbt(*), toldl(*), tolsat(*)	(3(12x,e12.5))
itermx(*)	(12x,i2)
iopt1 - iopt10	(12x,10i5)

iopg1 - iopg10	(12x,10i5)
iopr1 - iopr10	(12x,10i5)
iopr11 - iopr20	(12x,10i5)
iodb1 - iodb10	(12x,10i5)
uebal	(12x,a24)
nxmod	(12x,i2)

If **nxmod** > 0:

Do **n** = 1, **nxmod**:

uxmod(j,n), j =1, 3)	(12x,i24),
jxmod(n), **kxmod(n)**, **xlkmod(n)**	(12x,i2,22x,i2,22x,e12.5)

End do

End if

Do until 'endit.' starting in column 1 of the **ubasis** line is encountered:

ubasis(n)	(26x,a24)
uspecb(n)	(24x,a24)
jflag(n), **csp(n)**	(10x,i2,8x,e12.5)

If **jflag(n)** $\geq$ 17 and **jflag(n)** $\leq$ 21:

uphas1(n), **uphas2(n)**	(10x,a24,11x,a24)

End if

End do

If **iopt4** = 2:

Do until 'endit.' starting in column 4 of the **usolb** line is encountered:

usolb(n)	(3x,a24)

Do until 'endit.' starting in column 7 of the **umemb** line is encountered:

(**umemb(i,n)**, **xbarb(i,n)**	(6x,a24,3x,f10.4)

End do

End do

End if

(*) Default values are recommended (i.e., leave these blank on the **input** file)

Summary of EQ3NR **input** file parameters:

utitl Up to 100 lines of text that describe the input problem, terminated by an 'endit.' beginning in column 1.

tempc Temperature, °C.

rho Aqueous solution density, g/ml. The default value is 1.0.

tdspkg Total dissolved salts, mg/kg of solution.

tdspl Total dissolved salts, mg/L (do not enter both of **tdspkg** and **tdspl**).

fep Redox parameter:

= *pe* if **iopt1** = -2.

= *Eh* if **iopt1** = -1.

= log oxygen fugacity if **iopt1** = 0.

uredox Name of an auxiliary master species; this defines a redox couple that specifies the oxygen fugacity (**iopt2** = 1; enter only one of **fep**, **uredox**).

tolbt Convergence tolerance on Newton-Raphson residual functions. The default value is 1×10^{-6}.

toldl Convergence tolerance on Newton-Raphson correction terms. The default value is 1×10^{-6}.

tolsat Saturation tolerance; this only determines whether or not a mineral is listed as "saturated" on the **output** file; it has no effect on Newton-Raphson iteration. This value is used to test the affinity, not the saturation index. The default value is 0.5 kcal; reasonable values range from 0.1 kcal to 1.0 kcal.

itermx Limit on the number of Newton-Raphson iterations. The default value is 30.

iopt1 Option switch for determining the redox parameter input:

-3 = This option instructs the code to look for an input constraint on $O_{2(g)}$ in a normal constraint block for a basis species (see **uspecb**, **jflag**, **csp**, **uphas1**, and **uphas2** below; in this case, **uspecb** ='o2(g)'). By choosing **jflag** = 19, 20, or 21, the oxygen fugacity can by fixed by a heterogenous reaction for a mineral, solid solution end-member component, or gas species, respectively (defined by the **uphas1** and **uphas2** inputs, see below).

-2 = The *pe* is specified in **fep**.

-1 = The *Eh* is specified in **fep**.

0 = The log oxygen fugacity is specified in **fep**.

1 = An aqueous redox couple identified by the **uredox** variable constrains the oxygen fugacity.

iopt2 Option switch for automatic basis switching:

0 = Turns it off.

1 = Turns it on.

iopt3 Option switch for writing a **pickup** file for input to EQ6:

-1 = No **pickup** file is generated.

0 = A **pickup** file is generated.

iopt4 Option switch for solid solutions:

0 = Solid solutions are ignored.

1 = Only solid solutions for which compositions are given on the **input** file are considered.

2 = Both input and hypothetical solid solutions are considered.

iopt5 Option switch, not currently used.

iopt6 Option switch, controls convergence testing:

-1 = Only residual functions are tested.

0 = Both residual functions and correction terms are tested.

iopt7 Option switch, determines **pickup** file format:

0 = Normal version 7.0 format.

1 = Post-7.0 version format (do not use).

iopt8 - iopt10
Option switches, not currently used.

iopg1 Activity coefficient option switch- choice of basic equations:

-1 = Use the Davies equation.

0 = Use the B-dot equation.

1 = Use Pitzer's equations.

Caution: a compatible **data1** file must be used.

iopg2 Activity coefficient option switch- choice of *pH* scale:

-1 = Evaluate single-ion expressions and use as is.

0 = Convert results from single-ion expressions to the NBS *pH* scale.

1 = Convert results from single-ion expressions to the scale on which $log\,\gamma_{H^+} = 0$.

iopr1 Print option switch for species read from the data file:

0 = Do nothing.

1 = List all the species after the data file has been read (this can be lengthy).

iopr2 Print option switch for reactions read from the data file:

0 = Do nothing.

1 = Print all reactions (this can be extremely lengthy).

2 = Also print the *log K* values.

3 = Also print the coefficients of the interpolating polynomials.

iopr3 Print option ordering switch for the aqueous species distribution table:

0 = Present in order of decreasing concentration.

1 = Present in the order in which the species appeared on the data file.

iopr4 Print option cutoff switch for the aqueous species distribution table:

0 = Print all aqueous species.

1 = Print only those with concentrations greater than 1×10^{-20} molal.

2 = Print only those with concentrations greater than 1×10^{-12} molal.

3 = Do not print the aqueous species distribution.

iopr5 Print option cutoff switch for the mass balance percentage tables:

-1 = Do not print.

0 = Print, cutting off at 99% of the total for each table.

1 = Print all species in each table.

iopr6 Print option switch for the mean ionic properties table:

0 = Do nothing.

1 = Print the table.

iopr7 Print option for mineral affinity/saturation index tables:

-1 = Do not print.

0 = Print, excluding minerals whose affinities are less than -10 kcal.

1 = Print data for all minerals.

iopr8 Print option for hard core diameters and hydration numbers of aqueous species:

-1 = Do not print.

0 = Print, these data.

iopr9 Print option switch for Pitzer interaction coefficients:

0 = Print only warnings, if any.

1 = Print the species in the model and the number of coefficients.

2 = Print the species and the associated pairs and triplets of species for which the coefficients are defined.

iopr10 Print option switch for the mean ionic properties table:

0 = Do nothing.

1 = Print the stoichiometric concentrations of the basis species.

iopr11 - iopr20

Print option switches, not currently used.

iodb1 Debugging print option switch for general informational messages:

0 = Do nothing.

1 = Print certain messages which may be of diagnostic value.

2 = Print a higher level of such messages.

iodb2 Debugging print option switch for pre-Newton-Raphson optimization:

0 = Do nothing.

1 = Print a summary of the optimization process.

2 = Print a more detailed summary.

iodb3 Option switch for the iteration variable kill option:

0 = Do nothing.

1 = Invoke the killer option (this is intended for the use of code developers only).

iodb4 Debugging print option switch for Newton-Raphson iteration:

0 = Print a summary of the process.

1 = Print a more detailed summary of the process.

iodb5 Debugging print option switch for stoichiometric equivalence factors:

0 = Do nothing.

1 = Print the equivalence factors for oxygen and hydrogen mass balances only.

2 = Print the equivalence factors for all mass balances.

iodb6 Debugging print option switch for calculation of equivalence factors:

0 = Do nothing.

1 = Print details of these calculations.

iodb7 Debugging print option switch for reactions:

0 = Do nothing.

1 = Print the reactions on a file called **rlist**, before and after any basis switching operations (intended for use only by code developers).

iodb8 - iodb10

Debugging print option switches, not currently used.

uebal Name of ionic species for electrical balancing; if **uebal** is **'pickl.'**, the code picks a species; if **uebal** is blank, no electrical balancing is done. In most instances, users should not have the code do electrical balancing.

nxmod Number of alter/suppress options (number of species to be suppressed or whose *log K* values are to be modified).

uxmod Name of a species to be suppressed or whose corresponding equilibrium constant is to be modified for use in the current run.

0 = pure mineral.

1 = solid solution.

2 = special reactant.

3 = aqueous species.

4 = gas.

kxmod The nxmod alter/suppress option (see **jxmod** and **uxmod**):

-1 = Suppress.

0 = Replace the *log K* by **xlkmod**.

1 = Augment the *log K* by **xlkmod** units.

2 = The *log K* changed as if the corresponding Gibbs energy of reaction were decremented by **xlkmod** kcal/mole.

xlkmod	Equilibrium constant alteration function (see above).
ubasis	The name of a run-time basis species (if not the same as **uspecb**, the name of the species to switch into the basis set in place of the species whose name is input in **uspecb**).
uspecb	The name of a data file basis species.
jflag	The option flag which defines the type of input constraint (see below).
csp	A floating point datum whose meaning is determined by the corresponding value of the **jflag** option flag; usually this is a concentration of some type.
uphas1	The name of an aqueous basis species or a mineral, solid solution, or gas species required to define an input constraint under the **jflag** = 17, 18, 19, 20, and 21 options.
uphas2	The name of solid solution end-member component required to define an input constraint under the **jflag** = 20 option.
usolb	The name of a solid solution for which a composition is entered for the purpose of computing the corresponding affinity and saturation index.
umemb	The name of an end-member component of the solid-solution whose name is input in **usolb**.
xbarb	The mole fraction of a solid solution end-member component whose name is input in **umemb**.

Table of **jflag** Options:

jflag	**csp**
-1	Suppression option- no csp input.; This has the same effect as entering a concentration of zero. This is a convenient way to keep unwanted auxiliary basis species from appearing in the model.
0	Total molality.
1	Total molarity.
2	Total concentration, mg/L.
3	Total concentration, mg/kg of solution.
4	Free molality.
5	Free molarity.
6	Free concentration, mg/L.
7	Free concentration, mg/kg of solution.

8	Free concentration, $cm^3(STP)/cm^3$ solution.
16	The log activity. This is the means of entering *pH* (*-pH* = $log a_{H^+}$).
17	Combination log activity function- enter the name of the corresponding ion as **uphas1**. This is the means of entering *pHCl* (*-pHCl* = $log a_{H^+} + log a_{Cl^-}$).
18	Mean log activity- enter name of corresponding ion as **uphas1**.
19	Equilibrium with a pure mineral- no **csp** input; enter name of mineral as **uphas1.**
20	Equilibrium with a solid solution end-member component- no **csp** input; enter name of the solid solution as **uphas1** and the name of the end member as **uphas2.**
21	Equilibrium with gas- enter the log fugacity of the gas as csp; enter the name of the gas as **uphas1.**
27	Equilibrium with other basis species, but unconstrained by any mass balance relations- no **csp** input; this option is available only for auxiliary basis species. This is the default for $O_{2(aq)}$ and $H_{2(aq)}$.
30	Eliminate an auxiliary basis species from the active basis set- no **csp** input. This is the default for all auxiliary basis species other than $O_{2(aq)}$ and $H_{2(aq)}$.

Example of an EQ3NR **input** file in "W" format.

```
EQ3NR input file name= swmaj.3i
Description= "Sea water, major ions only"
Version number= 3245   Stage number= 01
Created 06/08/90     Creator= T.J. Wolery
Revised 06/08/90     Revisor= T.J. Wolery

  Sea Water, including only the major ions. This is a considerably
pared-down version of swtst.3i, which contains the full benchmark
sea water test case of Nordstrom et al. (1979, Table III).

  Purpose: to test the code on a small problem involving a moderately
concentrated solution. The activity coefficients of the aqueous
species are calculated from the B-dot equation and related equations.

                        References

Nordstrom, D. K., et al., 1979, A comparison of computerized chemical
  models for equilibrium calculations in aqueous systems, in Jenne,
  E. A., editor, Chemical Modeling in Aqueous Systems, ACS Symposium
  Series, v. 93, American Chemical Society, Washington, D.C.,
  p. 857-892.

endit.
     tempc=          25.
       rho=     1.02336    tdspkg=          0.     tdspl=          0.
       fep=       0.500    uredox=
    tolbt=          0.     toldl=          0.     tolsat=         0.
   itermx=  0
*                1    2    3    4    5    6    7    8    9   10
  iopt1\10=     -1    0    0    0    0    0    0    0    0    0
  iopg1\10=      0    0    0    0    0    0    0    0    0    0
  ioprl\10=      0    0    0    0    0    0    0    0    0    0
 ioprl1\20=      0    0    0    0    0    0    0    0    0    0
  iodb1\10=      0    0    0    0    0    0    0    0    0    0
    uebal= none
    nxmod=  0
*
data file master species= na+
   switch with species=
   jflag=  3   csp= 10768.
data file master species= k+
   switch with species=
   jflag=  3   csp= 399.1
data file master species= ca++
   switch with species=
   jflag=  3   csp= 412.3
data file master species= mg++
   switch with species=
```

```
    jflag=  3    csp= 1291.8
data file master species= h+
    switch with species=
    jflag= 16    csp= -8.22
data file master species= hco3-
    switch with species=
    jflag=  0    csp= .002022
data file master species= cl-
    switch with species=
    jflag=  3    csp= 19353.
data file master species= so4--
    switch with species=
    jflag=  3    csp= 2712.
endit.
```

The EQ3NR **input** file in "D" format is illustrated by the following example, which contains the same problem that was just presented in "W" format. The mapping between the two is largely self-evident. The most immediately obvious characteristic of an **input** file in "D" format is its use of the "|" character to delimit fields in which data are entered. Note that an asterisk ("*") is used to mark the choice of units for entering total dissolved salts. It may also be used to choose among *two of the options for electrical balancing ("code selects" and "not performed").* If electrical balancing is to be done on a specified ion, the name of the ion should simply be entered in the field provided. The principal option switches are set in blocks in which an asterisk is used to mark the desired choice. If more than one choice is marked for the same option, the code writes an error message and execution stops. In the case of debugging option switches, the user enters a numerical input of 0, 1, or 2. If 2 is a valid input, a "2" appears on the line following the final "|". When EQ3NR reads an **input** file in "D" format, it immediately checks the range limits and notes discrepancies in the instant echo on the **output** file. When the code reads an **input** file in "W" format, such checks are made after the **input** file has been read.

Note that the **iopt1** option is handled in a special way. It does not appear in the options block, but in the block of inputs for the basis species. Here it appears as a "species" called 'REDOX'. This is used in the example to input the *Eh* by entering the corresponding string 'EH'. The available options are as follows:

iopt1	"D" format string	Meaning
-3	'REDOX COUPLE'	*A normal basis species input line for $O_{2(g)}$ is expected to immediately* follow the 'REDOX' input line. The "D" format string for this option is not descriptive of the actual option; something like 'O2 INPUT' would have been a better choice.
-2	'PE'	*pe*.
-1	'EH'	*Eh*, volts.
0	'LOGFO2'	log oxygen fugacity.
1	'REDOX COUPLE'	Use redox couple defined by **uredox** input. The auxiliary basis species corresponding to the **uredox** variable is defined by the species on the species input line *immediately* following the 'REDOX' input line. It is not defined in the constraint field on the 'REDOX' input line itself, as one might expect.

Although these strings are listed here in upper case, the code does not consider case in interpreting them. Similar strings noted below are treated in the same manner.

Instead of entering a **kxmod** number to define the type of an "**nxmod**" alter/suppress option, one uses one of the following strings:

kxmod	"D" format string	Meaning
-1	'SUPPRESS'	Suppress the phase/reaction.
0	'REPLACE'	Replace the *log K*.
1	'AUGMENTK'	Augment the *log K*.
2	'AUGMENTG'	The *log K* changed as if the corresponding Gibbs energy of reaction were decremented by **xlkmod** kcal/mole.

The **jflag** options are also handled in "D" format by character strings. These are listed in the following table.

jflag	"D" format string
0	'MOLALITY'
1	'MOLARITY'
2	'MG/L'
3	'MG/KG'
4	'FREE MOLAL'
5	'FREE MOLAR'
6	'FREE MG/L'
7	'FREE MG/KG'
8	'FREE CM3/CM' (*)
16	'LOG ACTIVITY (†)
17	'LOG ACTIVITY COMBO' (††)
18	'LOG MEAN ACTIVITY'
19	'MINERAL'
20	'SOLID SOLUTION'
21	'GAS'
27	'DEPENDENT'
30	'ELIMINATED'

(*) This is a typographical error in the code.

(†) One may use the string 'PH' to enter the *pH*.

(††) One may use the string 'PHCL' to enter the *pHCl*.

Example of the same EQ3NR **input** file in "D" format.

```
EQ3NR input file name= swmaj.3i
Description= "Sea water, major ions only"
Version number= 3245   Stage number= 01
Created 06/08/90    Creator= T.J. Wolery
Revised 06/08/90    Revisor= T.J. Wolery

  Sea water, including only the major ions. This is a considerably
pared-down version of swtst.3i, which contains the full benchmark
sea water test case of Nordstrom et al. (1979, Table III).

  Purpose: to test the code on a small problem involving a moderately
concentrated solution. The activity coefficients of the aqueous
species are calculated from the B-dot equation and related equations.

                          References

Nordstrom, D. K., et al., 1979, A comparison of computerized chemical
  models for equilibrium calculations in aqueous systems, in Jenne,
  E. A., editor, Chemical Modeling in Aqueous Systems, ACS Symposium
  Series, v. 93, American Chemical Society, Washington, D.C.,
  p. 857-892.

Temperature (C)          | 25.00        |Density(gm/cm3)|  1.02336
Total Dissolved Salts    |              | mg/kg | mg/l  |*not used
Electrical Balancing on  |              | code selects  |*not performed
SPECIES   | BASIS SWITCH/CONSTRAINT | CONCENTRATION| UNITS OR TYPE
redox     |                         | 0.5000       |eh
na+       |                         | 10768.       |mg/kg
k+        |                         | 399.10       |mg/kg
ca++      |                         | 412.30       |mg/kg
mg++      |                         | 1291.9       |mg/kg
h+        |                         | 8.2200       |ph
hco3-     |                         |0.20220E-02   |molality
cl-       |                         | 19353.       |mg/kg
so4--     |                         | 2712.0       |mg/kg
Input Solid Solutions
 none     |                         |              |
SUPPRESSED SPECIES   (suppress,replace,augmentk,augmentg)    value
 none                |             |
OPTIONS
 - SOLID SOLUTIONS -
   * ignore solid solutions
     process hypothetical solid solutions
     process input and hypothetical solid solutions
 - LOADING OF SPECIES INTO MEMORY -
   * does nothing
     lists species loaded into memory
 - ECHO DATABASE INFORMATION -
   * does nothing
     lists all reactions
     lists reactions and log K values
     lists reactions, log K values and polynomial coef.
 - LIST OF AQUEOUS SPECIES (ordering) -
   * in order of decreasing concentration
     in same order as input file
 - LIST OF AQUEOUS SPECIES (concentration limit) -
   * all species
     only species > 10**-20 molal
     only species > 10**-12 molal
     not printed
 - LIST OF AQUEOUS SPECIES (by element) -
   * print major species
     print all species
     don't print
 - MINERAL SATURATION STATES -
   * print if affinity > -10 kcals
     print all
     don't print
 - pH SCALE CONVENTION -
   * modified NBS
     internal
     rational
 - ACTIVITY COEFFICIENT OPTIONS -
   * use B-dot equation
```

```
    Davies' equation
    Pitzer's equations
- AUTO BASIS SWITCHING -
  * off
    on
- PITZER DATABASE INFORMATION -
  * print only warnings
    print species in model and number of Pitzer coefficients
    print species in model and names of Pitzer coefficients
- PICKUP FILE -
  * write pickup file
    don't write pickup file
- LIST MEAN IONIC PROPERTIES -
  * don't print
    print
- LIST AQUEOUS SPECIES, ION SIZES, AND HYDRATION NUMBERS -
  * print
    don't print
- CONVERGENCE CRITERIA -
  * test both residual functions and correction terms
    test only residual functions
-------------------------------------------------------------------
DEBUGGING SWITCHES (o-off, 1,2-on, default is off)
-------------------------------------------------------------------
0  generic debugging information                                  2
0  print details of pre-Newton-Raphson iteration                  2
0  print details of Newton-Raphson iteration
0  print details of stoichiometric factors                        2
0  print details of stoichiometric factors calculation
0  write reactions on RLIST
0  list stoichiometric concentrations of master species
0  request iteration variables to be killed
-------------------------------------------------------------------
DEVELOPMENT OPTIONS  (used for code development)
-------------------------------------------------------------------
 none
-------------------------------------------------------------------
TOLERANCES                    (desired values)       (defaults)
-------------------------------------------------------------------
      residual functions                            1.e-10
        correction terms                            1.e-10
        saturation state                            0.5
number of N-R iterations                            30
-------------------------------------------------------------------
```

EQ3NR **input** files in "D" format are treated differently from ones in "W" format in one important way. The input from files in "D" format are checked against allowed range limits, where such exist, as soon as the relevant lines of data are read. Error or warning messages may therefore appear right after these lines in the instant echo portion of the **output** file. Essentially the same checks are made for input in "W" format, but after the current problem has been read from the **input** file. Any relevant error or warning messages therefore appear after the instant echo on the **output** file.

6.2. Cautions

In the absence of analytical data, it is recommended that users take the default condition of **jflag** = 27 for the auxiliary basis species $O_{2(aq)}$ and $H_{2(aq)}$. The reason is that one (or the other) will then usually have a low but sufficiently high concentration to insure some poising of the oxygen fugacity. This has little significance if one is not passing the solution model on to EQ6. However, if one does, this is helpful in avoiding computational difficulties in that code which may arise in trying to treat very ill-poised systems. As an example, a user might be interested in the dissolution of a feldspar in CO_2-charged water. This problem appears to have no redox aspect, but EQ6 expects a problem to have a redox aspect unless the user invokes special options. If some $O_{2(aq)}$ is not present in the modeled solution and these special options are not invoked, the code will run with very small step sizes, trying to do the nearly impossible job of accurately calculating the oxygen fugacity when there is hardly anything present to define it. If on running EQ3NR one sets **iopt1** = 0 and sets **fep** to -0.700, the oxygen fugacity will at the atmospheric value and a concen-

tration of $O_{2(aq)}$ will be computed which is in equilibrium with this. This is enough to poise the system in the example cited, and EQ6 can then be run quite nicely without having to invoke the above noted special options.

If the user creates an aqueous system model in which data are input for an auxiliary basis species but not for the corresponding strict basis species, the results from the code calculation may not be what the user intended. Suppose an analyst reports 2 mg/L of dissolved *Fe* for some water. On all of the existing EQ3/6 data files, Fe^{2+} is the strict basis species and Fe^{3+} is in the auxiliary basis. Generally speaking, the quantity reported by the analyst really means total dissolved iron of either form. To input this correctly, the user must enter 2 mg/L for Fe^{2+}. Then letting **jflag** for Fe^{3+} default to 30, the mass balances for the two forms are combined and the calculation is done correctly. On the other hand, if the user inputs 2 mg/L for Fe^{3+}, the **jflag** value for Fe^{2+} will default to -1, an internal value equivalent to **jflag** = 0 and **csp** = 0. In other words, Fe^{2+} (and its associated ion pairs and complexes) will be absent from the model, and the input total concentration will be distributed only among Fe^{3+} and its ion pairs and complexes.

As a general rule, it is not wise to create EQ3NR models in which an auxiliary basis species is present in the absence of the corresponding strict basis species. The calculations may be valid. However, the resulting aqueous solution model can not be input to EQ6. In situations in which an auxiliary basis species is present in the actual absence of the corresponding strict basis species, the roles of these species should be reversed by a basis switch. The instructions for the switch and the desired **jflag** and **csp** values should be entered on the **input** file as input for the original strict basis species. As the **csp** input is initially interpreted in terms of this species, and will then be recalculated by the code for stoichiometric equivalence with the species switched into the strict basis set, it may be necessary to recalculate the analytical data for stoichiometric equivalence with the original basis species. The code will then invert this same calculation when it makes the basis switch.

7. Sample Problems: Inputs and Outputs

7.1. Introduction

This chapter presents the **input** and output files for several speciation-solubility modeling problems that are successfully executed by EQ3NR. Each example begins with a short discussion. The **input** file and **output** files are then presented. The reader is encouraged to compare the **input** file examples presented here with the **input** file description presented in Chapter 6. Note that each **output** file begins with an "instant" echo of the **input** file. The examples presented here were run on a Sun SPARCstation IPC using optimized code, using the "W" format input files and the "R10" **com** and **hmw** data files. More examples of EQ3NR **input** files can be found in Appendix F of the EQ6 Theoretical Manual and User's Guide (Wolery and Daveler, 1992). The examples presented here are relatively simple. Two **input** files addressing more complex problems are presented without external comments or computed results in Appendix E.

On the **output** files one will occasionally see "+999" and "-999". These are respectively treated in EQ3/6 as the logarithms of plus infinity and zero, respectively. In the context of their appearance in these files, they generally signify a condition of "no data.".

7.2. Sea Water Test Case, with Major Cations and Anions Only

The first sample problem is the sea water test case from Nordstrom et al. (1979), but in which only the major cations and anions are included. This simplified test case is a classic example in geochemistry (Garrels and Thompson, 1962; Garrels and Christ, 1965). The **input** file was presented in both "W" and "D" formats in the previous chapter, and will not be repeated here. We will here focus on the **output** file. The corresponding **pickup** file is presented in Chapter 8. The **data1** file used is the **com** file, and the activity coefficients are computed from the B-dot equation (**iopg1** = 0).

The **output** file for this problem is presented in its entirety. The output begins with the name of the code and the version identification ("`EQ3NR, version 3245.1090R124`"). This is followed by a copyright notice and a set of disclaimers. This in turn is followed by a time and data stamp ("`Run  09:31      3Dec91`").

The instant echo of the **input** file appears following the line, "`--- reading the input file ---`". The **input** file contents appear shifted one character to the right, to be consistent with the use of older carriage control conventions. If all goes well this is followed by a blank line and the message, "`--- the input file has been successfully read ---`".

The next action of the code is to read the **data1** file. Note that the code prints messages when it starts ("`--- reading the data1 file ---`") and finishes ("`--- the data1 file has been successfully read ---`") this action.

This is followed by "EQ3NR" in large block letters. The code name and version identification are given, along with the version identification for the EQLIB library. The title from the **input** file is printed, followed by the name and version identification of the supporting data file. Problem and data file statistics are then printed, followed by a list of the problem inputs, including any default values or truncated values taken by the code. The table appearing under the header

"`--- input constraints ---`" is part of this summary. It is followed by a table headed by "`--- inactive aqueous species ---`". The entries in this table, if any, correspond to species which are either lacking thermodynamic data or have been suppressed by user options on the **input** file.

This is followed by the table headed by "`--- modified input constraints ---`". This lists the model inputs as they have been modified by the program before Newton-Raphson iteration commences. The modified constraints may differ from the original ones in several ways. They include all conversions of concentration units to the molal scale, any defaults provided by the program, and the effects of any basis switching. This table also shows the status of any auxiliary basis species that pertain to the model but which were not listed on the **input** file. Users should make it a point to examine this table to ensure that the model they are getting is indeed the one they want.

At this point, the code has set up the problem and is ready to solve it. The code then sets up starting estimates and refines them somewhat according to a pre-Newton-Raphson optimization algorithm described in Chapter 9. If this is successful, the code writes, "`--- optimization ended within requested limits ---`". This step is not always successful, which leads to the appearance of a message to that effect. However, this does not mean that the code has or will fail to solve the problem. It only means that the optimization algorithm failed to satisfy a set of tolerances before handing the problem over to hybrid Newton-Raphson iteration.

Some data are then printed showing the progress of hybrid Newton-Raphson iteration. The user does not need to be concerned with these data as long as the iteration process succeeds. The printing of these data on the **output** file is primarily done to provide an obvious break between the problem setup and the problem results. If iteration is successful, a message is printed to that effect ("`Hybrid newton-raphson iteration converged in  4 steps.`").

The data that follow should all be self-explanatory. Note that in the table headed by "`----- distribution of aqueous species -----`" many organic species appear. This occurs because the code is treating these species as complexes of HCO_3^-. This problem was discussed in Chapter 5 (at the end of Section 5.2). The R10 versions of the **com** and **sup** data files have a small number of organic species are in the auxiliary basis. There is a larger number of non-basis organic species, which are related by their associated reactions to these auxiliary basis species. These in turn are each related to HCO_3^-. Looking in the table headed by "`--- modified input constraints ---`", we see what the species are and that their **jflag** values have been assigned default values of 30. This eliminates them from the active basis set and causes them and all related non-basis organic species to be treated as complexes of HCO_3^-.

Knowing what these auxiliary basis species are, one could go back and enter **jflag** = 0 and **csp** = 0 for each of them, and then no organic species would appear in the computed model, which is really what one might normally desire. However, in this case, the results are not sensibly different from those that were desired anyway. Because the oxygen fugacity is high, the computed concentrations of all of these organic species are vanishingly small from a practical point of view.

Organic species can be more easily kept out of a speciation model using the newer R16 data files. The species 'acetic acid(aq)', one of the organics in the auxiliary basis, is now treated as the "mother" of all other organics. This was accomplished by writing the reactions of the other organic auxiliary basis species so that these species are converted to acetic acid instead of bicarbonate. Thus, entering **jflag** = 0 and **csp** = 0 for acetic acid suffices to prevent the appearance of any organic species (caution: the species $CH_{4(aq)}$ is not treated as an organic; enter **jflag** = 0 and **csp** = 0 for it as well if this species is not desired). Similarly, on the R16 data files, the species S_2^{2-} is the "mother" of other polysulfide species such as S_3^{2-} and S_4^{2-}, and $S_2O_3^{2-}$ is the "mother" of similar partially oxidized sulfur species such as $S_2O_4^{2-}$ and $S_2O_5^{2-}$. The R16 data files became available close to the publication deadline for this series of reports. It was not feasible to update the examples using these data files.

In the "----- summary of aqueous redox reactions -----", we see that the tabulated redox parameters have the same values for all of the redox couples in the table. This indicates that all these couples are in mutual equilibrium. This is not a scientific conclusion about the chemistry of sea water. Rather, it is an example that what comes out of a model must be consistent with what went into it. The input constraints for this problem assumed this equilibrium.

In the "----- summary of stoichiometric mineral saturation states -----", note that low to moderate supersaturations are predicted for several carbonate minerals, including calcite and aragonite. These are to be expected for surface sea water at 25°C (see for example, Berner, 1971).

Note that when the code is done with this problem, it looks for another on the **input** file. Not finding any, it terminates by writing run time statistics.

The EQ3NR **output** file for the sea water test case:

```
EQ3NR, version 3245R124

Copyright (c) 1987, 1990 The Regents of the University of California,
Lawrence Livermore National Laboratory.  All rights reserved.

Supported by EQLIB, version 3245R153

Copyright (c) 1987, 1990 The Regents of the University of California,
Lawrence Livermore National Laboratory.  All rights reserved.

This work was produced at the University of California,
Lawrence Livermore National Laboratory (UC LLNL) under
contract no. W-7405-ENG-48 between the U. S. Department of
Energy (DOE) and The Regents of the University of California
(University) for the operation of  LLNL.  Copyright is
reserved to the University for purposes of controlled
dissemination, commercialization through formal licensing,
or other disposition under terms of Contract 48; DOE
policies, regulations, and orders; and U. S. statutes.

                    DISCLAIMER

This computer code was prepared as an account of work
sponsored by an agency of the United States Government.
Neither the United States Government nor the University of
California nor any of their employees, makes any warranty,
express or implied, or assumes any liability or responsi-
bility for the accuracy, completeness, or usefulness of any
information, apparatus, product, or process disclosed, or
represents that its use would not infringe privately-owned
rights.  Reference herein to any specific commercial,
product, process, or service by trade name, trademark,
manufacturer, or otherwise, does not necessarily constitute
```

or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or the University of California, and shall not be used for advertising or product endorsement purposes.

LLNL YUCCA MOUNTAIN PROJECT DISCLAIMER

The Lawrence Livermore National Laboratory, a participant in the Yucca Mountain Project, has not determined that this software constitutes "approved code" for the conduct of "quality affecting work" for the Yucca Mountain Project.

```
Run  09:31      3Dec91


--- reading the input file ---
EQ3NR input file name= swmaj.3i
Description= "Sea water, major ions only"
Version number= 3245   Stage number= 01
Created 06/08/90     Creator= T.J. Wolery
Revised 06/08/90     Revisor= T.J. Wolery

  Sea water, including only the major ions. This is a considerably
pared-down version of swtst.3i, which contains the full benchmark
sea water test case of Nordstrom et al. (1979, Table III).

  Purpose: to test the code on a small problem involving a moderately
concentrated solution. The activity coefficients of the aqueous
species are calculated from the B-dot equation and related equations.

                          References

Nordstrom, D. K., et al., 1979, A comparison of computerized chemical
  models for equilibrium calculations in aqueous systems, in Jenne,
  E. A., editor, Chemical Modeling in Aqueous Systems, ACS Symposium
  Series, v. 93, American Chemical Society, Washington, D.C.,
  p. 857-892.

endit.
       tempc=  0.25000E+02
         rho=  0.10234E+01     tdspkg=  0.00000E+00      tdspl=  0.00000E+00
         fep=  0.50000E+00     uredox=
      tolbt=  0.00000E+00      toldl=  0.00000E+00     tolsat=  0.00000E+00
     itermx=  0
                  1     2     3     4     5     6     7     8     9    10
   iopt1-10=     -1     0     0     0     0     0     0     0     0     0
   iopg1-10=      0     0     0     0     0     0     0     0     0     0
   iopr1-10=      0     0     0     0     0     0     0     0     0     0
  iopr11-20=      0     0     0     0     0     0     0     0     0     0
   iodb1-10=      0     0     0     0     0     0     0     0     0     0
      uebal= none
      nxmod=  0
data file master species= na+
   switch with species=
   jflag=  3   csp=  0.10768E+05
data file master species= k+
   switch with species=
   jflag=  3   csp=  0.39910E+03
data file master species= ca++
   switch with species=
   jflag=  3   csp=  0.41230E+03
data file master species= mg++
   switch with species=
   jflag=  3   csp=  0.12918E+04
data file master species= h+
   switch with species=
   jflag= 16   csp= -0.82200E+01
data file master species= hco3-
   switch with species=
   jflag=  0   csp=  0.20220E-02
data file master species= cl-
   switch with species=
   jflag=  3   csp=  0.19353E+05
data file master species= so4--
   switch with species=
   jflag=  3   csp=  0.27120E+04
endit.

--- the input file has been successfully read ---

--- reading the datal file ---

--- the datal file has been successfully read ---

* note - (eqlib/inbdot) The following aqueous species
       have been assigned a default hard core diameter of
         4.000 Angstroms-
         cacl2(aq)
```

```
caco3(aq)
caso4(aq)
kcl(aq)
khso4(aq)
koh(aq)
mgco3(aq)
nach3coo(aq)
```

```
eeee   qqq   33333  n   n  rrrr
e   q   q      3   nn  n  r   r
eeee  q   q    33   n n n  rrrr
e     q q q     3   n  nn  r  r
eeee   qqq   3333   n   n  r   r
          q
```

```
eq3nr.3245R124x
 supported by eqlib.3245R153
```

```
EQ3NR input file name= swmaj.3i
Description= "Sea water, major ions only"
Version number= 3245   Stage number= 01
Created 06/08/90    Creator= T.J. Wolery
Revised 06/08/90    Revisor= T.J. Wolery

  Sea water, including only the major ions. This is a considerably
pared-down version of swtst.3i, which contains the full benchmark
sea water test case of Nordstrom et al. (1979, Table III).

  Purpose: to test the code on a small problem involving a moderately
concentrated solution. The activity coefficients of the aqueous
species are calculated from the B-dot equation and related equations.

                          References

Nordstrom, D. K., et al., 1979, A comparison of computerized chemical
  models for equilibrium calculations in aqueous systems, in Jenne,
  E. A., editor, Chemical Modeling in Aqueous Systems, ACS Symposium
  Series, v. 93, American Chemical Society, Washington, D.C.,
  p. 857-892.
```

```
data0.com.R10
THERMODYNAMIC DATABASE
generated by gembochs/INGRES  15-apr-91
--------------------------------------------------------------------

  the activity coefficients of aqueous solute species
and the activity of water are calculated according to the
b-dot equation plus others


temperature=  25.00 degrees celsius
pressure=    1.0132 bars


       78 elements are in the data base
      100 elements can be loaded into memory
        9 elements are active in this problem


      852 aqueous species are in the data base
      258 aqueous species were loaded into memory
      800 aqueous species can be loaded into memory
      133 aqueous species are active in this problem

      773 aqueous reactions are in the data base
      179 aqueous reactions were loaded into memory
      699 aqueous reactions can be loaded into memory


          886 minerals are in the data base
           84 minerals were loaded into memory
          850 minerals can be loaded into memory
           84 minerals are active in this problem

           12 solid solutions are in the data base
           50 solid solutions can be loaded into memory

           76 gases are in the data base
           16 gases were loaded into memory
           80 gases can be loaded into memory
           16 gases are active in this problem
```

```
iopt1  =  -1 (redox option switch)
iopt2  =   0 (automatic basis switching switch)
iopt3  =   0 (interfacing output control switch)
iopt4  =   0 (turn-on solid solutions switch)
iopt5  =   0 (not used)
iopt6  =   0 (conv. test criteria switch)
iopt7  =   0 (0/1 3245/post-3245 pickup file)
iopt8  =   0 (not used)
iopt9  =   0 (not used)
iopt10 =   0 (not used)


iopg1  =  0 (act. coeff. choice)
iopg2  =  0 (ph scale convention switch)
iopg3  =  0 (not used)
iopg4  =  0 (not used)
iopg5  =  0 (not used)
iopg6  =  0 (not used)
iopg7  =  0 (not used)
iopg8  =  0 (not used)
iopg9  =  0 (not used)
iopg10 =  0 (not used)


iopr1  =  0 (list loading of species)
iopr2  =  0 (list reactions and log k values)
iopr3  =  0 (aqueous species print order control)
iopr4  =  0 (aqueous species print cut-off control)
iopr5  =  0 (mass balance percentages print control)
iopr6  =  0 (mean ionic act coeff print control)
iopr7  =  0 (mineral affinity print control)
iopr8  =  0 (ion size and hydr. no. print control)
iopr9  =  0 (pitzer coefficients tabulation)
iopr10 =  0 (print concbs array)
iopr11 =  0 (not used)
iopr12 =  0 (not used)
iopr13 =  0 (not used)
iopr14 =  0 (not used)
iopr15 =  0 (not used)
iopr16 =  0 (not used)
iopr17 =  0 (not used)
iopr18 =  0 (not used)
iopr19 =  0 (not used)
iopr20 =  0 (not used)


iodb1  =  0 (print info. messages switch)
iodb2  =  0 (print pre-newton-raphson optimizations switch)
iodb3  =  0 (request iteration variables to kill)
iodb4  =  0 (print newton-raphson iterations switch)
iodb5  =  0 (list stoichiometric equivalences)
iodb6  =  0 (controls iodb5 level of detail)
iodb7  =  0 (write reactions on file rlist switch)
iodb8  =  0 (not used)
iodb9  =  0 (not used)
iodb10 =  0 (not used)


solution density =  1.02336 g/ml

total dissolved salts =       0.00 mg/kg solution
total dissolved salts =       0.00 mg/l


tolbt  =  0.10000E-05 (convergence tolerance on residual functions)
toldl  =  0.10000E-05 (convergence tolerance on correction terms)
tolsat =  0.50000E+00 (phase saturation tolerance, does not affect
                       convergence)


                    --- input constraints ---

  species                  csp        jflag   input type/co-species

na+                    1.07680E+04     3     tot conc, mg/kg
k+                     3.99100E+02     3     tot conc, mg/kg
ca++                   4.12300E+02     3     tot conc, mg/kg
mg++                   1.29180E+03     3     tot conc, mg/kg
h+                    -8.22000E+00    16     log activity
hco3-                  2.02200E-03     0     tot conc, molal
cl-                    1.93530E+04     3     tot conc, mg/kg
so4--                  2.71209E+03     3     tot conc, mg/kg


                  --- inactive aqueous species ---

(o-phth)--              benzene(aq)
```

--- modified input constraints ---

species	csp	jflag	input type/co-species
ca++	1.02874E-02	0	tot conc, molal
cl-	5.45882E-01	0	tot conc, molal
h+	-8.22000E+00	16	log activity
hco3-	2.02200E-03	0	tot conc, molal
k+	1.02076E-02	0	tot conc, molal
mg++	5.31496E-02	0	tot conc, molal
na+	4.68382E-01	0	tot conc, molal
so4--	2.82313E-02	0	tot conc, molal
(o-phth)--	0.00000E+00	30	eliminated species
acetic acid(aq)	0.00000E+00	30	eliminated species
acetone(aq)	0.00000E+00	30	eliminated species
benzene(aq)	0.00000E+00	30	eliminated species
clo4-	0.00000E+00	30	eliminated species
co2(aq)	0.00000E+00	30	eliminated species
co3--	0.00000E+00	30	eliminated species
ethane(aq)	0.00000E+00	30	eliminated species
h2(aq)	0.00000E+00	27	dependent species
hs-	0.00000E+00	30	eliminated species
methane(aq)	0.00000E+00	30	eliminated species
methanol(aq)	0.00000E+00	30	eliminated species
o2(aq)	0.00000E+00	27	dependent species
oh-	0.00000E+00	30	eliminated species
so3--	0.00000E+00	30	eliminated species

--- optimization ended within requested limits ---

iter= 0
del( )= 0.00000E+00, delfnc= 0.00000E+00
beta(conc so4--)= 7.30728E-02, betfnc= 0.00000E+00
bbig= 7.30728E-02, ubbig= so4--
bneg= 0.00000E+00, ubneg= none
bgamx= -1.52413E-04, ubgamx= mg4(oh)4++++
bsigmm= 0.00000E+00
bxi= 0.00000E+00
btfcnr= 0.00000E+00

iter= 1
del(conc so4--)= -2.71823E-02, delfnc= 0.00000E+00
beta(conc so4--)= 4.33286E-03, betfnc= 9.40705E-01
bbig= 4.33286E-03, ubbig= so4--
bneg= 0.00000E+00, ubneg= none
bgamx= 3.79456E-03, ubgamx= mg4(oh)4++++
bsigmm= -4.72140E-03
bxi= -7.47492E-03
btfcnr= 9.66315E-01

iter= 2
del(conc so4--)= -1.78199E-03, delfnc= 9.34443E-01
beta(conc so4--)= 8.88619E-05, betfnc= 9.79491E-01
bbig= 8.88619E-05, ubbig= so4--
bneg= 0.00000E+00, ubneg= none
bgamx= 1.61629E-04, ubgamx= mg4(oh)4++++
bsigmm= -1.18720E-04
bxi= -3.19161E-04
btfcnr= 9.97835E-01

iter= 3
del(conc so4--)= -3.60684E-05, delfnc= 9.79759E-01
beta(conc so4--)= 1.86766E-06, betfnc= 9.78982E-01
bbig= 1.86766E-06, ubbig= so4--
bneg= 0.00000E+00, ubneg= none
bgamx= 3.78998E-06, ubgamx= mg4(oh)4++++
bsigmm= -1.99320E-06
bxi= -7.48465E-06
btfcnr= 9.98734E-01

iter= 4
del(conc so4--)= -7.53517E-07, delfnc= 9.79109E-01
beta(conc so4--)= 4.07346E-08, betfnc= 9.78189E-01
bbig= 4.07346E-08, ubbig= so4--
bneg= 0.00000E+00, ubneg= none
bgamx= 8.28345E-08, ubgamx= mg4(oh)4++++
bsigmm= -4.06084E-08
bxi= -1.63586E-07
btfcnr= 9.98643E-01

Hybrid newton-raphson iteration converged in 4 steps.

----- Summary of the Aqueous Phase -----

----- Elemental composition of the aqueous phase -----

element	mg/l	mg/kg	moles/kg
o	0.91080E+06	0.89001E+06	0.5562741837E+02
ca	421.93	412.30	0.1028743958E-01
cl	19805.	19353.	0.5458822606E+00
h	0.11451E+06	0.11190E+06	0.1110187334E+03
c	24.854	24.286	0.2022000015E-02
k	408.42	399.10	0.1020760495E-01
mg	1322.0	1291.8	0.5314955849E-01
na	11020.	10768.	0.4683822421E+00
s	926.41	905.26	0.2823129792E-01

----- elemental composition as strict basis species -----

ecies	mg/l	mg/kg	moles/kg
h2o	0.10256E+07	0.10021E+07	0.5562741837E+02
ca++	421.93	412.30	0.1028743958E-01
cl-	19805.	19353.	0.5458822606E+00
h+	0.11451E+06	0.11190E+06	0.1110187334E+03
hco3-	126.26	123.38	0.2022000015E-02
k+	408.42	399.10	0.1020760495E-01
mg++	1322.0	1291.8	0.5314955849E-01
na+	11020.	10768.	0.4683822421E+00
so4--	2775.4	2712.0	0.2823129792E-01

--- equivalent composition of the aqueous phase (ete balances) ---

original basis		*existing basis*	
species	moles/kg h2o	species	moles/kg h2o
h2o	0.5562741837E+02	h2o	0.5562741837E+02
ca++	0.1028743958E-01	ca++	0.1028743958E-01
cl-	0.5458822606E+00	cl-	0.5458822606E+00
h+	0.1110187334E+03	h+	0.1110187334E+03
hco3-	0.2022000015E-02	hco3-	0.2022000015E-02
k+	0.1020760495E-01	k+	0.1020760495E-01
mg++	0.5314955849E-01	mg++	0.5314955849E-01
na+	0.4683822421E+00	na+	0.4683822421E+00
so4--	0.2823129792E-01	so4--	0.2823129792E-01

single ion activities and activity coefficients are here defined
with respect to the modified nbs ph scale

	ph	eh	pe
modified nbs ph scale	8.2200	0.5000	8.4522E+00
rational ph scale	8.1079	0.5066	8.5642E+00

phcl = 8.6852

activity of water = 0.98233
log activity of water = -0.00774

true osmotic coefficient= 0.91574
stoichiometric osmotic coefficient= 0.88520

sum of true molalities= 1.0809268297521
sum of stoichiometric molalities= 1.1182172082006

true ionic strength= 0.6226746935852
stoichiometric ionic strength= 0.6966516633063

----- electrical balance totals -----

	equiv/kg
sigma(mz) cations =	0.5599125897E+00
sigma(mz) anions =	-0.5589585101E+00
total charge =	0.1118871100E+01
mean charge =	0.5594355499E+00
charge imbalance =	0.9540796580E-03

total charge = sigma(mz) cations + abs (sigma(mz) anions)
mean charge = 1/2 total charge

the electrical imbalance is

0.853E-01 per cent of the total charge

```
0.171     per cent of the mean charge
0.170     per cent of sigma(mz) cations
0.171     per cent of abs ( sigma(mz) anions )
```

----- activity ratios of ions -----

```
log ( act(na++                    ) / act(h+)xx 2 ) =     13.7680
log ( act(cl-                     ) x act(h+)xx 1 ) =     -8.6852
log ( act(hco3-                   ) x act(h+)xx 1 ) =    -11.2683
log ( act(k+                      ) / act(h+)xx 1 ) =      6.0012
log ( act(mg++                    ) / act(h+)xx 2 ) =     14.5174
log ( act(na+                     ) / act(h+)xx 1 ) =      7.6792
log ( act(so4--                   ) x act(h+)xx 2 ) =    -19.0522
log ( act(acetic acid(aq)         )               ) =   -125.8688
log ( act(acetone(aq)             )               ) =   -248.3402
log ( act(clo4-                   ) x act(h+)xx 1 ) =    -63.0544
log ( act(co2(aq)                 )               ) =     -4.9159
log ( act(co3--                   ) x act(h+)xx 2 ) =    -21.5971
log ( act(ethane(aq)              )               ) =   -209.3925
log ( act(h2(aq)                  )               ) =    -36.4494
log ( act(hs-                     ) x act(h+)xx 1 ) =   -118.7090
log ( act(methane(aq)             )               ) =   -116.7571
log ( act(methanol(aq)            )               ) =   -100.1856
log ( act(o2(aq)                  )               ) =    -19.3301
log ( act(oh-                     ) x act(h+)xx 1 ) =    -14.0028
log ( act(so3--                   ) x act(h+)xx 2 ) =    -56.0115
```

----- distribution of aqueous species -----

species	molal conc	log conc	log g	activity	log act
cl-	0.5244E+00	-0.2804	-0.1848	0.3426E+00	-0.4652
na+	0.4449E+00	-0.3518	-0.1891	0.2878E+00	-0.5408
mg++	0.4072E-01	-1.3902	-0.5325	0.1195E-01	-1.9226
nacl(aq)	0.1648E-01	-1.7830	0.0000	0.1648E-01	-1.7830
so4--	0.1322E-01	-1.8788	-0.7334	0.2442E-02	-2.6122
k+	0.9981E-02	-2.0008	-0.2180	0.6042E-02	-2.2188
ca++	0.9268E-02	-2.0330	-0.6390	0.2128E-02	-2.6720
mgso4(aq)	0.7531E-02	-2.1232	0.0000	0.7531E-02	-2.1232
naso4-	0.6650E-02	-2.1772	-0.1559	0.4644E-02	-2.3331
mgcl+	0.4638E-02	-2.3336	-0.1891	0.3001E-02	-2.5227
hco3-	0.1281E-02	-2.8924	-0.1559	0.8946E-03	-3.0483
caso4(aq)	0.6712E-03	-3.1731	0.0000	0.6712E-03	-3.1731
nahco3(aq)	0.3672E-03	-3.4351	0.0000	0.3672E-03	-3.4351
cacl+	0.2271E-03	-3.6437	-0.1891	0.1470E-03	-3.8328
mghco3+	0.1794E-03	-3.7462	-0.1891	0.1161E-03	-3.9353
kso4-	0.1603E-03	-3.7955	-0.1559	0.1118E-03	-3.9514
mgco3(aq)	0.7927E-04	-4.1009	0.0000	0.7927E-04	-4.1009
kcl(aq)	0.6629E-04	-4.1786	0.0000	0.6629E-04	-4.1786
cacl2(aq)	0.5676E-04	-4.2460	0.0000	0.5676E-04	-4.2460
cahco3+	0.3277E-04	-4.4846	-0.1891	0.2120E-04	-4.6736
caco3(aq)	0.3147E-04	-4.5020	0.0000	0.3147E-04	-4.5020
co3--	0.3068E-04	-4.5132	-0.6439	0.6964E-05	-5.1571
co2(aq)	0.1080E-04	-4.9667	0.0508	0.1214E-04	-4.9159
naco3-	0.9382E-05	-5.0277	-0.1559	0.6552E-05	-5.1836
oh-	0.2523E-05	-5.5980	-0.1848	0.1649E-05	-5.7828
naoh(aq)	0.3100E-06	-6.5086	0.0000	0.3100E-06	-6.5086
caoh+	0.7574E-07	-7.1207	-0.1891	0.4901E-07	-7.3097
h+	0.7799E-08	-8.1079	-0.1121	0.6026E-08	-8.2200
koh(aq)	0.3416E-08	-8.4665	0.0000	0.3416E-08	-8.4665
hso4-	0.2008E-08	-8.6972	-0.1559	0.1402E-08	-8.8531
hcl(aq)	0.4415E-09	-9.3552	0.0000	0.4414E-09	-9.3552
mg4(oh)4++++	0.1287E-11	-11.8904	-2.7012	0.2561E-14	-14.5915
khso4(aq)	0.5789E-12	-12.2374	0.0000	0.5789E-12	-12.2374
o2(aq)	0.4161E-19	-19.3808	0.0508	0.4677E-19	-19.3301
h2so4(aq)	0.8468E-20	-20.0722	0.0000	0.8468E-20	-20.0722
clo-	0.8399E-25	-25.0758	-0.1559	0.5866E-25	-25.2317
hclo(aq)	0.1309E-25	-25.8832	0.0000	0.1309E-25	-25.8832
ho2-	0.2519E-29	-29.5988	-0.1559	0.1759E-29	-29.7547
formate	0.9274E-36	-36.0328	-0.1559	0.6476E-36	-36.1887
h2(aq)	0.3161E-36	-36.5001	0.0508	0.3553E-36	-36.4494
hso5-	0.2356E-37	-37.6279	-0.1559	0.1645E-37	-37.7838
so3--	0.1452E-38	-38.8381	-0.7334	0.2682E-39	-39.5715
hso3-	0.3713E-40	-40.4302	-0.1559	0.2593E-40	-40.5861
formic acid(aq)	0.2209E-40	-40.6558	0.0000	0.2209E-40	-40.6558
clo2-	0.1789E-42	-42.7473	-0.1559	0.1250E-42	-42.9032
clo3-	0.2721E-46	-46.5652	-0.1559	0.1900E-46	-46.7211
h2so3(aq)	0.1592E-46	-46.7980	0.0000	0.1592E-46	-46.7980
so2(aq)	0.1153E-46	-46.9382	0.0000	0.1153E-46	-46.9382
hclo2(aq)	0.1114E-47	-47.9532	0.0000	0.1114E-47	-47.9532
s2o8--	0.8163E-53	-53.0882	-0.7334	0.1508E-53	-53.8216
clo4-	0.2241E-54	-54.6496	-0.1848	0.1464E-54	-54.8344
s2o6--	0.2098E-62	-62.6782	-0.7334	0.3876E-63	-63.4116
s2o5--	0.5643E-85	-85.2485	-0.7334	0.1043E-85	-85.9819
methanol(aq)	0.6522-100	-100.1856	0.0000	0.6522-100	-100.1856
hs-	0.4964-110	-110.3042	-0.1848	0.3244-110	-110.4890
s2o4--	0.4126-110	-110.3844	-0.7334	0.7624-111	-111.1178
h2s(aq)	0.1900-111	-111.7213	0.0000	0.1900-111	-111.7213
s--	0.3483-114	-114.4707	-0.7334	0.6251-115	-115.2041
s2o3--	0.1576-115	-115.8025	-0.7334	0.2911-116	-116.5359
methane(aq)	0.1750-116	-116.7571	0.0000	0.1750-116	-116.7571

acetate	0.5623-122	-122.2500	-0.1559	0.3927-122	-122.4060
mgch3coo+	0.1350-122	-122.8695	-0.1891	0.8738-123	-123.0586
nach3coo(aq)	0.7468-123	-123.1268	0.0000	0.7468-123	-123.1268
hs2o3-	0.2595-123	-123.5859	-0.1559	0.1812-123	-123.7418
cach3coo+	0.1955-123	-123.7089	-0.1891	0.1265-123	-123.8980
acetic acid(aq)	0.1353-125	-125.8688	0.0000	0.1353-125	-125.8688
s3o6--	0.3869-147	-147.4124	-0.7334	0.7148-148	-148.1458
ethyne(aq)	0.2582-183	-183.5880	0.0000	0.2582-183	-183.5880
ethanol(aq)	0.2015-188	-188.6957	0.0000	0.2015-188	-188.6957
ethylene(aq)	0.7020-193	-193.1537	0.0000	0.7020-193	-193.1537
s2--	0.2120-196	-196.6346	-0.7334	0.4286-197	-197.3680
ethane(aq)	0.4050-209	-209.3925	0.0000	0.4050-209	-209.3925
propanoate	0.1592-212	-212.7979	-0.1559	0.1112-212	-212.9538
s4o6--	0.5749-216	-216.2404	-0.7334	0.1062-216	-216.9738
propanoic acid(aq)	0.5105-216	-216.2920	0.0000	0.5105-216	-216.2920
acetone(aq)	0.4569-248	-248.3402	0.0000	0.4569-248	-248.3402
1-propyne(aq)	0.7997-270	-270.0971	0.0000	0.7997-270	-270.0971
s3--	0.1329-278	-278.8766	-0.7334	0.2455-279	-279.6100
1-propanol(aq)	0.6533-279	-279.1849	0.0000	0.6533-279	-279.1849
1-propene(aq)	0.3364-281	-281.4731	0.0000	0.3364-281	-281.4731
propane(aq)	0.5617-300	-300.2505	0.0000	0.5617-300	-300.2505
butanoate	0.1481-303	-303.8295	-0.1559	0.1034-303	-303.9854
butanoic acid(aq)	0.4147-307	-307.3822	0.0000	0.4147-307	-307.3822
s5o6--	0.1125-313	-313.9488	-0.7334	0.2078-314	-314.6823
2-butanone(aq)	0.0000E+00	-339.0859	0.0000	0.0000E+00	-339.0859
1-butyne(aq)	0.0000E+00	-361.1213	0.0000	0.0000E+00	-361.1213
s4--	0.0000E+00	-361.3385	-0.7334	0.0000E+00	-362.0719
1-butanol(aq)	0.0000E+00	-370.8836	0.0000	0.0000E+00	-370.8836
1-butene(aq)	0.0000E+00	-372.6808	0.0000	0.0000E+00	-372.6808
n-butane(aq)	0.0000E+00	-391.1640	0.0000	0.0000E+00	-391.1640
pentanoate	0.0000E+00	-394.7512	-0.1559	0.0000E+00	-394.9071
pentanoic acid(aq)	0.0000E+00	-398.2820	0.0000	0.0000E+00	-398.2820
2-pentanone(aq)	0.0000E+00	-430.2495	0.0000	0.0000E+00	-430.2495
s5--	0.0000E+00	-444.0203	-0.7334	0.0000E+00	-444.7537
1-pentyne(aq)	0.0000E+00	-452.1309	0.0000	0.0000E+00	-452.1309
1-pentanol(aq)	0.0000E+00	-460.6032	0.0000	0.0000E+00	-460.6032
1-pentene(aq)	0.0000E+00	-463.7123	0.0000	0.0000E+00	-463.7123
n-pentane(aq)	0.0000E+00	-482.1474	0.0000	0.0000E+00	-482.1474
hexanoate	0.0000E+00	-485.7755	-0.1559	0.0000E+00	-485.9314
hexanoic acid(aq)	0.0000E+00	-489.2916	0.0000	0.0000E+00	-489.2916
2-hexanone(aq)	0.0000E+00	-521.0979	0.0000	0.0000E+00	-521.0979
1-hexyne(aq)	0.0000E+00	-543.2359	0.0000	0.0000E+00	-543.2359
1-hexanol(aq)	0.0000E+00	-552.2359	0.0000	0.0000E+00	-552.2359
1-hexene(aq)	0.0000E+00	-554.5535	0.0000	0.0000E+00	-554.5535
n-hexane(aq)	0.0000E+00	-573.2742	0.0000	0.0000E+00	-573.2742
heptanoate	0.0000E+00	-576.7339	-0.1559	0.0000E+00	-576.8898
heptanoic acid(aq)	0.0000E+00	-580.1547	0.0000	0.0000E+00	-580.1547
2-heptanone(aq)	0.0000E+00	-612.0489	0.0000	0.0000E+00	-612.0489
1-heptyne(aq)	0.0000E+00	-634.3775	0.0000	0.0000E+00	-634.3775
1-heptanol(aq)	0.0000E+00	-644.2425	0.0000	0.0000E+00	-644.2425
1-heptene(aq)	0.0000E+00	-645.5264	0.0000	0.0000E+00	-645.5264
n-heptane(aq)	0.0000E+00	-664.2253	0.0000	0.0000E+00	-664.2253
octanoate	0.0000E+00	-667.6849	-0.1559	0.0000E+00	-667.8408
octanoic acid(aq)	0.0000E+00	-670.8858	0.0000	0.0000E+00	-670.8858
2-octanone(aq)	0.0000E+00	-702.9999	0.0000	0.0000E+00	-702.9999
1-octyne(aq)	0.0000E+00	-725.3872	0.0000	0.0000E+00	-725.3872
1-octanol(aq)	0.0000E+00	-734.9736	0.0000	0.0000E+00	-734.9736
1-octene(aq)	0.0000E+00	-736.6386	0.0000	0.0000E+00	-736.6386
n-octane(aq)	0.0000E+00	-755.2202	0.0000	0.0000E+00	-755.2202

----- major aqueous species contributing to mass balances -----

aqueous species accounting for 99% or more of ca++

species	molal conc	per cent
ca++	0.9268E-02	90.09
caso4(aq)	0.6712E-03	6.52
cacl+	0.2271E-03	2.21
cacl2(aq)	0.5676E-04	0.55
total		99.37

aqueous species accounting for 99% or more of cl-

species	molal conc	per cent
cl-	0.5244E+00	96.06
nacl(aq)	0.1648E-01	3.02
total		99.08

aqueous species accounting for 99% or more of hco3-

species	molal conc	per cent
hco3-	0.1281E-02	63.36
nahco3(aq)	0.3672E-03	18.16
mghco3+	0.1794E-03	8.87

mgco3(aq)	0.7927E-04	3.92
cahco3+	0.3277E-04	1.62
caco3(aq)	0.3147E-04	1.56
co3--	0.3068E-04	1.52
total		99.00

aqueous species accounting for 99% or more of k+

species	molal conc	per cent
k+	0.9981E-02	97.78
kso4-	0.1601E-03	1.57
total		99.35

aqueous species accounting for 99% or more of mg++

species	molal conc	per cent
mg++	0.4072E-01	76.62
mgso4(aq)	0.7531E-02	14.17
mgcl+	0.4638E-02	8.73
total		99.51

aqueous species accounting for 99% or more of na+

species	molal conc	per cent
na+	0.4449E+00	94.98
nacl(aq)	0.1648E-01	3.52
naso4-	0.6650E-02	1.42
total		99.92

aqueous species accounting for 99% or more of so4--

species	molal conc	per cent
so4--	0.1322E-01	46.82
mgso4(aq)	0.7531E-02	26.68
naso4-	0.6650E-02	23.56
caso4(aq)	0.6712E-03	2.38
total		99.43

----- summary of aqueous redox reactions -----

couple	eh, volts	pe-	log fo2	ah, kcal
default	0.500	0.8452E+01	-16.432	11.531
acetic a/hco3-	0.500	0.8452E+01	-16.432	11.531
acetone(/hco3-	0.500	0.8452E+01	-16.432	11.531
clo4- /cl-	0.500	0.8452E+01	-16.432	11.531
ethane(a/hco3-	0.500	0.8452E+01	-16.432	11.531
h2(aq) /h2o	*0.500*	*0.8452E+01*	*-16.432*	*11.531*
hs- /so4--	0.500	0.8452E+01	-16.432	11.531
methane(/hco3-	0.500	0.8452E+01	-16.432	11.531
methanol/hco3-	0.500	0.8452E+01	-16.432	11.531
o2(aq) /h2o	0.500	0.8452E+01	-16.432	11.531
so3-- /so4--	0.500	0.8452E+01	-16.432	11.531

----- summary of aqueous non-equilibrium non-redox reactions -----

couple	affinity, kcal
none	

----- summary of stoichiometric mineral saturation states -----

(minerals with affinities .lt. -10 kcal are not listed)

mineral	log q/k	aff, kcal	state	mineral	log q/k	aff, kcal	state
anhydrite	-0.978	-1.334		aragonite	0.507	0.691	ssatd
arcanite	-5.249	-7.161		artinite	-1.928	-2.631	
bassanite	-1.027	-2.219		bischofite	-7.292	-9.948	
bloedite	-5.782	-7.888		brucite	-1.796	-2.450	
calcite	0.651	0.888	ssatd	caso4:0.5h2o(beta)	-1.795	-2.448	
dolomite	3.235	4.414	ssatd	dolomite-dis	1.691	2.307	ssatd
dolomite-ord	3.235	4.414	ssatd	epsomite	-2.627	-3.584	
gaylussite	-4.613	-6.294		glauberite	-3.509	-4.787	
gypsum	-0.817	-1.115		halite	-2.592	-3.536	

```
hexahydrite              -2.855    -3.894                 huntite                   1.946     2.654  ssatd
hydromagnesite           -3.387    -4.621                 ice                      -0.146    -0.200  satd
kainite                  -6.931    -9.455                 kalicinite               -5.551    -7.573
kieserite                -4.276    -5.833                 lansfordite              -1.631    -2.225
magnesite                 0.955     1.303  ssatd          mg1.25so4(oh)0.5:0       -6.024    -8.219
mgl.5so4(oh)             -6.222    -8.488                 mirabilite               -2.632    -3.590
monohydrocalcite         -3.190    -0.260  satd           na2co3                   -7.091    -9.674
na2co3:7h2o              -5.910    -8.063                 na4ca(so4)3:2h2o         -6.794    -9.269
nahcolite                -3.476    -4.743                 natron                   -5.598    -7.637
nesquehonite             -1.770    -2.414                 oxychloride-mg           -5.537    -7.553
pentahydrite             -3.186    -4.347                 periclase                -6.826    -9.312
picromerite              -7.192    -9.811                 pirssonite               -4.749    -6.479
starkeyite               -3.566    -4.865                 sylvite                  -3.530    -4.816
syngenite                -4.742    -6.469                 thenardite               -3.385    -4.618
thermonatrite            -6.880    -9.386

    2 approx. saturated pure minerals
    0 approx. saturated end-members of specified solid solutions
    0 saturated end-members of hypothetical solid solutions


    7 supersaturated pure minerals
    0 supersatd. end-members of specified solid solutions
    0 supersatd. hypothetical solid solution phases


                    ----- summary of gases -----

      gas                        fugacity    log fugacity

   c(g)                      0.195644-173    -173.70853
   ca(g)                     0.219654-141    -141.65826
   ch4(g)                    0.123922-113    -113.90685
   cl2(g)                    0.937359E-30     -30.02809
   co(g)                     0.307670E-40     -40.29442
   co2(g)                    0.357270E-03      -3.44700
   h2(g)                     0.452523E-33     -33.34436
   h2o(g)                    0.255185E-01      -1.59314
   h2s(g)                    0.184894-110    -110.73308
   hcl(g)                    0.101490E-14     -14.99358
   k(g)                      0.174855E-70     -70.75732
   mg(g)                     0.830804-118    -118.08050
   n2(g)                     0.434634E-68     -68.36188
   o2(g)                     0.370023E-16     -16.43177
   s2(g)                     0.372234-180    -180.42918
   so2(g)                    0.779493E-47     -47.10819

------ end of output -----
--- pickup file has been successfully written ---

--- reading the input file ---

--- no further input found ---

         start time =  09:31       3Dec91
           end time =  09:31       3Dec91

         user time =       5.170
          cpu time =       0.960
Normal exit
```

7.3. The Sea Water Test Case, Using Pitzer's Equations

The preceding test case can also be run using Pitzer's equations. The results presented here were obtained using the **hmw** data file, which is based on Harvie, Møller, and Weare (1984). The only functional difference between the **input** file used here and that in the previous case is that **iopg1** = 1 in the present case. The **input** files are shown in both "W" and "D" formats. The **output** file is reproduced here beginning with the message announcing the end of Newton-Raphson iteration.

The results differ from those in the previous section because the underlying models are different. In fact, some quantities, including individual species molalities and activity coefficients, can not be compared in a meaningful way. This is due to the fact that the models utilize different sets of species to represent the systems. Note that far fewer species appear in the present case. The only

ion pairs represented are $CaCO_{3(aq)}$, $MgCO_{3(aq)}$, and $MgOH^+$. Also note that no organics appear on the present **output** file, as there are none in this model.

Some parameters which can be compared are the activities and activity coefficients of the electrically neutral species common to both models (mostly basis species). For the ions, electrically neutral combinations of these parameters can also be compared. The single-ion activities and activity coefficients themselves can be compared, but not in a truly exact sense. Note that in both the present case and the previous one, the **iopg1** option switch is set to 0, causing all single-ion activities and activity coefficients to be normalized to the NBS *pH* scale. However, the exact definition of this scale in a solution as concentrated as sea water has some ionic strength dependence, and the ionic strength of this water is slightly different in the two models (0.6964 versus 0.6227 molal). The saturation indices of the various minerals common to both models can be compared with no problem. In the present case, the *SI* for calcite is +0.645; in the previous case, it was +0.651. This is reasonably good agreement, though the results in the present case are without *doubt more accurate.*

The EQ3NR **input** file (**swmajp.3i**), the sea water benchmark test case using Pitzer's equations ("W" format):

```
EQ3NR input file name= swmajp.3i
Description= "Sea water, major ions only, using Pitzer's equations"
Version number= 3245     Stage number= 01
Created 06/08/90     Creator= T.J. Wolery
Revised 06/08/90     Revisor= T.J. Wolery

  Sea water, including only the major ions. This is a considerably
pared-down version of swtst.3i, which contains the full benchmark sea
water test case of Nordstrom et al. (1979, Table III).

  Purpose: to test the code on a small problem involving a moderately
concentrated solution, using Pitzer's equations to calculate the
activity coefficients of the aqueous species. Input file swmaj.3i
runs the same problem using the B-dot equation and related equations
instead of Pitzer's equations. Input file swmajd.3i runs the same
problem using the Davies equation.

  This problem is best addressed using the thermodynamic data base of
Harvie, Moller, and Weare (1984).

                        References

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-SO4-
  OH-HCO3-CO3-CO2-H2O system to high ionic strengths at 25 C:
  Geochimica et Cosmochimica Acta, v. 48, p. 723-751.

Nordstrom, D. K., et al., 1979, A comparison of computerized chemical
  models for equilibrium calculations in aqueous systems, in Jenne,
  E. A., editor, Chemical Modeling in Aqueous Systems, ACS Symposium
  Series, v. 93, American Chemical Society, Washington, D.C.,
  p. 857-892.

endit.
      tempc=        25.
        rho=    1.02336     tdspkg=          0.     tdspl=          0.
        fep=      0.500     uredox=
      tolbt=         0.      toldl=          0.    tolsat=          0.
     itermx=  0
*              1    2    3    4    5    6    7    8    9   10
   ioptl-10=  -1    0    0    0    0    0    0    0    0    0
   iopg1-10=   1    0    0    0    0    0    0    0    0    0
   iopr1-10=   0    0    0    0    0    0    0    0    0    0
  iopr11-20=   0    0    0    0    0    0    0    0    0    0
   iodb1-10=   0    0    0    0    0    0    0    0    0    0
      webal= none
     nxmod=  0
*
data file master species= na+
   switch with species=
    jflag=  3    csp=  10768.
data file master species= k+
```

```
   switch with species=
   jflag=  3   csp= 399.1
data file master species= ca++
   switch with species=
   jflag=  3   csp= 412.3
data file master species= mg++
   switch with species=
   jflag=  3   csp= 1291.8
data file master species= h+
   switch with species=
   jflag= 16   csp= -8.22
data file master species= hco3-
   switch with species=
   jflag=  0   csp= .002022
data file master species= cl-
   switch with species=
   jflag=  3   csp= 19353.
data file master species= so4--
   switch with species=
   jflag=  3   csp= 2712.
endit.
```

The EQ3NR input file (swmajp.3i), sea water benchmark test case using Pitzer's equations ("D" format):

```
-------------------------------------------------------------------------------
EQ3NR input file name= swmajp.3i
Description= "Sea water, major ions only, using Pitzer's equations"
Version number= 3245   Stage number= 01
Created 06/08/90    Creator= T.J. Wolery
Revised 06/00/90    Revisor= T.J. Wolery

  Sea water, including only the major ions. This is a considerably
pared-down version of swtst.3i, which contains the full benchmark sea
water test case of Nordstrom et al. (1979, Table III).

  Purpose: to test the code on a small problem involving a moderately
concentrated solution, using Pitzer's equations to calculate the
activity coefficients of the aqueous species. Input file swmaj.3i
runs the same problem using the B-dot equation and related equations
instead of Pitzer's equations. Input file swmajd.3i runs the same
problem using the Davies equation.

  This problem is best addressed using the thermodynamic data base of
Harvie, Moller, and Weare (1984).

                          References

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-SO4-
  OH-HCO3-CO3-CO2-H2O system to high ionic strengths at 25 C:
  Geochimica et Cosmochimica Acta, v. 48, p. 723-751.

Nordstrom, D. K., et al., 1979, A comparison of computerized chemical
  models for equilibrium calculations in aqueous systems, in Jenne,
  E. A., editor, Chemical Modeling in Aqueous Systems, ACS Symposium
  Series, v. 93, American Chemical Society, Washington, D.C.,
  p. 857-892.

-------------------------------------------------------------------------------
Temperature (C)          | 25.00        |Density(gm/cm3)|  1.02336
-------------------------------------------------------------------------------
Total Dissolved Salts    |              | mg/kg | mg/l  |*not used
-------------------------------------------------------------------------------
Electrical Balancing on  |              | code selects  |*not performed
-------------------------------------------------------------------------------
SPECIES    | BASIS SWITCH/CONSTRAINT | CONCENTRATION| UNITS OR TYPE
-------------------------------------------------------------------------------
redox      |                         | 0.5000       |eh
na+        |                         | 10768.       |mg/kg
k+         |                         | 399.10       |mg/kg
ca++       |                         | 412.30       |mg/kg
mg++       |                         | 1291.8       |mg/kg
h+         |                         | 8.2200       |ph
hco3-      |                         |0.20220E-02   |molality
cl-        |                         | 19353.       |mg/kg
so4--      |                         | 2712.0       |mg/kg
-------------------------------------------------------------------------------
Input Solid Solutions
-------------------------------------------------------------------------------
 none      |                         |              |
-------------------------------------------------------------------------------
SUPPRESSED SPECIES   (suppress,replace,augmentx,augmenty)    value
-------------------------------------------------------------------------------
 none                |                         |
-------------------------------------------------------------------------------
```

```
OPTIONS
-------------------------------------------------------------------------
 - SOLID SOLUTIONS -
    * ignore solid solutions
      process hypothetical solid solutions
      process input and hypothetical solid solutions
 - LOADING OF SPECIES INTO MEMORY -
    * does nothing
      lists species loaded into memory
 - ECHO DATABASE INFORMATION -
    * does nothing
      lists all reactions
      lists reactions and log K values
      lists reactions, log K values and polynomial coef.
 - LIST OF AQUEOUS SPECIES (ordering) -
    * in order of decreasing concentration
      in same order as input file
 - LIST OF AQUEOUS SPECIES (concentration limit) -
    * all species
      only species > 10**-20 molal
      only species > 10**-12 molal
      not printed
 - LIST OF AQUEOUS SPECIES (by element) -
    * print major species
      print all species
      don't print
 - MINERAL SATURATION STATES -
    * print if affinity > -10 kcals
      print all
      don't print
 - pH SCALE CONVENTION -
    * modified NBS
      internal
      rational
 - ACTIVITY COEFFICIENT OPTIONS -
      use B-dot equation
      Davies' equation
    * Pitzer's equations
 - AUTO BASIS SWITCHING -
    * off
      on
 - PITZER DATABASE INFORMATION -
    * print only warnings
      print species in model and number of Pitzer coefficients
      print species in model and names of Pitzer coefficients
 - PICKUP FILE -
    * write pickup file
      don't write pickup file
 - LIST MEAN IONIC PROPERTIES -
    * don't print
      print
 - LIST AQUEOUS SPECIES, ION SIZES, AND HYDRATION NUMBERS -
    * print
      don't print
 - CONVERGENCE CRITERIA -
    * test both residual functions and correction terms
      test only residual functions
-------------------------------------------------------------------------
DEBUGGING SWITCHES (0-off, 1,2-on, default is off)
-------------------------------------------------------------------------
0  generic debugging information                                        2
0  print details of pre-Newton-Raphson iteration                        2
0  print details of Newton-Raphson iteration
0  print details of stoichiometric factors                              2
0  print details of stoichiometric factors calculation
0  write reactions on RLIST
0  list stoichiometric concentrations of master species
0  request iteration variables to be killed
-------------------------------------------------------------------------
DEVELOPMENT OPTIONS  (used for code development)
-------------------------------------------------------------------------
 none
-------------------------------------------------------------------------
TOLERANCES                    (desired values)        (defaults)
-------------------------------------------------------------------------
      residual functions  |                        |1.e-10
        correction terms  |                        |1.e-10
        saturation state  |                        |0.5
number of N-R iterations  |                        |30
-------------------------------------------------------------------------
```

The EQ3NR **output** file (**swmaip.3o**), the sea water benchmark test case using Pitzer's equations (beginning with the message announcing the end of Newton-Raphson iteration):

```
                    .
                    .
                    .
          (Material deleted)
                    .
                    .
                    .
Hybrid newton-raphson iteration converged in   4 steps.


                  ----- Summary of the Aqueous Phase -----

       ----- Elemental composition of the aqueous phase -----

    element       mg/l          mg/kg          moles/kg

      o       0.91080E+06    0.89001E+06    0.5562742039E+02
      ca        421.93         412.30         0.1028743949E-01
      cl        19805.         19353.         0.5458822600E+00
      h       0.11451E+06    0.11190E+06    0.1110186924E+03
      c         24.854         24.286         0.2[illegible]E-02
      k         408.42         399.10         0.1020760493E-01
      mg        1322.0         1291.8         0.5314955770E-01
      na        11020.         10768.         0.4683822413E+00
      s         926.41         905.26         0.2823129677E-01

       ----- elemental composition as strict basis species -----

    species          mg/l          mg/kg          moles/kg

  h2o           0.10256E+07    0.10021E+07    0.5562742039E+02
  ca++           421.92         412.30         0.1028743949E-01
  cl-            19805.         19353.         0.5458822600E+00
  h+            0.11451E+06    0.11190E+06    0.1110186924E+03
  hco3-          126.26         123.38         0.2022000000E-02
  k+             408.42         399.10         0.1020760493E-01
  mg++           1322.0         1291.8         0.5314955770E-01
  na+            11020.         10768.         0.4683822413E+00
  so4--          2775.4         2712.0         0.2823129677E-01

       --- equivalent composition of the aqueous phase (cte balances) ---

        original basis                 existing basis

  species          moles/kg h2o    species          moles/kg h2o

h2o              0.5562742039E+02  h2o              0.5562742039E+02
ca++             0.1028743949E-01  ca++             0.1028743949E-01
cl-              0.5458822600E+00  cl-              0.5458822600E+00
h+               0.1110186924E+03  h+               0.1110186924E+03
hco3-            0.2022000000E-02  hco3-            0.2022000000E-02
k+               0.1020760493E-01  k+               0.1020760493E-01
mg++             0.5314955770E-01  mg++             0.5314955770E-01
na+              0.4683822413E+00  na+              0.4683822413E+00
so4--            0.2823129677E-01  so4--            0.2823129677E-01

single ion activities and activity coefficients are here defined
with respect to the modified nbs ph scale


                                      ph          eh          pe

  modified nbs ph scale             8.2200      0.5000    8.4522E+00
  rational ph scale                 8.1132      0.5063    8.5590E+00

phcl =      8.6722


                        activity of water =  0.98198
                    log activity of water =  -0.00790

                  true osmotic coefficient=    0.90273
          stoichiometric osmotic coefficient=  0.90255

                     sum of true molalities=   1.1180579526987
          sum of stoichiometric molalities=    1.1182763340013

                     true ionic strength=    0.6964348915728
```

stoichiometric ionic strength= 0.6967640957883

----- electrical balance totals -----

	equiv/kg
sigma(mz) cations =	0.6052455191E+00
sigma(mz) anions =	-0.6043365584E+00
total charge =	0.1209582077E+01
mean charge =	0.6047910387E+00
charge imbalance =	0.9089606646E-03

total charge = sigma(mz) cations + abs (sigma(mz) anions)
mean charge = 1/2 total charge

the electrical imbalance is

0.751E-01	per cent of the total charge
0.150	per cent of the mean charge
0.150	per cent of sigma(mz) cations
0.150	per cent of abs (sigma(mz) anions)

----- activity ratios of ions -----

log (act(ca++) / act(h+)xx 2) =	13.7845
log (act(cl-) x act(h+)xx 1) =	-8.6722
log (act(hco3-) x act(h+)xx 1) =	-11.2062
log (act(k+) / act(h+)xx 1) =	6.0291
log (act(mg++) / act(h+)xx 2) =	14.5357
log (act(na+) / act(h+)xx 1) =	7.7243
log (act(so4--) x act(h+)xx 2) =	-19.0096
log (act(co2(aq)) =	-4.8609
log (act(co3--) x act(h+)xx 2) =	-21.5455
log (act(oh-) x act(h+)xx 1) =	-14.0046

----- distribution of aqueous species -----

species	molal conc	log conc	log g	activity	log act
cl-	0.5459E+00	-0.2629	-0.1893	0.3530E+00	-0.4522
na+	0.4684E+00	-0.3294	-0.1663	0.3194E+00	-0.4957
mg++	0.5306E-01	-1.2752	-0.6291	0.1247E-01	-1.9043
so4--	0.2823E-01	-1.5493	-1.0204	0.2694E-02	-2.5696
ca++	0.1026E-01	-1.9887	-0.6668	0.2210E-02	-2.6553
k+	0.1021E-01	-1.9911	-0.1998	0.6444E-02	-2.1909
hco3-	0.1816E-02	-2.7409	-0.2452	0.1032E-02	-2.9862
co3--	0.8641E-04	-4.0635	-1.0420	0.7844E-05	-5.1055
mgco3(aq)	0.8292E-04	-4.0813	0.0000	0.8292E-04	-4.0813
caco3(aq)	0.2456E-04	-4.6097	0.0000	0.2456E-04	-4.6097
co2(aq)	0.1224E-04	-4.9121	0.0512	0.1378E-04	-4.8609
mgoh+	0.3362E-05	-5.4735	-0.0279	0.3152E-05	-5.5014
oh-	0.3029E-05	-5.5188	-0.2658	0.1642E-05	-5.7846
h+	0.7706E-08	-8.1132	-0.1068	0.6026E-08	-8.2200
hso4-	0.2332E-08	-8.6322	-0.1788	0.1545E-08	-8.8110

----- major aqueous species contributing to mass balances -----

aqueous species accounting for 99% or more of ca++

species	molal conc	per cent
ca++	0.1026E-01	99.76
total		99.76

aqueous species accounting for 99% or more of cl-

species	molal conc	per cent
cl-	0.5459E+00	100.00
total		100.00

aqueous species accounting for 99% or more of hco3-

species	molal conc	per cent
hco3-	0.1816E-02	89.81
co3--	0.8641E-04	4.27
mgco3(aq)	0.8292E-04	4.10
caco3(aq)	0.2456E-04	1.21
total		99.39

aqueous species accounting for 99% or more of k+

species	molal conc	per cent
k+	0.1021E-01	100.00
total		100.00

aqueous species accounting for 99% or more of mg++

species	molal conc	per cent
mg++	0.5306E-01	99.84
total		99.84

aqueous species accounting for 99% or more of na+

species	molal conc	per cent
na+	0.4684E+00	100.00
total		100.00

aqueous species accounting for 99% or more of so4--

species	molal conc	per cent
so4--	0.2823E-01	100.00
total		100.00

----- summary of aqueous redox reactions -----

couple	eh, volts	pe-	log fo2	ah, kcal
default	0.500	0.8452E+01	-16.432	11.531

----- summary of aqueous non-equilibrium non-redox reactions -----

couple	affinity, kcal
none	

----- summary of stoichiometric mineral saturation states -----

(minerals with affinities .lt. -10 kcal are not listed)

mineral	log q/k	aff, kcal	state	mineral	log q/k	aff, kcal	state
anhydrite	-0.863	-1.177		aragonite	0.458	0.626	ssatd
arcanite	-5.175	-7.060		bischofite	-7.311	-9.975	
bloedite	-5.720	-7.803		brucite	-2.589	-3.532	
calcite	0.645	0.880	ssatd	dolomite	2.312	3.154	ssatd
epsomite	-2.648	-3.613		gaylussite	-4.476	-6.107	
glauberite	-3.542	-4.832		gypsum	-0.660	-0.901	
halite	-2.518	-3.436		hexahydrite	-2.886	-3.938	
kainite	-6.948	-9.479		kalicinite	-5.458	-7.447	
kieserite	-4.359	-5.947		magnesite	0.824	1.124	ssatd
mirabilite	-2.412	-3.291		na2co3:7h2o	-5.692	-7.765	
na4ca(so4)3:2h2o	-6.691	-9.128		nahcolite	-3.079	-4.200	
natron	-5.351	-7.300		nesquehonite	-1.866	-2.546	
oxychloride-mg	-5.686	-7.757		picromerite	-7.145	-9.748	
pirssonite	-4.630	-6.317		sylvite	-3.543	-4.834	
syngenite	-4.736	-6.461		thenardite	-3.274	-4.466	
thermonatrite	-6.587	-8.986					

0 approx. saturated pure minerals
0 approx. saturated end-members of specified solid solutions
0 saturated end-members of hypothetical solid solutions

4 supersaturated pure minerals
0 supersatd. end-members of specified solid solutions
0 supersatd. hypothetical solid solution phases

----- summary of gases -----

gas	fugacity	log fugacity

```
co2(g)                          0.417773E-03     -3.37906
h2(g)                           0.459241E-33    -33.33796
o2(g)                           0.369764E-16    -16.43208

----- end of output -----
--- pickup file has been successfully written ---

--- reading the input file ---

--- no further input found ---

        start time = 17:13       5Dec91
          end time = 17:13       5Dec91

        user time =       1.650
         cpu time =       0.750
normal exit
```

7.4. Using Mineral Solubility Constraints: An Example

This test case is taken from INTERA (1983), which used it to compare EQ3/6 with PHREEQE (Parkhurst, Plummer, and Thorstenson, 1980). In it, dissolved calcium is constrained to satisfy equilibrium with calcite and dissolved iron is constrained to satisfy equilibrium with hematite. The bicarbonate is constrained to satisfy an equilibrium CO_2 fugacity of 10^{-2} bar. On top of that, the *pH* is adjusted to satisfy electrical balance. The **input** files are presented here in both formats. The **output** file is presented beginning with the message announcing the end of Newton-Raphson iteration. The results shown were obtained using the **com** data file. The activity coefficients were computed from the B-dot equation. Note the appearance of the organic species, which all have negligible concentrations at the relatively high oxygen fugacity.

Note that the calculated saturation indices for calcite and hematite written on the output file are indeed zero. Similarly, the calculated equilibrium CO_2 fugacity is 10^{-2} bar. The *pH* adjustment is relatively minor. In problems such as this, if the input *pH* value is too far off the mark (usually more than 1-3 *pH* units), convergence problems are likely to occur. More than that, problems using some of the kind of constraints used here, especially in combination, may be ill-defined and have no realistic solution.

The EQ3NR input file (oxcalhem.3i), mineral solubility equilibrium test case ("W" format):

```
EQ3NR input file name= oxcalhem.3i
Description= "Oxygenated, calcite and hematite saturated solution"
Version number= 3245   Stage number= 01
Created 06/08/90     Creator= T.J. Wolery
Revised 06/08/90     Revisor= T.J. Wolery

  Oxygenated, calcite and hematite saturated solution. This problem
is part of Example 4 from INTERA (1983), who report a comparison study
of EQ3/6 with PHREEQE (Parkhurst, Thorstenson, and Plummer, 1980).

  Purpose: to provide a pickup file for construction of the EQ6 test
case methane.6i.

  In the orginal problem, uranium was specified in terms of
U++++.

                          References

INTERA Environmental Consultants, Inc., 1983, Geochemical Models
  Suitable for Performance Assessment of Nuclear Waste Storage:
  Comparison of PHREEQE and EQ3/EQ6: Office of Nuclear Waste
  Isolation, Battelle Project Management Division, Columbus, Ohio,
  ONWI-473, 114 p.

Parkhurst, D. L., Thorstenson, D. C., and Plummer, L. N., 1980,
  PHREEQE- A Computer Program for Geochemical Calculations: Water
  Resources Investigations Report 80-96, U.S. Geological Survey,
```

```
  Reston, Virginia, 210 p.

endit.
     tempc=          25.
       rho=           1.    tdspkg=          0.     tdspl=          0.
       fep=       -0.70     uredox=
     tolbt=           0.     toldl=          0.   tolsat=          0.
    itermx=  0
*              1    2    3    4    5    6    7    8    9   10
  iopt1-10=    0    0    0    0    0    0    0    0    0    0
  iopg1-10=    0    0    0    0    0    0    0    0    0    0
  iopr1-10=    0    0    0    0    0    0    0    0    0    0
 iopr11-20=    0    0    0    0    0    0    0    0    0    0
  iodb1-10=    0    0    0    0    0    0    0    0    0    0
     uebal= h+
     nxmod=  0
*
data file master species= na+
    switch with species=
    jflag=  0   csp= 7.0e-3
data file master species= ca++
    switch with species=
    jflag= 19   csp= 0.
 mineral= calcite
data file master species= fe++
    switch with species=
    jflag= 19   csp= 0.
 mineral= hematite
data file master species= uo2++
    switch with species=
    jflag=  0   csp= 4.0e-5
data file master species= hco3-
    switch with species=
    jflag= 21   csp= -2.0
 mineral= co2(g)
data file master species= so4--
    switch with species=
    jflag=  0   csp= 1.0e-3
data file master species= cl-
    switch with species=
    jflag=  0   csp= 5.0e-3
data file master species= h+
    switch with species=
    jflag= 16   csp= -7.40
endit.
```

The EQ3NR input file (oxcalhem.3i), mineral solubility equilibrium test case ("D" format):

```
-------------------------------------------------------------------------------
EQ3NR input file name= oxcalhem.3i
Description= 'Oxygenated, calcite and hematite saturated solution'
Version number= 3245    Stage number= 01
Created 06/08/90     Creator= T.J. Wolery
Revised 06/08/90     Revisor= T.J. Wolery

  Oxygenated, calcite and hematite saturated solution. This problem
is part of Example 4 from INTERA (1983), who report a comparison study
of EQ3/6 with PHREEQE (Parkhurst, Thorstenson, and Plummer, 1980).

  Purpose: to provide a pickup file for construction of the EQ6 test
case methane.6i.

  In the orginal problem, uranium was specified in terms of
U++++.

                              References

INTERA Environmental Consultants, Inc., 1983, Geochemical Models
  Suitable for Performance Assessment of Nuclear Waste Storage:
  Comparison of PHREEQE and EQ3/EQ6: Office of Nuclear Waste
  Isolation, Battelle Projec' Management Division, Columbus, Ohio,
  ONWI-473, 114 p.

Parkhurst, D. L., Thorstenson, D. C., and Plummer, L. N., 1980,
  PHREEQE- A Computer Program for Geochemical Calculations: Water
  Resources Investigations Report 80-96, U.S. Geological Survey,
  Reston, Virginia, 210 p.

-------------------------------------------------------------------------------
Temperature (C)          | 25.00        |Density(gm/cm3)|  1.00000
-------------------------------------------------------------------------------
Total Dissolved Salts    |              | mg/kg | mg/l  |*not used
-------------------------------------------------------------------------------
Electrical Balancing on |h+             | code selects  | not performed
-------------------------------------------------------------------------------
SPECIES    | BASIS SWITCH/CONSTRAINT | CONCENTRATION| UNITS OR TYPE
-------------------------------------------------------------------------------
```

redox		-.7000	logfo2
na+		0.70000E-02	molality
ca++	calcite	0.	mineral
fe++	hematite	0.	mineral
su2++		0.40000E-04	molality
hco3-	co2(g)	-2.0000	log fugacity
so4--		0.10000E-02	molality
cl-		0.50000E-02	molality
h+		7.4000	ph

Input Solid Solutions

none

SUPPRESSED SPECIES (suppress,replace,augmentk,augmentg) value

none

OPTIONS

- SOLID SOLUTIONS -
  * ignore solid solutions
    process hypothetical solid solutions
    process input and hypothetical solid solutions
- LOADING OF SPECIES INTO MEMORY -
  * does nothing
    lists species loaded into memory
- ECHO DATABASE INFORMATION -
  * does nothing
    lists all reactions
    lists reactions and log K values
    lists reactions, log K values and polynomial coef.
- LIST OF AQUEOUS SPECIES (ordering) -
  * in order of decreasing concentration
    in same order as input file
- LIST OF AQUEOUS SPECIES (concentration limit) -
  * all species
    only species > 10**-20 molal
    only species > 10**-12 molal
    not printed
- LIST OF AQUEOUS SPECIES (by element) -
  * print major species
    print all species
    don't print
- MINERAL SATURATION STATES -
  * print if affinity > -10 kcals
    print all
    don't print
- pH SCALE CONVENTION -
  * modified NBS
    internal
    rational
- ACTIVITY COEFFICIENT OPTIONS -
  * use B-dot equation
    Davies' equation
    Pitzer's equations
- AUTO BASIS SWITCHING -
  * off
    on
- PITZER DATABASE INFORMATION -
  * print only warnings
    print species in model and number of Pitzer coefficients
    print species in model and names of Pitzer coefficients
- PICKUP FILE -
  * write pickup file
    don't write pickup file
- LIST MEAN IONIC PROPERTIES -
  * don't print
    print
- LIST AQUEOUS SPECIES, ION SIZES, AND HYDRATION NUMBERS -
  * print
    don't print
- CONVERGENCE CRITERIA -
  * test both residual functions and correction terms
    test only residual functions

DEBUGGING SWITCHES (0-off, 1,2-on, default is off)

0	generic debugging information	2
0	print details of pre-Newton-Raphson iteration	2
0	print details of Newton-Raphson iteration	
0	print details of stoichiometric factors	2
0	print details of stoichiometric factors calculation	
0	write reactions on RLIST	
0	list stoichiometric concentrations of master species	
0	request iteration variables to be killed	

DEVELOPMENT OPTIONS (used for code development)

none

TOLERANCES	(desired values)	(defaults)

residual functions		1.e-10
correction terms		1.e-10
saturation state		0.5
number of N-R iterations		30

The EQ3NR **output** file (**oxcalhem.3o**), mineral solubility equilibrium test case (beginning with the message announcing the end of Newton-Raphson iteration):

•
•
•
(Material deleted)
•
•
•

Hybrid newton-raphson iteration converged in 6 steps.

----- Summary of the Aqueous Phase -----

----- Elemental composition of the aqueous phase -----

element	mg/l	mg/kg	moles/kg
o	0.88836E+06	0.88836E+06	0.5552474398E+02
ca	74.248	74.248	0.1852584728E-02
cl	177.26	177.26	0.5000000000E-02
fe	0.50317E-07	0.50317E-07	0.9009750988E-12
h	0.11190E+06	0.11190E+06	0.1110204565E+03
c	48.284	48.284	0.4019986372E-02
na	160.93	160.93	0.7000000000E-02
s	32.066	32.066	0.1000000001E-02
u	9.5212	9.5212	0.3999999985E-04

----- elemental composition as strict basis species -----

species	mg/l	mg/kg	moles/kg
h2o	0.10003E+07	0.10003E+07	0.5552474398E+02
ca++	74.248	74.248	0.1852584728E-02
cl-	177.26	177.26	0.5000000000E-02
fe++	0.50317E-07	0.50317E-07	0.9009750988E-12
h+	0.11190E+06	0.11190E+06	0.1110204565E+03
hco3-	245.29	245.29	0.4019986372E-02
na+	160.93	160.93	0.7000000000E-02
so4--	96.064	96.064	0.1000000001E-02
uo2++	10.801	10.801	0.3999999985E-04

--- equivalent composition of the aqueous phase (cte balances) ---

original basis		existing basis	
species	moles/kg h2o	species	moles/kg h2o
h2o	0.5552474398E+02	h2o	0.5552474398E+02
ca++	0.1852584728E-02	ca++	0.1852584728E-02
cl-	0.5000000000E-02	cl-	0.5000000000E-02
fe++	0.9009750988E-12	fe++	0.9009750988E-12
h+	0.1110204565E+03	h+	0.1110204565E+03
hco3-	0.4019986372E-02	hco3-	0.4019986372E-02
na+	0.7000000000E-02	na+	0.7000000000E-02
so4--	0.1000000001E-02	so4--	0.1000000001E-02
uo2++	0.3999999985E-04	uo2++	0.3999999985E-04

Single ion activities and activity coefficients are here defined
with respect to the modified nbs ph scale

	ph	eh	pe
modified nbs ph scale	7.3108	0.7862	1.3290E+01
rational ph scale	7.2655	0.7889	1.3336E+01

phcl = 9.6624

```
                    activity of water =  0.99968
                log activity of water =   -0.00014

           true osmotic coefficient=     0.95018
stoichiometric osmotic coefficient=     0.90417

          sum of true molalities=     0.0188859440972
sum of stoichiometric molalities=     0.0198469958820

          true ionic strength=     0.0131467506839
stoichiometric ionic strength=     0.0139741601196

 ----- electrical balance totals -----

                                        equiv/kg

             sigma(mz) cations =      0.1042838414E-01
             sigma(mz) anions  =     -0.1042838413E-01
                  total charge =      0.2085676827E-01
                   mean charge =      0.1042838414E-01
              charge imbalance =      0.7865843393E-12

 total charge = sigma(mz) cations + abs ( sigma(mz) anions )
 mean charge = 1/2 total charge


 the electrical imbalance is

            0.377E-08 per cent of the total charge
            0.754E-08 per cent of the mean charge
            0.754E-08 per cent of sigma(mz) cations
            0.754E-08 per cent of abs ( sigma(mz) anions )


--- electrical balancing on h+                          ---

    log activity

  input       -7.4000
  final       -7.3108
  adj          0.89175E-01


          ----- activity ratios of ions -----

log ( act(ca++                    ) / act(h+)xx 2 ) =     11.6624
log ( act(cl-                     ) x act(h+)xx 1 ) =     -9.6624
log ( act(fe++                    ) / act(h+)xx 2 ) =     -7.5360
log ( act(hco3-                   ) x act(h+)xx 1 ) =     -9.8137
log ( act(na+                     ) / act(h+)xx 1 ) =      5.0998
log ( act(so4--                   ) x act(h+)xx 2 ) =    -17.8693
log ( act(uo2++                   ) / act(h+)xx 2 ) =      3.8852
log ( act(acetic acid(aq)         )                 ) =   -154.4231
log ( act(acetone(aq)             )                 ) =   -306.9034
log ( act(clo4-                   ) x act(h+)xx 1 ) =    -32.5681
log ( act(co2(aq)                 )                 ) =     -3.4689
log ( act(co3--                   ) x act(h+)xx 2 ) =    -20.1425
log ( act(ethane(aq)              )                 ) =   -261.5369
log ( act(fe+++                   ) / act(h+)xx 3 ) =      0.0545
log ( act(h2(aq)                  )                 ) =    -44.3076
log ( act(hs-                     ) x act(h+)xx 1 ) =   -140.9896
log ( act(methane(aq)             )                 ) =   -146.7584
log ( act(methanol(aq)            )                 ) =   -122.3211
log ( act(o2(aq)                  )                 ) =     -3.5983
log ( act(oh-                     ) x act(h+)xx 1 ) =     -13.9952
log ( act(so3--                   ) x act(h+)xx 2 ) =     -62.6945
log ( act(u++                     ) / act(h+)xx 3 ) =     -58.4613
log ( act(u++++                   ) / act(h+)xx 4 ) =     -28.2603
log ( act(uo2+                    ) / act(h+)xx 1 ) =     -15.2316

          ----- distribution of aqueous species -----

     species          molal conc    log conc      log g     activity     log act

na+              0.6942E-02    -2.1585    -0.0525  0.6151E-02    -2.2110
cl-              0.4994E-02    -2.3015    -0.0500  0.4451E-02    -2.3516
hco3-            0.3511E-02    -2.4546    -0.0483  0.3141E-02    -2.5029
ca++             0.1721E-02    -2.7643    -0.1950  0.1098E-02    -2.9592
so4--            0.8941E-03    -3.0406    -0.1991  0.5654E-03    -3.2477
co2(aq)          0.3389E-03    -3.4699     0.0010  0.3397E-03    -3.4689
o2(aq)           0.2516E-03    -3.5993     0.0010  0.2522E-03    -3.5983
caso4(aq)        0.8021E-04    -4.0958     0.0000  0.8021E-04    -4.0958
cahco3+          0.4336E-04    -4.3629    -0.0525  0.3842E-04    -4.4154
nahco3(aq)       0.2755E-04    -4.5598     0.0000  0.2755E-04    -4.5598
naso4-           0.2568E-04    -4.5904    -0.0483  0.2298E-04    -4.6387
uo2(co3)2--      0.2458E-04    -4.6094    -0.1991  0.1554E-04    -4.8084
uo2(co3)3----    0.1235E-04    -4.9083    -0.8063  0.1930E-05    -5.7145
caco3(aq)        0.7031E-05    -5.1530     0.0000  0.7031E-05    -5.1530
co3--            0.4696E-05    -5.3283    -0.1926  0.3014E-05    -5.5209
nacl(aq)         0.4575E-05    -5.3396     0.0000  0.4575E-05    -5.3396
(uo2)2co3(oh)3-  0.1229E-05    -5.9103    -0.0483  0.1100E-05    -5.9587
```

cacl+	0.1112E-05	-5.9538	-0.0525	0.9854E-06	-6.0064
uo2(oh)2(aq)	0.3654E-06	-6.4372	0.0000	0.3654E-06	-6.4372
oh-	0.2321E-06	-6.6344	-0.0500	0.2068E-06	-6.6844
uo2co3(aq)	0.2277E-06	-6.6426	0.0000	0.2277E-06	-6.6426
naco3-	0.6773E-07	-7.1692	-0.0483	0.6060E-07	-7.2175
h+	*0.5427E-07*	-7.2655	-0.0454	0.4888E-07	-7.3108
uo2(oh)3-	0.1051E-07	-7.9783	-0.0483	0.9406E-08	-8.0266
cacl2(aq)	0.4944E-08	-8.3059	0.0000	0.4944E-08	-8.3059
caoh+	0.3581E-08	-8.4460	-0.0525	0.3173E-08	-8.4985
hso4-	0.2944E-08	-8.5311	-0.0483	0.2634E-08	-8.5794
uo2oh+	0.2623E-08	-8.5813	-0.0525	0.2324E-08	-8.6338
nash(aq)	0.8311E-09	-9.0803	0.0000	0.8311E-09	-9.0803
(uo2)3(co3)6(6-)	0.2507E-09	-9.6009	-1.8211	0.3784E-11	-11.4220
hcl(aq)	0.4652E-10	-10.3324	0.0000	0.4652E-10	-10.3324
uo2++	0.2936E-10	-10.5323	-0.2042	0.1835E-10	-10.7365
uo2so4(aq)	0.1200E-10	-10.9209	0.0000	0.1200E-10	-10.9209
(uo2)3(oh)5+	0.6413E-11	-11.1929	-0.0525	0.5683E-11	-11.2455
(uo2)3(oh)7-	0.9110E-12	-12.0405	-0.0483	0.8151E-12	-12.0888
fe(oh)3(aq)	0.7608E-12	-12.1187	0.0000	0.7608E-12	-12.1187
(uo2)2(oh)2++	0.5198E-12	-12.2841	-0.2042	0.3248E-12	-12.4883
fe(oh)2+	0.1336E-12	-12.8741	-0.0525	0.1184E-12	-12.9266
uo2cl+	0.1322E-12	-12.8788	-0.0525	0.1171E-12	-12.9313
uo2(so4)2--	0.0848E-13	-13.0531	-0.1991	0.5595E-13	-13.2522
(uo2)4(oh)7+	0.2126E-13	-13.6725	-0.0525	0.1883E-13	-13.7250
fe(oh)4-	0.6502E-14	-14.1869	-0.0483	0.5818E-14	-14.2352
uo2(oh)4--	0.4742E-14	-14.3241	-0.1991	0.2998E-14	-14.5231
(uo2)3(oh)4++	0.2025E-14	-14.6936	-0.2042	0.1265E-14	-14.8978
(uo2)3(oh)5co2+	0.9136E-15	-15.0392	-0.0525	0.8096E-15	-15.0918
(uo2)2oh+++	0.3789E-16	-16.4215	-0.4498	0.1345E-16	-16.8713
uo2cl2(aq)	0.2721E-16	-16.5653	0.0000	0.2721E-16	-16.5653
feco3+	0.2364E-17	-17.6263	-0.0525	0.2095E-17	-17.6789
feoh++	0.1801E-18	-18.7446	-0.2042	0.1125E-18	-18.9488
h2so4(aq)	0.1290E-18	-18.8893	0.0000	0.1290E-18	-18.8893
hclo(aq)	0.1013E-18	-18.9945	0.0000	0.1013E-18	-18.9945
clo-	0.6254E-19	-19.2039	-0.0483	0.5596E-19	-19.2522
fe+++	0.3325E-21	-21.4781	-0.3998	0.1324E-21	-21.8780
fe++	0.1090E-21	-21.9627	-0.1950	0.6956E-22	-22.1576
uo2+	0.3237E-22	-22.4899	-0.0525	0.2868E-22	-22.5424
fehco3+	0.2767E-22	-22.5580	-0.0525	0.2452E-22	-22.6105
ho2-	0.1811E-22	-22.7421	-0.0483	0.1620E-22	-22.7904
feco3(aq)	0.1126E-22	-22.9485	0.0000	0.1126E-22	-22.9485
feso4+	0.7153E-23	-23.1455	-0.0525	0.6338E-23	-23.1980
feso4(aq)	0.6233E-23	-23.2053	0.0000	0.6233E-23	-23.2053
fecl2+	0.3994E-24	-24.3986	-0.0525	0.3539E-24	-24.4511
fecl+	0.2415E-24	-24.6171	-0.0525	0.2140E-24	-24.6697
fecl++	0.1458E-24	-24.8361	-0.2042	0.9113E-25	-25.0403
clo3-	0.1093E-24	-24.9616	-0.0483	0.9776E-25	-25.0099
fe(so4)2-	0.7739E-25	-25.1113	-0.0483	0.6925E-25	-25.1596
clo4-	0.6205E-25	-25.2072	-0.0500	0.5530E-25	-25.2573
(uo2)11(co3)6(oh)1	0.8484E-27	-27.0714	-0.1991	0.5364E-27	-27.2705
fe(oh)2(aq)	0.7307E-28	-28.1362	0.0000	0.7307E-28	-28.1362
clo2-	0.9782E-29	-29.0096	-0.0483	0.8753E-29	-29.0579
fecl2(aq)	0.4844E-29	-29.3148	0.0000	0.4844E-29	-29.3148
hso5-	0.2536E-29	-29.5959	-0.0483	0.2269E-29	-29.6442
uo2(co3)3(5-)	0.3528E-30	-30.4525	-1.2627	0.1926E-31	-31.7153
fehso4++	0.2031E-30	-30.6923	-0.2042	0.1269E-30	-30.8965
fecl4-	0.9421E-32	-32.0259	-0.0483	0.8430E-32	-32.0742
u(oh)4(aq)	0.1519E-32	-32.8185	0.0000	0.1519E-32	-32.8185
fe(oh)3-	0.7254E-33	-33.1394	-0.0483	0.6490E-33	-33.1878
hclo2(aq)	0.6329E-33	-33.1987	0.0000	0.6329E-33	-33.1987
fecl4--	0.5435E-33	-33.2648	-0.1991	0.3437E-33	-33.4638
uo2clo3+	0.6273E-35	-35.2025	-0.0525	0.5559E-35	-35.2550
fe2(oh)4 2++++	0.2332E-35	-35.6322	-0.7850	0.3827E-36	-36.4172
fe3(oh)4(5+)	0.3268E-41	-41.4857	-1.2055	0.2036E-42	-42.6912
formate	0.3461E-43	-43.4608	-0.0483	0.3097E-43	-43.5091
h2(aq)	0.4913E-44	-44.3006	0.0010	0.4924E-44	-44.3076
s2o8--	0.6070E-45	-45.2168	-0.1991	0.3838E-45	-45.4158
u(co3)4----	0.1113E-45	-45.9537	-0.8063	0.1738E-46	-46.7599
formic acid(aq)	0.8570E-47	-47.0670	0.0000	0.8570E-47	-47.0670
so3--	0.1337E-47	-47.8738	-0.1991	0.8455E-48	-48.0729
hso3-	0.7413E-48	-48.1300	-0.0483	0.6633E-48	-48.1783
u(co3)5(6-)	0.4365E-49	-49.3600	-1.8211	0.6590E-51	-51.1811
uoh+++	0.5234E-50	-50.2812	-0.4498	0.1858E-50	-50.7310
uo2so3(aq)	0.8780E-52	-52.0565	0.0000	0.8780E-52	-52.0565
h2so3(aq)	0.3304E-53	-53.4810	0.0000	0.3304E-53	-53.4810
so2(aq)	0.2351E-53	-53.6288	0.0000	0.2351E-53	-53.6288
u(so4)2(aq)	0.2248E-53	-53.6482	0.0000	0.2248E-53	-53.6482
uso4++	0.9007E-54	-54.0454	-0.2042	0.5628E-54	-54.2496
u++++	0.1912E-56	-56.7186	-0.7850	0.3136E-57	-57.5036
ucl+++	0.2009E-57	-57.6971	-0.4498	0.7130E-58	-58.1469
s2o6--	0.2893E-70	-70.5386	-0.1991	0.1830E-70	-70.7376
u+++	0.1138E-79	-79.9440	-0.4498	0.4038E-80	-80.3938
uo2(so3)2--	0.1683E-98	-98.7739	-0.1991	0.1064E-98	-98.9730
s2o5--	0.1060-100	-100.9748	-0.1991	0.6701-101	-101.1738
methanol(aq)	0.4774-122	-122.3211	0.0000	0.4774-122	-122.3211
s2o4--	0.1055-133	-133.9766	-0.1991	0.6674-134	-134.1756
hs-	0.2351-141	-141.6288	-0.0500	0.2095-141	-141.6788
h2s(aq)	0.9956-142	-142.0019	0.0000	0.9956-142	-142.0019
methane(aq)	0.1744-146	-146.7584	0.0000	0.1744-146	-146.7584
s--	0.7070-147	-147.1040	-0.1991	0.4977-147	-147.3031
s2o3--	0.5488-147	-147.2606	-0.1991	0.3470-147	-147.4596
acetate	0.1510-151	-151.8211	-0.0483	0.1351-151	-151.8695
ch3coo-	0.2534-153	-153.5961	-0.0525	0.2246-153	-153.6487

hs2o3-	0.1959-153	-153.7081	-0.0483	0.1752-153	-153.7564
nach3coo(aq)	0.5489-154	-154.2605	0.0000	0.5489-154	-154.2605
acetic acid(aq)	0.3775-154	-154.4231	0.0000	0.3775-154	-154.4231
uo2s2o3(aq)	0.1066-155	-155.9724	0.0000	0.1066-155	-155.9724
fech3coo++	0.1714-164	-164.7660	-0.2042	0.1071-164	-164.9702
fech3coo+	0.2068-172	-172.6844	-0.0525	0.1832-172	-172.7370
s3o6--	0.2018-177	-177.6952	-0.1991	0.1276-177	-177.8942
ethyne(aq)	0.9642-220	-220.0158	0.0000	0.9642-220	-220.0158
ethanol(aq)	0.1061-232	-232.9742	0.0000	0.1061-232	-232.9742
ethylene(aq)	0.3633-237	-237.4398	0.0000	0.3633-237	-237.4398
s2--	0.2040-251	-251.6903	-0.1991	0.1290-251	-251.8893
ethane(aq)	0.2905-261	-261.5369	0.0000	0.2905-261	-261.5369
propanoate	0.3075-264	-264.5121	-0.0483	0.2752-264	-264.5604
propanoic acid(aq)	0.1025-266	-266.9894	0.0000	0.1025-266	-266.9894
s4o6--	0.1134-268	-268.9455	-0.1991	0.7168-269	-269.1446
acetone(aq)	0.1249-306	-306.9034	0.0000	0.1249-306	-306.9034
fe(ch3coo)2+	0.3312-308	-308.4800	-0.0525	0.2934-308	-308.5325
fe(ch3coo)2(aq)	0.0000E+00	-323.4116	0.0000	0.0000E+00	-323.4116
1-propyne(aq)	0.0000E+00	-328.6679	0.0000	0.0000E+00	-328.6679
1-propanol(aq)	0.0000E+00	-345.6065	0.0000	0.0000E+00	-345.6065
1-propene(aq)	0.0000E+00	-347.9023	0.0000	0.0000E+00	-347.9023
s3--	0.0000E+00	-356.3547	-0.1991	0.0000E+00	-356.5537
propane(aq)	0.0000E+00	-374.5379	0.0000	0.0000E+00	-374.5379
butanoate	0.0000E+00	-377.6867	-0.0483	0.0000E+00	-377.7350
butanoic acid(aq)	0.0000E+00	-380.2227	0.0000	0.0000E+00	-380.2227
s5o6--	0.0000E+00	-389.0763	-0.1991	0.0000E+00	-389.2754
2-butanone(aq)	0.0000E+00	-419.7922	0.0000	0.0000E+00	-419.7922
1-butyne(aq)	0.0000E+00	-441.8352	0.0000	0.0000E+00	-441.8352
fe(ch3coo)3(aq)	0.0000E+00	-454.5947	0.0000	0.0000E+00	-454.5947
1-butanol(aq)	0.0000E+00	-459.4482	0.0000	0.0000E+00	-459.4482
s4--	0.0000E+00	-461.2389	-0.1991	0.0000E+00	-461.4380
1-butene(aq)	0.0000E+00	-461.2530	0.0000	0.0000E+00	-461.2530
n-butane(aq)	0.0000E+00	-487.5945	0.0000	0.0000E+00	-487.5945
pentanoate	0.0000E+00	-490.7515	-0.0483	0.0000E+00	-490.7998
pentanoic acid(aq)	0.0000E+00	-493.2655	0.0000	0.0000E+00	-493.2655
2-pentanone(aq)	0.0000E+00	-533.0989	0.0000	0.0000E+00	-533.0989
1-pentyne(aq)	0.0000E+00	-554.9879	0.0000	0.0000E+00	-554.9879
s5--	0.0000E+00	-566.3431	-0.1991	0.0000E+00	-566.5422
1-pentanol(aq)	0.0000E+00	-571.3108	0.0000	0.0000E+00	-571.3108
1-pentene(aq)	0.0000E+00	-574.4276	0.0000	0.0000E+00	-574.4276
n-pentane(aq)	0.0000E+00	-600.7209	0.0000	0.0000E+00	-600.7209
hexanoate	0.0000E+00	-603.9188	-0.0483	0.0000E+00	-603.9671
hexanoic acid(aq)	0.0000E+00	-606.4181	0.0000	0.0000E+00	-606.4181
2-hexanone(aq)	0.0000E+00	-646.0903	0.0000	0.0000E+00	-646.0903
1-hexyne(aq)	0.0000E+00	-668.2359	0.0000	0.0000E+00	-668.2359
1-hexanol(aq)	0.0000E+00	-685.0866	0.0000	0.0000E+00	-685.0866
1-hexene(aq)	0.0000E+00	-687.4117	0.0000	0.0000E+00	-687.4117
n-hexane(aq)	0.0000E+00	-713.9908	0.0000	0.0000E+00	-713.9908
heptanoate	0.0000E+00	-717.0203	-0.0483	0.0000E+00	-717.0686
heptanoic acid(aq)	0.0000E+00	-719.4243	0.0000	0.0000E+00	-719.4243
2-heptanone(aq)	0.0000E+00	-759.1843	0.0000	0.0000E+00	-759.1843
1-heptyne(aq)	0.0000E+00	-781.5206	0.0000	0.0000E+00	-781.5206
1-heptanol(aq)	0.0000E+00	-799.2362	0.0000	0.0000E+00	-799.2362
1-heptene(aq)	0.0000E+00	-800.5277	0.0000	0.0000E+00	-800.5277
n-heptane(aq)	0.0000E+00	-827.0849	0.0000	0.0000E+00	-827.0849
octanoate	0.0000E+00	-830.1143	-0.0483	0.0000E+00	-830.1626
octanoic acid(aq)	0.0000E+00	-832.2984	0.0000	0.0000E+00	-832.2984
2-octanone(aq)	0.0000E+00	-872.2784	0.0000	0.0000E+00	-872.2784
1-octyne(aq)	0.0000E+00	-894.6733	0.0000	0.0000E+00	-894.6733
1-octanol(aq)	0.0000E+00	-912.1104	0.0000	0.0000E+00	-912.1104
1-octene(aq)	0.0000E+00	-913.7830	0.0000	0.0000E+00	-913.7830
n-octane(aq)	0.0000E+00	-940.2229	0.0000	0.0000E+00	-940.2229

----- major aqueous species contributing to mass balances -----

aqueous species accounting for 99% or more of ca++

species	molal conc	per cent
ca++	0.1721E-02	92.89
caso4(aq)	0.8021E-04	4.33
cahco3+	0.4336E-04	2.34
total		99.56

aqueous species accounting for 99% or more of cl-

species	molal conc	per cent
cl-	0.4994E-02	99.89
total		99.89

aqueous species accounting for 99% or more of fe++

species	molal conc	per cent
fe(oh)3(aq)	0.7608E-12	84.45
fe(oh)2+	0.1336E-12	14.83

total	99.28

aqueous species accounting for 99% or more of hco3-

species	molal conc	per cent
hco3-	0.3511E-02	87.33
co2(aq)	0.3389E-03	8.43
cahco3+	0.4336E-04	1.08
nahco3(aq)	0.2755E-04	0.69
uo2(co3)2--	0.2458E-04	1.22
uo2(co3)3----	0.1235E-04	0.92
total		99.67

aqueous species accounting for 99% or more of na+

species	molal conc	per cent
na+	0.6942E-02	99.17
total		99.17

aqueous species accounting for 99% or more of so4--

species	molal conc	per cent
so4--	0.8941E-03	89.41
caso4(aq)	0.8021E-04	8.02
naso4-	0.2568E-04	2.57
total		100.00

aqueous species accounting for 99% or more of uo2++

species	molal conc	per cent
uo2(co3)2--	0.2458E-04	61.46
uo2(co3)3----	0.1235E-04	30.88
(uo2)2co3(oh)3-	0.1229E-05	6.15
uo2(oh)2(aq)	0.3654E-06	0.91
total		99.40

----- summary of aqueous redox reactions -----

couple	eh, volts	pe-	log fo2	ah, kcal
default	0.786	0.1329E+02	-0.700	18.132
acetic a/hco3-	0.786	0.1329E+02	-0.700	18.132
acetone(/hco3-	0.786	0.1329E+02	-0.700	18.132
clo4- /cl-	*0.786*	*0.1329E+02*	*-0.700*	*18.132*
ethane(a/hco3-	0.786	0.1329E+02	-0.700	18.132
fe+++ /fe++	0.786	0.1329E+02	-0.700	18.132
h2(aq) /h2o	0.786	0.1329E+02	-0.700	18.132
hs- /so4--	0.786	0.1329E+02	-0.700	18.132
methane(/hco3-	0.786	0.1329E+02	-0.700	18.132
methanol/hco3-	0.786	0.1329E+02	-0.700	18.132
o2(aq) /h2o	0.786	0.1329E+02	-0.700	18.132
so3-- /so4--	0.786	0.1329E+02	-0.700	18.132
u+++ /uo2++	0.786	0.1329E+02	-0.700	18.132
u++++ /uo2++	0.786	0.1329E+02	-0.700	18.132
uo2+ /uo2++	0.786	0.1329E+02	-0.700	18.132

----- summary of aqueous non-equilibrium non-redox reactions -----

couple	affinity, kcal
none	

----- summary of stoichiometric mineral saturation states -----

(minerals with affinities .lt. -10 kcal are not listed)

mineral	log q/k	aff, kcal	state	mineral	log q/k	aff, kcal	state
anhydrite	-1.900	-2.593		aragonite	-0.144	-0.197	satd
bassanite	-2.545	-3.473		calcite	0.000	0.000	satd
caso4:0.5h2o(beta)	-2.714	-3.702		cauo4	-0.499	-0.601	
fe(oh)3	-5.602	-7.642		goethite	-0.480	-0.655	
gypsum	-1.725	-2.353		halite	-6.148	-8.388	
hematite	*0.000*	*0.000*	*satd*	*ice*	*-0.139*	*-0.189*	*satd*
mirabilite	-6.531	-8.911		monohydrocalcite	-0.834	-1.138	

```
na2u2o7                 -4.621     -6.305      nahcolite               -4.601     -6.277
rutherfordine           -1.823     -2.487      schoepite               -0.948     -1.293
schoepite-dehy(.39      -2.839     -1.873      schoe-  .e-dehy(.64     -2.321     -3.166
schoepite-dehy(.85      -1.212     -1.653      schoe,ite-dehy(.9)      -1.131     -1.544
schoepite-dehy(1.0      -1.218     -1.661      uo2(oh)2(beta)          -1.060     -1.447
uo3(alpha)              -4.754     -6.486      uo3(beta)               -4.425     -6.037
uo3(gamma)              -3.823     -5.215

    4 approx. saturated pure minerals
    0 approx. saturated end-members of specified solid solutions
    0 saturated end-members of hypothetical solid solutions


    0 supersaturated pure minerals
    0 supersatd. end-members of specified solid solutions
    0 supersatd. hypothetical solid solution phases



                  ----- summary of gases -----

      gas                         fugacity    log fugacity

  c(g)                        0.101555-187    -187.99330
  ca(g)                       0.238726-151    -151.62210
  ch4(g)                      0.123543-143    -143.90818
  cl2(g)                      0.751235E-24     -24.12422
  co(g)                       0.193508E-46     -46.71330
  co2(g)                      0.100000E-01      -2.00000
  h2(g)                       0.627133E-41     -41.20264
  h2o(g)                      0.259693E-01      -1.58554
  h2s(g)                      0.969907-141    -141.01372
  hcl(g)                      0.105959E-15     -15.97078
  na(g)                       0.134780E-74     -74.87037
  o2(g)                       0.199526E+00      -0.70000
  s2(g)                       0.532224-225    -225.27391
  so2(g)                      0.158936E-53     -53.79878
  u(g)                        0.260406-289    -289.58435
  u2cl10(g)                   0.140826-209    -209.85132
  u2cl8(g)                    0.511217-219    -219.29139
  ucl(g)                      0.310198-247    -247.50836
  ucl2(g)                     0.298213-198    -198.52547
  ucl3(g)                     0.320712-150    -150.49389
  ucl4(g)                     0.938216-114    -114.02770
  ucl5(g)                     0.323587-115    -115.49001
  ucl6(g)                     0.517701-113    -113.28592
  uo2cl2(g)                   0.480758E-62     -62.31807

----- end of output -----
--- pickup file has been successfully written ---

--- reading the input file ---

--- no further input found ---

        start time =  09:27       3Dec91
          end time =  09:27       3Dec91

        user time =       6.230
         cpu time =       1.240
normal exit
```

7.5. Calculating the Composition of a Custom *pH* Buffer: An Example

This short example illustrates the use of EQ3NR to calculate the composition of a custom *pH* buffer solution. Such buffers are highly useful in laboratory experiments, for example in determining the *pH* dependence of mineral dissolution kinetics (e.g., Knauss and Wolery, 1986). The model is defined by choosing the desired *pH* at the given temperature and the concentration of the buffering component. Electrical balancing is used to determine how much acid or base to include in the buffer recipe. The details are explained in the title on the **input** file, which is presented in both formats. This example was computed using the **com** data file. The activity coefficients were computed from the B-dot equation.

In this case, the buffer recipe is to consist of 0.05 molal boric and some unknown concentration of *NaOH*. The p*H* is to be 8.00 at 70°C. The desired concentration of *NaOH* is determined by

electrical balancing on the sodium ion. The calculated concentration of Na^+ is 0.005691 molal. Hence the buffer should contain 0.005691 molal of *NaOH*.

By doing very similar calculations with electrical balancing on the hydrogen ion, it is possible to confirm consistency between code calculations and standard *pH* buffer recipes. The *pH* of such solutions at other temperatures can then be found simply by changing the temperature on the data file. Some examples of this are given by Knauss and Wolery (1986).

When doing calculations involving buffers intended for use in the laboratory, it is always wise to check the saturation index results to ensure that the desired buffer composition is not supersaturated with respect to some solid phase. If it is, the buffer solution may be impossible to make up, or if it can be made up, it may not be stable owing to eventual precipitation of the supersaturated phase. The buffer solution may also be hard to make up if the solid form of the buffer component (such as boric acid) is undersaturated, but fairly close to saturation.

The EQ3NR **input** file (**custombuf.3i**),custom pH buffer test case ("W" format):

```
EQ3NR input file name= custombuf.3i
Description= 'Custom borate pH buffer, pH 8.00 at 70 C'
Version number= 3245   Stage number= 01
Created 06/08/90     Creator= T.J. Wolery
Revised 06/08/90     Revisor= T.J. Wolery

  Compute the amount of NaOH required for the custom borate pH
buffer solution-

    pH 8.00 (at 70 C) buffer: 0.05 m H3BO3 + x m NaOH

This buffer is a spin-off of buffers used by Knauss and Wolery
(1986) in mineral dissolution rate experiments. The H3BO3
concentration has been dropped by an order of magnitude to reduce
interference with the analysis of alkali cations (such as K+) in
the leachate. The molality of Na+ is adjusted so as to find the
molality of NaOH (x) required to obtain a buffer pH of 8.00 at
70 C, the intended temperature of the experiments. The adjusted
molality is given as the "final" value in "moles/kg."

  Purpose: to test electrical balancing on a solute other than H+
and to demonstrate the code's abilty to calculate the composition of
a custom pH buffer.

                        References

Knauss, K. G., and Wolery, T. J., 1986, Dependence of albite
  dissolution kinetics on pH and time at 25 C and 70 C: Geochimica
  et Cosmochimica Acta, v. 50, p. 2481-2497.

endit.
      tempc=          70.
        rho=     1.00000     tdspkg=          0.      tdspl=          0.
         fep=      -0.700     uredox=
      tolbt=          0.      toldl=          0.     tolsat=          0.
     itermx=  0
*                1    2    3    4    5    6    7    8    9   10
  iopt1-10=      0    0    0    0    0    0    0    0    0    0
  iopg1-10=      0    0    0    0    0    0    0    0    0    0
  iopr1-10=      0    0    0    0    0    0    0    0    0    0
 iopr11-20=      0    0    0    0    0    0    0    0    0    0
  iodb1-10=      0    0    0    0    0    0    0    0    0    0
      ucbal= na+
      nxmod=  0
*
data file master species= h+
   switch with species=
   jflag= 16    csp= -8.00
data file master species= na+
   switch with species=
   jflag=  0    csp= 0.004
data file master species= b(oh)3(aq)
   switch with species=
   jflag=  0    csp= 0.05
endit.
```

The EQ3NR input file (custombut.3i),custom pH buffer test case ("D" format):

```
EQ3NR input file name= custombuf.3i
Description= "Custom borate pH buffer, pH 8.00 at 70 C"
Version number= 3245    Stage number= 01
Created 06/08/90     Creator= T.J. Wolery
Revised 06/08/90     Revisor= T.J. Wolery

  Compute the amount of NaOH required for the custom borate pH
buffer solution-

    pH 8.00 (at 70 C) buffer: 0.05 m H3BO3 + x m NaOH

This buffer is a spin-off of buffers used by Knauss and Wolery
(1986) in mineral dissolution rate experiments. The H3BO3
concentration has been dropped by an order of magnitude to reduce
interference with the analysis of alkali cations (such as K+) in
the leachate. The molality of Na+ is adjusted so as to find the
molality of NaOH (x) required to obtain a buffer pH of 8.00 at
70 C, the intended temperature of the experiments. The adjusted
molality is given as the "final" value in "moles/kg."

  Purpose: to test electrical balancing on a solute other than H+
and to demonstrate the code's ability to calculate the composition of
a custom pH buffer.

                          References

Knauss, K. G., and Wolery, T. J., 1986, Dependence of albite
  dissolution kinetics on pH and time at 25 C and 70 C: Geochimica
  et Cosmochimica Acta, v. 50, p. 2481-2497.

-------------------------------------------------------------------
Temperature (C)      | 70.00      |Density(gm/cm3)|  1.00000
-------------------------------------------------------------------
Total Dissolved Salts |            | mg/kg | mg/l  |*not used
-------------------------------------------------------------------
Electrical Balancing on |Na+       | code selects  | not performed
-------------------------------------------------------------------
SPECIES  | BASIS SWITCH/CONSTRAINT | CONCENTRATION | UNITS OR TYPE
-------------------------------------------------------------------
redox    |                         | -.7000        |logfo2
h+       |                         | 8.0000        |ph
na+      |                         | 0.40000E-02   |molality
b(oh)3(aq)|                        | 0.50000E-01   |molality
-------------------------------------------------------------------
Input Solid Solutions
-------------------------------------------------------------------
 none    |                         |               |
-------------------------------------------------------------------
SUPPRESSED SPECIES   (suppress,replace,augmentk,augmentg)    value
-------------------------------------------------------------------
 none                |            |
-------------------------------------------------------------------
OPTIONS
-------------------------------------------------------------------
 - SOLID SOLUTIONS -
   * ignore solid solutions
     process hypothetical solid solutions
     process input and hypothetical solid solutions
 - LOADING OF SPECIES INTO MEMORY -
   * does nothing
     lists species loaded into memory
 - ECHO DATABASE INFORMATION -
   * does nothing
     lists all reactions
     lists reactions and log K values
     lists reactions, log K values and polynomial coef.
 - LIST OF AQUEOUS SPECIES (ordering) -
   * in order of decreasing concentration
     in same order as input file
 - LIST OF AQUEOUS SPECIES (concentration limit) -
   * all species
     only species > 10**-20 molal
     only species > 10**-12 molal
     not printed
 - LIST OF AQUEOUS SPECIES (by element) -
   * print major species
     print all species
     don't print
 - MINERAL SATURATION STATES -
   * print if affinity > -10 kcals
     print all
     don't print
 - pH SCALE CONVENTION -
   * modified NBS
```

```
    internal
    rational
- ACTIVITY COEFFICIENT OPTIONS -
  * use B-dot equation
    Davies' equation
    Pitzer's equations
- AUTO BASIS SWITCHING -
  * off
    on
- PITZER DATABASE INFORMATION -
  * print only warnings
    print species in model and number of Pitzer coefficients
    print species in model and names of Pitzer coefficients
- PICKUP FILE -
  * write pickup file
    don't write pickup file
- LIST MEAN IONIC PROPERTIES -
  * don't print
    print
- LIST AQUEOUS SPECIES, ION SIZES, AND HYDRATION NUMBERS -
  * print
    don't print
- CONVERGENCE CRITERIA -
  * test both residual functions and correction terms
    test only residual functions
-------------------------------------------------------------------
DEBUGGING SWITCHES (0-off, 1,2-on, default is off)
-------------------------------------------------------------------
0  generic debugging information                                  1
0  print details of pre-Newton-Raphson iteration                  2
0  print details of Newton-Raphson iteration
0  print details of stoichiometric factors                        2
0  print details of stoichiometric factors calculation
0  write reactions on RLIST
0  list stoichiometric concentrations of master species
0  request iteration variables to be killed
-------------------------------------------------------------------
DEVELOPMENT OPTIONS  (used for code development)
-------------------------------------------------------------------
 none
-------------------------------------------------------------------
TOLERANCES                  (desired values)       (defaults)
-------------------------------------------------------------------
      residual functions                          1.e-10
        correction terms                          1.e-10
        saturation state                          0.5
number of N-R iterations                          30
-------------------------------------------------------------------
```

The EQ3NR output file (**custombuf.3o**), custom pH buffer test case (beginning with the message announcing the end of Newton-Raphson iteration):

```
                    .
                    .
                    .
          (Material deleted)
                    .
                    .
                    .
Hybrid newton-raphson iteration converged in   5 steps.


                  ----- Summary of the Aqueous Phase -----

          ----- Elemental composition of the aqueous phase -----

    element       mg/l          mg/kg          moles/kg

      o       0.89042E+06    0.89042E+06    0.5565315583E+02
      b         540.55         540.55       0.4999999993E-01
      h       0.11203E+06    0.11203E+06    0.1111499637E+03
      na        130.83         130.83       0.5690579408E-02

          ----- Elemental composition as strict basis species -----

    species            mg/l          mg/kg          moles/kg

  h2o             0.10026E+07    0.10026E+07    0.5565315583E+02
  b(oh)3(aq)        3091.7         3091.7       0.4999999993E-01
  h+              0.11203E+06    0.11203E+06    0.1111499637E+03
```

na+	130.83	130.83	0.5690579408E-02

--- equivalent composition of the aqueous phase (cto balances) ---

original basis		existing basis	
species	moles/kg h2o	species	moles/kg h2o
h2o	0.5565315583E+02	h2o	0.5565315583E+02
b(oh)3(aq)	0.4999999993E-01	b(oh)3(aq)	0.4999999993E-01
h+	0.1111499637E+03	h+	0.1111499637E+03
na+	0.5690579408E-02	na+	0.5690579408E-02

single ion activities and activity coefficients are here defined
with respect to the modified nbs ph scale

	ph	eh	pe
modified nbs ph scale	8.0000	0.6660	9.7819E+00
rational ph scale	7.9649	0.6684	9.8169E+00

activity of water = 0.99900
log activity of water = -0.00043

true osmotic coefficient= 0.99439
stoichiometric osmotic coefficient= 0.99396

sum of true molalities= 0.0558475100840
sum of stoichiometric molalities= 0.0558718616625

true ionic strength= 0.0056661824285
stoichiometric ionic strength= 0.0028537623461

----- electrical balance totals -----

	equiv/kg
sigma(mz) cations =	0.5666182429E-02
sigma(mz) anions =	-0.5666182358E-02
total charge =	0.1133236479E-01
mean charge =	0.5666182393E-02
charge imbalance =	0.7043948671E-10

total charge = sigma(mz) cations + abs (sigma(mz) anions)
mean charge = 1/2 total charge

the electrical imbalance is

0.622E-06 per cent of the total charge
0.124E-05 per cent of the mean charge
0.124E-05 per cent of sigma(mz) cations
0.124E-05 per cent of abs (sigma(mz) anions)

--- electrical balancing on na+ ---

	mg/l	mg/kg	moles/kg
input	91.959	91.959	0.4000000000E-02
final	130.83	130.83	0.5690579408E-02
adj	38.866	38.866	0.1690579408E-02

----- activity ratios of ions -----

log (act(b(oh)3(aq)	)	= -1.3533
log (act(na+	) / act(h+)xx 1)	= 5.7143
log (act(h2(aq)	)	= -37.7891
log (act(o2(aq)	)	= -3.7841
log (act(oh-	) x act(h+)xx 1)	= -12.8090

----- distribution of aqueous species -----

species	molal conc	log conc	log g	activity	log act
b(oh)3(aq)	0.4433E-01	-1.3533	0.0000	0.4433E-01	-1.3533
na+	0.5666E-02	-2.2467	-0.0390	0.5180E-02	-2.2857
bo2-	0.5649E-02	-2.2480	-0.0369	0.5189E-02	-2.2849
o2(aq)	0.1643E-03	-3.7843	0.0001	0.1644E-03	-3.7841
nab(oh)4(aq)	0.2435E-04	-4.6135	0.0000	0.2435E-04	-4.6135
oh-	0.1693E-04	-4.7712	-0.0378	0.1552E-04	-4.8090
naoh(aq)	0.5624E-07	-7.2499	0.0000	0.5624E-07	-7.2499
h+	0.1084E-07	-7.9649	-0.0351	0.1000E-07	-8.0000
b2o(oh)5-	0.6786E-12	-12.1684	-0.0369	0.6234E-12	-12.2052
b3o3(oh)4-	0.3014E-15	-15.5208	-0.0369	0.2769E-15	-15.5577

```
  ho2-                  0.5225E-19  -19.2818    -0.0369   0.4801E-19  -19.3187
  b4o5(oh)4--           0.1518E-27  -27.8188    -0.1504   0.1074E-27  -27.9692
  h2(aq)                0.1625E-37  -37.7892     0.0001   0.1625E-37  -37.7891
  bh4-                  0.1043-190 -190.9817    -0.0369   0.9581-191 -191.0186

          ----- major aqueous species contributing to mass balances -----

aqueous species accounting for 99% or more of b(oh)3(aq)

     species             molal conc     per cent

  b(oh)3(aq)             0.4433E-01     88.65
  bo2-                   0.5649E-02     11.30
- - - - - - - - - - - - - - - - - - - - - - - -
  total                                 99.95

aqueous species accounting for 99% or more of na+

     species             molal conc     per cent

  na+                    0.5666E-02     99.57
- - - - - - - - - - - - - - - - - - - - - - - -
  total                                 99.57


          ----- summary of aqueous redox reactions -----

     couple          eh, volts       pe-         log fo2      ah, kcal

  default               0.666     0.9782E+01      -0.700       15.359
  h2(aq)  /h2o          0.666     0.9782E+01      -0.700       15.359
  o2(aq)  /h2o          0.666     0.9782E+01      -0.700       15.359

          ----- summary of aqueous non-equilibrium non-redox reactions -----

     couple        affinity, kcal

     none

          ----- summary of stoichiometric mineral saturation states -----

          (minerals with affinities .lt. -10 kcal are not listed)

  mineral             log q/k    aff, kcal  state          mineral             log q/k    aff, kcal  state

boric acid             -1.644      -2.582                 ice                  -0.330      -0.519

     0 approx. saturated pure minerals
     0 approx. saturated end-members of specified solid solutions
     0 saturated end-members of hypothetical solid solutions


     0 supersaturated pure minerals
     0 supersatd. end-members of specified solid solutions
     0 supersatd. hypothetical solid solution phases



                    ----- summary of gases -----

     gas                        fugacity     log fugacity

  b(g)                     0.217492-171    -171.66256
  h2(g)                    0.221420E-34     -34.65478
  h2o(g)                   0.255598E+00      -0.59244
  na(g)                    0.942117E-63     -63.02580
  o2(g)                    0.199526E+00      -0.70000

----- end of output -----
--- pickup file has been successfully written ---

--- reading the input file ---

--- no further input found ---

         start time = 09:25       3Dec91
           end time = 09:25       3Dec91

         user time =       3.490
          cpu time =       0.870
normal exit
```

7.6. Computing Oxygen Fugacity from Mineral Equilibria: An Example

In this example, the oxygen fugacity of a hydrothermal solution at 250°C is estimated by assuming equilibrium between the aqueous solution, hematite, and magnetite. Note the use of the option **ioptl** = -3, which opens up a species input block for $O_{2(g)}$. Note that equilibrium with magnetite is the constraint assigned to Fe^{2+}. Equilibrium with hematite is assigned to $O_{2(g)}$. This problem also determines the concentration of dissolved iron. This test case was adapted from one given by Henley et al. (1984), which also involves equilibrium with quartz, albite, K-feldspar, and muscovite, and electrical balancing to determine the *pH*. The results shown here were obtained using the **com** data file. The activity coefficients were computed from the B-dot equation. The **input** file is presented in both formats. The **output** file is presented beginning with the message announcing the end of Newton-Raphson iteration.

Equilibrium between hematite ($Fe_2O_{3(c)}$) and magnetite ($Fe_3O_{4(c)}$) fixes the oxygen fugacity because the reaction between the two can be written as:

$$6Fe_2O_{3(c)} = 4Fe_3O_{4(c)} + O_{2(g)} \tag{223}$$

Because the thermodynamic activities of the two minerals are each unity, the corresponding mass action equation reduces to:

$$logf_{O_2} = logK \tag{224}$$

where K is the equilibrium constant for the stated reaction.

Note that on the **output** file that the calculated saturation indices for hematite and magnetite are indeed zero. The *pH* is -7.1045, the log oxygen fugacity is -35.301, and the concentration of dissolved iron is a very low 0.624×10^{-9} molal.

The EQ3NR **input** file (**fo2mineq.3i**), the oxygen fugacity from mineral equilibria test case ("W" format):

```
EQ3NR input file name= fo2mineq.3i
Description= "Compute fO2 from hematite-magnetite equilibria"
Version number= 3245   Stage number= 01
Created 06/08/90     Creator= T.J. Wolery
Revised 06/08/90     Revisor= T.J. Wolery

  Compute the oxygen fugacity assuming equilibrium with hematite
and magnetite. This is an extension of the problem in henleyph.3i,
taken from Henley et al. (1984, p. 96-97). In the original problem,
the pH was to be calculated from electrical balancing. The initial
value was 6.0, the adjusted value about 7.1. An initial value of
7.1 is used here.

  Purpose: to test the computation of fO2 from specified mineral
equilibria

                        References

Henley, R. W., Truesdell, A. H., Barton, P. B., Jr., and Whitney,
  J. A., 1984, Fluid-Mineral Equilibria in Hydrothermal Systems:
  Reviews in Economic Geology, v. 1, Society of Economic Geologists,
  The Economic Geology Publishing Company, El Paso, Texas.

endit.
     tempc=          250.
       rho=            1.    tdspkg=            0.     tdspl=            0.
       fep=            0.    uredox=
```

```
     tolbt=          0.     toldl=          0.     tolsat=          0.
    itermx=  0
*          1    2    3    4    5    6    7    8    9   10
  iopt1-10=   -3    0    0    0    0    0    0    0    0    0
  iopg1-10=    0    0    0    0    0    0    0    0    0    0
  iopr1-10=    0    0    0    0    0    0    0    0    0    0
 iopr11-20=    0    0    0    0    0    0    0    0    0    0
  iodb1-10=    0    0    0    0    0    0    0    0    0    0
     uebal= h+
     hxmod=  0
*
data file master species= na+
   switch with species=
   jflag= 19   csp= 0.
 mineral= albite
data file master species= k+
   switch with species=
   jflag= 19   csp= 0.
 mineral= muscovite
data file master species= cl-
   switch with species=
   jflag=  0   csp= 0.01
data file master species= sio2(aq)
   switch with species=
   jflag= 19   csp= 0.
 mineral= quartz
data file master species= al+++
   switch with species=
   jflag= 19   csp= 0.
 mineral= k-feldspar
data file master species= h+
   switch with species=
   jflag= 16   csp= -7.1
data file master species= fe++
   switch with species=
   jflag= 19   csp= 0.
 mineral= magnetite
data file master species= o2(g)
   switch with species=
   jflag= 19   csp= 0.
 mineral= hematite
endit.
```

The EQ3NR input file (**fo2mineq.3i**), the oxygen fugacity from mineral equilibria test case ("D" format):

```
-------------------------------------------------------------------------------
EQ3NR input file name= fo2mineq.3i
Description= "Compute fO2 from hematite-magnetite equilibria"
Version number= 3245   Stage number= 01
Created 06/08/90    Creator= T.J. Wolery
Revised 06/08/90    Revisor= T.J. Wolery

  Compute the oxygen fugacity assuming equilibrium with hematite
and magnetite. This is an extension of the problem in henleyph.3i,
taken from Henley et al. (1984, p. 96-97). In the original problem,
the pH was to be calculated from electrical balancing. The initial
value was 6.0, the adjusted value about 7.1. An initial value of
7.1 is used here.

  Purpose: to test the computation of fO2 from specified mineral
equilibria.

                          References

Henley, R. W., Truesdell, A. H., Barton, P. B., Jr., and Whitney,
  J. A., 1984, Fluid-Mineral Equilibria in Hydrothermal Systems:
  Reviews in Economic Geology, v. 1, Society of Economic Geologists,
  The Economic Geology Publishing Company, El Paso, Texas.

-------------------------------------------------------------------------------
Temperature (C)          |250.00        |Density(gm/cm3)|  1.00000
-------------------------------------------------------------------------------
Total Dissolved Salts    |              | mg/kg | mg/l  |*not used
-------------------------------------------------------------------------------
Electrical Balancing on |h+             | code selects  | not performed
-------------------------------------------------------------------------------
SPECIES    | BASIS SWITCH/CONSTRAINT | CONCENTRATION| UNITS OR TYPE
-------------------------------------------------------------------------------
na+        |albite                   |          0.  |mineral
k+         |muscovite                |          0.  |mineral
cl-        |                         |0.10000E-01   |molality
sio2(aq)   |quartz                   |          0.  |mineral
al+++      |k-feldspar               |          0.  |mineral
h+         |                         |     7.1000   |ph
fe++       |magnetite                |          0.  |mineral
```

redox o2(g)	hematite	0.	redox couple mineral

Input Solid Solutions

none

SUPPRESSED SPECIES (suppress,replace,augmentk,augmentg) value

none

OPTIONS

- SOLID SOLUTIONS -
  - * ignore solid solutions
    process hypothetical solid solutions
    process input and hypothetical solid solutions
- LOADING OF SPECIES INTO MEMORY -
  - * does nothing
    lists species loaded into memory
- ECHO DATABASE INFORMATION -
  - * does nothing
    lists all reactions
    lists reactions and log K values
    lists reactions, log K values and polynomial coef.
- LIST OF AQUEOUS SPECIES (Ordering) -
  - * in order of decreasing concentration
    in same order as input file
- LIST OF AQUEOUS SPECIES (concentration *limit*) -
  - * all species
    only species > 10**-20 molal
    only species > 10**-12 molal
    not printed
- LIST OF AQUEOUS SPECIES (by element) -
  - * print major species
    print all species
    don't print
- MINERAL SATURATION STATES -
  - * print if affinity > -10 kcals
    print all
    don't print
- pH SCALE CONVENTION -
  - * modified NBS
    internal
    rational
- ACTIVITY COEFFICIENT OPTIONS -
  - * use B-dot equation
    Davies' equation
    Pitzer's equations
- AUTO BASIS SWITCHING -
  - * off
    on
- PITZER DATABASE INFORMATION -
  - * print only warnings
    print species in model and number of Pitzer coefficients
    print species in model and names of Pitzer coefficients
- PICKUP FILE -
  - * write pickup file
    don't write pickup file
- LIST MEAN IONIC PROPERTIES -
  - * don't print
    print
- LIST AQUEOUS SPECIES, ION SIZES, AND HYDRATION NUMBERS -
  - * print
    don't print
- CONVERGENCE CRITERIA -
  - * test both residual functions and correction terms
    test only residual functions

DEBUGGING SWITCHES (0-off, 1,2-on, default is off)

0	generic debugging information	2
0	print details of pre-Newton-Raphson iteration	2
0	print details of Newton-Raphson iteration	
0	print details of stoichiometric factors	2
0	print details of stoichiometric factors calculation	
0	write reactions on RLIST	
0	list stoichiometric concentrations of master species	
0	request iteration variables to be killed	

DEVELOPMENT OPTIONS (used for code development)

none

TOLERANCES	(desired values)	(defaults)
residual functions		1.e-10
correction terms		1.e-10
saturation state		0.5
number of N-R iterations		30

The EQ3NR **output** file (**fo2mineq.3o**), the oxygen fugacity from mineral equilibria test case (beginning with the message announcing the end of Newton-Raphson iteration):

```
.
.
.
(Material deleted)
.
.
.
Hybrid newton-raphson iteration converged in   4 steps.


                  ----- Summary of the Aqueous Phase -----

       ----- Elemental composition of the aqueous phase -----

    element       mg/l          mg/kg          moles/kg

      o       0.88831E+06    0.88831E+06    0.5552162122E+02
      al       2.1526         2.1526        0.7978017881E-04
      cl        354.53         354.53       0.1000000000E-01
      fe      0.36427E-03    0.36427E-03    0.6522560372E-08
      h       0.11190E+06    0.11190E+06    0.1110174095E+03
      k        26.877         26.877        0.6874113956E-03
      na       220.69         220.69        0.9599276272E-02
      si       177.78         177.78        0.6329961991E-02

       ----- elemental composition as strict basis species -----

    species          mg/l          mg/kg          moles/kg

  h2o           0.10002E+07    0.10002E+07    0.5552162122E+02
  al+++          2.1526         2.1526        0.7978017881E-04
  cl-            354.53         354.53        0.1000000000E-01
  fe++          0.36427E-03    0.36427E-03    0.6522560372E-08
  h+            0.11190E+06    0.11190E+06    0.1110174095E+03
  k+             26.877         26.877        0.6874113956E-03
  na+            220.69         220.69        0.9599276272E-02
  sio2(aq)       380.33         380.33        0.6329961991E-02

       --- equivalent composition of the aqueous phase (cte balances) ---

        original basis                  existing basis

  species          moles/kg h2o     species          moles/kg h2o

h2o             0.5552162122E+02   h2o             0.5552162122E+02
al+++           0.7978017881E-04   al+++           0.7978017881E-04
cl-             0.1000000000E-01   cl-             0.1000000000E-01
fe++            0.6522560372E-08   fe++            0.6522560372E-08
h+              0.1110174095E+03   h+              0.1110174095E+03
k+              0.6874113956E-03   k+              0.6874113956E-03
na+             0.9599276272E-02   na+             0.9599276272E-02
sio2(aq)        0.6329961991E-02   sio2(aq)        0.6329961991E-02

single ion activities and activity coefficients are here defined
with respect to the modified nbs ph scale


                                    ph          eh          pe

  modified nbs ph scale           7.1043      -0.4409    -4.2479E+00
  rational ph scale               7.0303      -0.4133    -4.1739E+00

phcl =      9.1965


                      activity of water =  0.99954
                   log activity of water =   -0.00020

              true osmotic coefficient=     0.95501
      stoichiometric osmotic coefficient=   0.94765

              sum of true molalities=     0.0266011622504
      sum of stoichiometric molalities=   0.0268077718714

              true ionic strength=     0.0100799754658
      stoichiometric ionic strength=   0.0105548045192

       ----- electrical balance totals -----
```

equiv/kg

sigma(mz) cations =	0.1007996578E-01
sigma(mz) anions =	-0.1007996578E-01
total charge =	0.2015993155E-01
mean charge =	0.1007996578E-01
charge imbalance =	0.1393052340E-12

total charge = sigma(mz) cations + abs (sigma(mz) anions)
mean charge = 1/2 total charge

the electrical imbalance is

0.691E-09 per cent of the total charge
0.138E-08 per cent of the mean charge
0.138E-08 per cent of sigma(mz) cations
0.138E-08 per cent of abs (sigma(mz) anions)

--- electrical balancing on h+ ---

log activity

input	-7.1000
final	-7.1043
adj	-0.42640E-02

----- activity ratios of ions -----

log (act(al+++	) / act(h+)xx 3) =	-1.8029
log (act(cl-	) x act(h+)xx 1) =	-9.1965
log (act(fe++	) / act(h+)xx 2) =	3.6967
log (act(k+	) / act(h+)xx 1) =	3.8484
log (act(na+	) / act(h+)xx 1) =	4.9916
log (act(sio2(aq)	) =	-2.2057
log (act(clo4-	) x act(h+)xx 1) =	-97.9770
log (act(fe+++	) / act(h+)xx 3) =	-5.4517
log (act(h2(aq)	) =	-5.1908
log (act(o2(aq)	) =	-37.9494
log (act(oh-	) x act(h+)xx 1) =	-11.1677

----- distribution of aqueous species -----

species	molal conc	log conc	log g	activity	log act
cl-	0.9806E-02	-2.0085	-0.0837	0.8087E-02	-2.0922
na+	0.9400E-02	-2.0269	-0.0858	0.7715E-02	-2.1127
sio2(aq)	0.6227E-02	-2.2057	0.0000	0.6227E-02	-2.2057
k+	0.6804E-03	-3.1673	-0.0886	0.5548E-03	-3.2559
nacl(aq)	0.1875E-03	-3.7271	0.0000	0.1875E-03	-3.7271
oh-	0.1048E-03	-3.9797	-0.0837	0.8641E-04	-4.0634
hsio3-	0.9054E-04	-4.0432	-0.0809	0.7516E-04	-4.1240
al(oh)4-	0.7913E-04	-4.1017	-0.0809	0.6569E-04	-4.1825
nahsio3(aq)	0.1201E-04	-4.9204	0.0000	0.1201E-04	-4.9204
kcl(aq)	0.7040E-05	-5.1524	0.0000	0.7040E-05	-5.1524
h2(aq)	0.6462E-05	-5.1896	-0.0017	0.6445E-05	-5.1908
al(oh)3(aq)	0.6518E-06	-6.1859	0.0000	0.6518E-06	-6.1859
naoh(aq)	0.2077E-06	-6.6826	0.0000	0.2077E-06	-6.6826
nah3sio4(aq)	0.9970E-07	-7.0013	0.0000	0.9970E-07	-7.0013
h+	0.9326E-07	-7.0303	-0.0740	0.7866E-07	-7.1043
h2sio4--	0.9626E-08	-8.0166	-0.3294	0.4508E-08	-8.3460
fe(oh)2(aq)	0.6443E-08	-8.1909	0.0000	0.6443E-08	-8.1909
hcl(aq)	0.1966E-08	-8.7065	0.0000	0.1966E-08	-8.7065
fe++	0.6408E-10	-10.1933	-0.3186	0.3077E-10	-10.5118
al(oh)2+	0.2741E-10	-10.5621	-0.0858	0.2250E-10	-10.6479
fecl+	0.1470E-10	-10.8326	-0.0858	0.1207E-10	-10.9184
fe(oh)2+	0.5369E-12	-12.2701	-0.0858	0.4406E-12	-12.3559
fe(oh)4-	0.2576E-13	-13.5890	-0.0809	0.2139E-13	-13.6699
fecl2(aq)	0.1678E-13	-13.7753	0.0000	0.1678E-13	-13.7753
fecl4--	0.3537E-15	-15.4514	-0.3294	0.1657E-15	-15.7808
aloh++	0.1452E-16	-16.8380	-0.3338	0.6732E-17	-17.1718
feoh++	0.2257E-19	-19.6446	-0.3338	0.1051E-19	-19.9784
al+++	0.3452E-22	-22.4619	-0.6537	0.7661E-23	-23.1157
fecl2+	0.2060E-22	-22.6860	-0.0858	0.1691E-22	-22.7718
fecl++	0.2775E-24	-24.5568	-0.3338	0.1287E-24	-24.8906
fe+++	0.7750E-26	-26.1107	-0.6537	0.1720E-26	-26.7644
fecl4-	0.1891E-27	-27.7232	-0.0809	0.1570E-27	-27.8041
al2(oh)2++++	0.2506E-31	-31.6011	-1.2852	0.1299E-32	-32.8863
clo-	0.7262E-32	-32.1390	-0.0809	0.6028E-32	-32.2198
ho2-	0.6463E-32	-32.1896	-0.0809	0.5365E-32	-32.2704
o2(aq)	0.1127E-37	-37.9482	-0.0012	0.1124E-37	-37.9494
fe2(oh)2++++	0.2856E-40	-40.5442	-1.2852	0.1481E-41	-41.8294
fe3(oh)4(5+)	0.2210E-52	-52.6540	-1.9745	0.2352E-54	-54.6285
clo2-	0.1500E-55	-55.8239	-0.0809	0.1245E-55	-55.9047
clo3-	0.2615E-71	-71.5826	-0.0809	0.2171E-71	-71.6634
clo4-	0.1625E-90	-90.7890	-0.0837	0.1340E-90	-90.8727

----- major aqueous species contributing to mass balances -----

aqueous species accounting for 99% or more of al+++

species	molal conc	per cent
al(oh)4-	0.7913E-04	99.18
total		99.18

aqueous species accounting for 99% or more of cl-

species	molal conc	per cent
cl-	0.9806E-02	98.06
nacl(aq)	0.1875E-03	1.87
total		99.93

aqueous species accounting for 99% or more of fe++

species	molal conc	per cent
fe(oh)2(aq)	0.6443E-08	98.78
fe++	0.6408E-10	0.98
total		99.77

aqueous species accounting for 99% or more of k+

species	molal conc	per cent
k+	0.6804E-03	98.98
kcl(aq)	0.7040E-05	1.02
total		100.00

aqueous species accounting for 99% or more of na+

species	molal conc	per cent
na+	0.9400E-02	97.92
nacl(aq)	0.1875E-03	1.95
total		99.87

aqueous species accounting for 99% or more of sio2(aq)

species	molal conc	per cent
sio2(aq)	0.6227E-02	98.38
hsio3-	0.9054E-04	1.43
total		99.81

----- summary of aqueous redox reactions -----

couple	eh, volts	pe-	log fo2	ah, kcal
default	-0.441	-0.4248E+01	-35.301	-10.169
clo4- /cl-	-0.441	-0.4248E+01	-35.301	-10.169
fe+++ /fe++	-0.441	-0.4248E+01	-35.301	-10.169
h2(aq) /h2o	-0.441	-0.4248E+01	-35.301	-10.169
o2(aq) /h2o	-0.441	-0.4248E+01	-35.301	-10.169

----- summary of aqueous non-equilibrium non-redox reactions -----

couple	affinity, kcal
none	

----- summary of stoichiometric mineral saturation states -----

(minerals with affinities .lt. -10 kcal are not listed)

mineral	log q/k	aff, kcal	state	mineral	log q/k	aff, kcal	state
albite	0.000	0.000	satd	albite high	-0.490	-1.173	
albite low	0.000	-0.001	satd	analcime	-0.388	-0.929	
analcime-dehy	-3.475	-8.318		andalusite	-2.190	-5.243	
annite	-0.767	-1.837		beidellite-h	-1.606	-3.844	
beidellite-k	-1.277	-3.056		beidellite-na	-0.951	-2.277	

```
boehmite                -1.095     -2.621                 chalcedony              -0.155     -0.370  satd
chamosite-7a            -2.151     -5.148                 clinoptilolite-na       -3.443     -8.241
coesite                 -0.489     -1.170                 corundum                -2.664     -6.378
cristobalite            -0.266     -0.636                 cristobalite-a          -0.266     -0.636
cristobalite-b          -0.441     -1.056                 cronstedtite-7a         -1.650     -3.949
daphnite-14a            -1.927     -4.614                 diaspore                -0.909     -2.176
fayalite                -1.606     -3.845                 fe(oh)2                 -3.085     -7.385
feo                     -1.738     -4.160                 ferrosilite             -0.819     -1.962
gibbsite                -1.501     -3.594                 goethite                -1.435     -3.434
greenalite              -2.242     -5.367                 hematite                 0.000      0.000  satd
hercynite               -1.541     -3.689                 ice                     -1.122     -2.686
jadeite                 -1.643     -3.934                 k-feldspar               0.000      0.000  satd
kalsilite               -1.730     -4.141                 kaolinite               -1.313     -3.144
kyanite                 -2.243     -5.370                 magnetite                0.000      0.000  satd
maximum microcline       0.015      0.035  satd           minnesotaite            -0.877     -2.100
muscovite                0.000      0.000  satd           natrolite               -2.493     -5.968
nepheline               -1.685     -4.033                 nontronite-h            -0.985     -2.359
nontronite-k            -0.660     -1.580                 nontronite-na           -0.336     -0.805
paragonite              -0.641     -1.535                 pyrophyllite            -1.490     -3.568
quartz                   0.000      0.000  satd           sanidine high           -0.349     -0.836
sillimanite             -2.319     -5.551                 sio2(am)                -0.513     -1.228
tridymite               -0.371     -0.889                 wustite                 -1.992     -4.768

    9 approx. saturated pure minerals
    0 approx. saturated end-members of specified solid solutions
    0 saturated end-members of hypothetical solid solutions


    0 supersaturated pure minerals
    0 supersatd. end-members of specified solid solutions
    0 supersatd. hypothetical solid solution phases



                    ----- summary : gases -----

      gas                          fugacity     log fugacity

  al(g)                        0.116238E-76     -76.93465
  cl2(g)                       0.196287E-34     -34.70600
  h2(g)                        0.268516E-02      -2.57150
  h2o(g)                       0.295122E+02       1.47000
  hcl(g)                       0.871965E-09      -9.05950
  k(g)                         0.312141E-32     -32.50565
  na(g)                        0.408837E-30     -30.38845
  o2(g)                        0.500495E-35     -35.30060
  si(g)                        0.198061E-84     -84.70320

----- end of output -----
--- pickup file has been successfully written ---

--- reading the input file ---

--- no further input found ---

        start time =  09:25       3Dec91
          end time =  09:25       3Dec91

        user time =       4.080
         cpu time =       0.890
normal exit
```

7.7. Computing Eh from a Redox Couple: An Example

This test case illustrates the computation of Eh (or *pe*, oxygen fugacity, or *Ah*) from data for both members of a redox couple. The fluid is an acid ($pH = 1.1$) mine water whose composition is taken from Nordstrom, Jenne, and Ball (1979). The redox state is calculated for the Fe^{2+}-Fe^{3+} couple. This is possible because the concentration of each form of dissolved iron is sufficiently high to be measured. The objective is to compare the *Eh* calculated for this couple with the measured *Eh* of +622 mV. This problem was run using the **com** data file, and the activity coefficients are calculated from the B-dot equation. The input file in both formats is given below, followed by the **output** file, beginning with the message announcing the completion of Newton-Raphson iteration.

In this particular case, the measured *Eh* of +622 mV was used as the default redox parameter. This was accomplished by setting **iopt1** = -1 and **fep** = 0.622. This was used to constrain all the redox couples in the solution, except that for Fe^{2+}-Fe^{3+}. For the latter couple, a separate analytical concentration was entered for each member, and the corresponding redox state then calculated. Of particular interest is the following table:

```
      ----- summary of aqueous redox reactions -----

   couple          eh, volts       pe-        log fo2      ah, kcal

default              0.622    0.1050E+02     -36.590       14.345
fe+++   /fe++        0.718    0.1212E+02     -30.115       16.557
h2(aq) /h2o          0.622    0.1050E+02     -36.590       14.345
hs-     /so4--       0.622    0.1050E+02     -36.590       14.345
o2(aq)  /h2o         0.622    0.1050E+02     -36.590       14.345
so4--   /so4--       0.622    0.1050E+02     -36.590       14.345
```

Here we see that the redox state of the ferrous-ferric couple expressed as *Eh* is +718 mV, higher than the measured value of +622 mV. The *Eh* of all other redox couples matches the default value. If we had set **iopg1** = 1 and **uredox** = 'fe+++', the default redox state would have been determined by that for the ferrous-ferric couple instead.

Is the difference between the computed *Eh* for the ferrous-ferric couple and the measured *Eh* significant? This is not immediately obvious. To pursue this question, one might like to know the likely error in the measured *Eh*. One might also like to estimate the uncertainty in the calculated *Eh* of the ferrous-ferric couple due to the analytical uncertainties in the measurement of the concentration of the two forms of dissolved iron. One might also like to estimate the component of uncertainty in this quantity arising from uncertainty in the measurement of the *pH*. Recall that the reported *pH* value was 1.1. Calibration buffers are generally available for *pH* values of about 4.0, 7.0, and 10.0. Thus it is likely that the measurement involved considerable extrapolation, and the true uncertainty is probably at least a few tenths of a *pH* unit. Note also that the calculated charge imbalance for the total water analysis on the high side (-16% of the total ionic charge). In addition to those considerations, uncertainty in the calculated *Eh* also arises from uncertainties in the thermodynamic data and the activity coefficients. Nordstrom, Jenne, and Ball (1979) were partly able to get around such problems by plotting the ferrous-ferric *Eh* versus the measured *Eh* for a suite of such waters.

Although this example involves the additional data required to evaluate the redox state of only one couple, data may be specified to allow the determination of the redox states of any number of redox couples. Generally speaking, two analytical data inputs are required per couple. However, if water is one of the members of a couple, only an analytical data input for the other member is required.

The EQ3NR **input** file (**acidmwb.3i**), the redox couple test case ("W" format):

```
EQ3NR input file name= acidmwb.3i
Description= "Acid mine water, Hornet Effluent"
Version number= 3245   Stage number= 01
Created 06/08/90    Creator= T.J. Wolery
Revised 06/08/90    Revisor= T.J. Wolery

  Acid mine water, Hornet Effluent. Analysis from Nordstrom, Jenne,
and Ball (1979, Table II, column B). Note that separate analyses are
present for Fe++ and Fe+++, permitting the calculation of the Eh
specific to this couple. This may then be compared with the measured
```

```
Eh.

  Purpose: to test the code in the case of an acid mine water.

                          References

Nordstrom, D. K., Jenne, E. A., and Ball, J. W., 1979, Redox
  equilibria of iron in acid mine waters, in Jenne, E. A., editor,
  Chemical Modeling in Aqueous Systems, ACS Symposium Series,
  v. 93, American Chemical Society, Washington, D.C., p. 51-79.

endit.
      tempc=          25.5
        rho=            1.     tdspkg=            0.     tdspl=            0.
        fep=         0.622     uredox=
     tolbt=            0.      toldl=            0.     tolsat=           0.
    itermx=  0
*               1    2    3    4    5    6    7    8    9   10
   iopt1-10=   -1    0    0    0    0    0    0    0    0    0
   iopg1-10=    0    0    0    0    0    0    0    0    0    0
   iopr1-10=    0    0    0    0    0    0    0    0    0    0
  iopr11-20=    0    0    0    0    0    0    0    0    0    0
   iodb1-10=    0    0    0    0    0    0    0    0    0    0
      ubal= none
     nxmod=  0
*
data file master species= h+
   switch with species=
   jflag= 16   csp= -1.10
data file master species= ca++
   switch with species=
   jflag=  2   csp= 173.
data file master species= mg++
   switch with species=
   jflag=  2   csp= 685.
data file master species= na+
   switch with species=
   jflag=  2   csp= 92.5
data file master species= k+
   switch with species=
   jflag=  2   csp= 128.
data file master species= fe++
   switch with species=
   jflag=  2   csp= 9050.
data file master species= fe+++
   switch with species=
   jflag=  2   csp= 2650.
data file master species= al+++
   switch with species=
   jflag=  2   csp= 1400.
data file master species= sio2(aq)
   switch with species=
   jflag=  2   csp= 130.
data file master species= so4--
   switch with species=
   jflag=  2   csp= 60000.
endit.
```

he EQ3NR input file (**acidmwb.3i**), the redox couple test case ("D" format):

```
EQ3NR input file name= acidmwb.3i
Description= "Acid mine water, Hornet Effluent"
Version number= 3245   Stage number= 01
Created 06/08/90     Creator= T.J. Wolery
Revised 06/08/90     Revisor= T.J. Wolery

  Acid mine water, Hornet Effluent. Analysis from Nordstrom, Jenne,
and Ball (1979, Table II, column 8). Note that separate analyses are
present for Fe++ and Fe+++, permitting the calculation of the Eh
specific to this couple. This may then be compared with the measured
Eh.

  Purpose: to test the code in the case of an acid mine water.

                          References

Nordstrom, D. K., Jenne, E. A., and Ball, J. W., 1979, Redox
  equilibria of iron in acid mine waters, in Jenne, E. A., editor,
  Chemical Modeling in Aqueous Systems, ACS Symposium Series,
  v. 93, American Chemical Society, Washington, D.C., p. 51-79.

Temperature (C)          | 25.50        |Density(gm/cm3)|  1.00000
Total Dissolved Salts    |              | mg/kg | mg/l  |*not used
```

Electrical Balancing on | | code selects | *not performed

SPECIES	BASIS SWITCH/CONSTRAINT	CONCENTRATION	UNITS OR TYPE
redox		0.6220	eh
h+		1.1000	*ph*
ca++		173.00	mg/l
mg++		685.00	mg/l
na+		92.500	mg/l
k+		128.00	mg/l
fe++		9050.0	mg/l
fe+++		2650.0	mg/l
al+++		1400.0	mg/l
sio2(aq)		130.00	mg/l
so4--		60000.	mg/l

Input Solid Solutions

none		

SUPPRESSED SPECIES (suppress,replace,augmentk,augmentg)	value
none	

OPTIONS

- SOLID SOLUTIONS -
  - * *ignore solid solutions*
    process hypothetical solid solutions
    process input and hypothetical solid solutions
- LOADING OF SPECIES INTO MEMORY -
  - * does nothing
    lists species loaded into memory
- ECHO DATABASE INFORMATION -
  - * does nothing
    lists all reactions
    lists reactions and log K values
    lists reactions, log K values and polynomial coef.
- *LIST OF AQUEOUS SPECIES (ordering)* -
  - * in order of decreasing concentration
    in same order as input file
- LIST OF AQUEOUS SPECIES (concentration limit) -
  - * all species
    only species > 10**-20 molal
    only species > 10**-12 molal
    not printed
- LIST OF AQUEOUS SPECIES (by element) -
  - * print major species
    print all species
    don't print
- MINERAL SATURATION STATES -
  - * print if affinity > -10 kcals
    print all
    don't print
- pH SCALE CONVENTION -
  - * modified NBS
    internal
    rational
- ACTIVITY COEFFICIENT OPTIONS -
  - * use B-dot equation
    Davies' equation
    Pitzer's equations
- AUTO BASIS SWITCHING -
  - * off
    on
- PITZER DATABASE INFORMATION -
  - * print only warnings
    print species in model and number of Pitzer coefficients
    print species in model and names of Pitzer coefficients
- PICKUP FILE -
  - * write pickup file
    don't write pickup file
- LIST MEAN IONIC PROPERTIES -
  - * don't print
    print
- LIST AQUEOUS SPECIES, ION SIZES, AND HYDRATION NUMBERS -
  - * print
    don't print
- CONVERGENCE CRITERIA -
  - * test both residual functions and correction terms
    test only residual functions

DEBUGGING SWITCHES (0-off, 1,2-on, default is off)

0	generic debugging information	2
0	print details of pre-Newton-Raphson iteration	2
0	print details of Newton-Raphson iteration	
0	print details of stoichiometric factors	2
0	print details of stoichiometric factors calculation	
0	write reactions on RLIST	
0	list stoichiometric concentrations of master species	
0	request iteration variables to be killed	

DEVELOPMENT OPTIONS (used for code development)		
none		
TOLERANCES	(desired values)	(defaults)
residual functions		1.e-10
correction terms		1.e-10
saturation state		0.5
number of N-R iterations		30

The EQ3NR **output** file (**acidmwb.3o**) the redox couple test case (beginning with the message announcing the end of Newton-Raphson iteration):

```
                    .
                    .
                    .
              (Material deleted)
                    .
                    .
                    .
Hybrid newton-raphson iteration converged in   7  teps.


                  ----- Summary of the Aqueous  hase -----

        ----- Elemental composition of the aqueous phase -----

     element      mg/l          mg/kg          moles/kg

       o       0.92814E+06   0.92614E+06   0.5 .1113224E+02
       al        1400.0        1400.0      0.5 8734687E-01
       ca         173.00        173.00     0.4 1658315 E-02
       fe        11700.        11700.      0.2 95009830E+00
       h       0.11228E+06   0.11228E+06   0.1 1113975425E+03
       k         128.00        128.00      0.3 3273799725E-02
       mg        685.00        685.00      0.2 8183505 8E-01
       na        92.500        92.500      0.4 23528825E-02
       si        60.767        60.767      0.2 63626771E-02
       s         20028.        20028.      0.6 245863179E+00

        ----- elemental composition as strict basis species -----

     species          mg/l          mg/kg          moles/kg

  h2o              0.10451E-07   0.10451E-07   0.5801113224E+02
  al+++             1400.0        1400.0       0.5188734687E-01
  ca++              173.00        173.00       0.4316583154E-02
  fe++              11700.        11700.       0.2095009830E+00
  h+              0.11228E-06   0.11228E-06    0.1113975425E+03
  k+                128.00        128.00       0.3273799725E-02
  mg++              685.00        685.00       0.2818350558E-01
  na+               92.500        92.500       0.4023528825E-02
  sio2(aq)          130.00        130.00       0.2163626771E-02
  so4--             60000.        60000.       0.6245863179E+00

       --- equivalent composition of the aqueous phase (cte balances) ---

        original basis                existing basis

  species          moles/kg h2o      specie           moles/kg h2o

h2o              0.5801113224E+02   h2o              0.5801113224E+02
al+++            0.5188734687E-01   al+++            0.5188734687E-01
ca++             0.4316583154E-02   ca++             0.4316583154E-02
fe++             0.1620499069E-00   fe++             0.1620499069E+00
h+               0.1113975425E+03   h+               0.1113975425E+03
k+               0.3273799725E-02   k+               0.3273799725E-02
mg++             0.2818350558E-01   mg++             0.2818350558E-01
na+              0.4023528825E-02   na+              0.4023528825E-02
sio2(aq)         0.2163626771E-02   sio2(aq)         0.2163626771E-02
so4--            0.6245863179E+00   so4--            0.6245863179E+00
fe+++            0.4745107604E-01   fe+++            0.4745107604E-01

Single ion activities and activity coefficients are here defined
With respect to the modified nbs ph scale
```

	ph	eh	pe
modified nbs ph scale	1.1000	0.6220	1.0497E+01
rational ph scale	0.9972	0.6281	1.0600E+01

activity of water = 0.98698
log activity of water = -0.00569

true osmotic coefficient= 0.86473
stoichiometric osmotic coefficient= 0.52784

sum of true molalities= 0.8410930958290
sum of stoichiometric molalities= 1.3779110304192

true ionic strength= 0.9234813342655
stoichiometric ionic strength= 2.3659275751669

----- electrical balance totals -----

	equiv/kg
sigma(mz) cations =	0.4554314096E+00
sigma(mz) anions =	-0.6295683831E+00
total charge =	0.1084999793E+01
mean charge =	0.5424998963E+00
charge imbalance =	-0.1741369735E+00

total charge = sigma(mz) cations + abs (sigma(mz) anions)
mean charge = 1/2 total charge

the electrical imbalance is

-16.0	per cent of the total charge
-32.1	per cent of the mean charge
-38.2	per cent of sigma(mz) cations
-27.7	per cent of abs (sigma(mz) anions)

----- activity ratios of ions -----

log (act(al+++) / act(h+)xx 3) =	-0.0861
log (act(ca++) / act(h+)xx 2) =	-1.0602
log (act(fe++) / act(h+)xx 2) =	0.4761
log (act(k+) / act(h+)xx 1) =	-1.6809
log (act(mg++) / act(h+)xx 2) =	-0.3497
log (act(na+) / act(h+)xx 1) =	-1.5500
log (act(sio2(aq)) =	-2.6648
log (act(so4--) x act(h+)xx 2) =	-3.8058
log (act(fe+++) / act(h+)xx 3) =	0.6877
log (act(h2(aq)) =	-26.2886
log (act(hs-) x act(h+)xx 1) =	-62.8963
log (act(o2(aq)) =	-39.4856
log (act(oh-) x act(h+)xx 1) =	-13.9843
log (act(so3--) x act(h+)xx 2) =	-30.6086

----- distribution of aqueous species -----

species	molal conc	log conc	log g	activity	log act
hso4-	0.2789E+00	-0.5546	-0.1662	0.1902E+00	-0.7208
so4--	0.1582E+00	-0.8008	-0.8050	0.2479E-01	-1.6058
h+	0.1006E+00	-0.9972	-0.1028	0.7943E-01	-1.1000
fe++	0.8774E-01	-1.0568	-0.6671	0.1888E-01	-1.7236
feso4(aq)	0.7431E-01	-1.1290	0.0000	0.7431E-01	-1.1290
fe+++	0.3466E-01	-1.4601	-1.1522	0.2442E-02	-2.6123
al(so4)2-	0.2970E-01	-1.5273	-0.1662	0.2025E-01	-1.6935
mgso4(aq)	0.1828E-01	-1.7380	0.0000	0.1828E-01	-1.7380
also4+	0.1635E-01	-1.7864	-0.1921	0.1051E-01	-1.9785
mg++	0.9902E-02	-2.0043	-0.5454	0.2820E-02	-2.5497
feso4+	0.8118E-02	-2.0905	-0.1921	0.5216E-02	-2.2826
al+++	0.5836E-02	-2.2339	-1.1522	0.4111E-03	-3.3861
fe(so4)2-	0.3598E-02	-2.4440	-0.1662	0.2454E-02	-2.6102
na+	0.3484E-02	-2.4579	-0.1921	0.2239E-02	-2.6500
k+	0.2795E-02	-2.5536	-0.2273	0.1656E-02	-2.7809
ca++	0.2552E-02	-2.5931	-0.6671	0.5493E-03	-3.2602
sio2(aq)	0.2164E-02	-2.6648	0.0000	0.2164E-02	-2.6648
caso4(aq)	0.1764E-02	-2.7534	0.0000	0.1764E-02	-2.7534
fehso4++	0.1052E-02	-2.9781	-0.8000	0.1667E-03	-3.7781
naso4-	0.5392E-03	-3.2683	-0.1662	0.3677E-03	-3.4345
kso4-	0.4569E-03	-3.3401	-0.1662	0.3116E-03	-3.5064
khso4(aq)	0.2165E-04	-4.6645	0.0000	0.2165E-04	-4.6645
h2so4(aq)	0.1494E-04	-4.8258	0.0000	0.1494E-04	-4.8258
feoh++	0.8195E-05	-5.0865	-0.8000	0.1299E-05	-5.8865
fe3(oh)4(5+)	0.3355E-05	-5.4743	-4.2722	0.1793E-09	-9.7465
fe(oh)2-	0.1316E-05	-5.8808	-0.1921	0.8155E-06	-6.0729
aloh++	0.3252E-06	-6.4878	-0.8000	0.5154E-07	-7.2878
fe2(oh)2++++	0.3857E-07	-7.4138	-2.8876	0.4995E-10	-10.3014

al2(oh)2++++	0.4354E-09	-9.3611	-2.8876	0.5640E-12	-12.2487
al(oh)2+	0.8338E-11	-11.0789	-0.1921	0.5357E-11	-11.2710
hsio3-	0.4473E-11	-11.3494	-0.1662	0.3050E-11	-11.5156
fe(oh)3(aq)	0.3146E-11	-11.5022	0.0000	0.3146E-11	-11.5022
mgh3sio4+	0.3347E-12	-12.4754	-0.1921	0.2150E-12	-12.6675
nahsio3(aq)	0.3010E-12	-12.5214	0.0000	0.3010E-12	-12.5214
oh-	0.2076E-12	-12.6828	-0.2015	0.1305E-12	-12.8843
nah3sio4(aq)	0.1308E-12	-12.8833	0.0000	0.1308E-12	-12.8833
cah3sio4+	0.3665E-13	-13.4359	-0.1921	0.2355E-13	-13.6280
caoh+	0.1567E-14	-14.8049	-0.1921	0.1007E-14	-14.9970
al3(oh)4(5+)	0.4751E-15	-15.3232	-4.2722	0.2539E-19	-19.5954
naoh(aq)	0.1908E-15	-15.7193	0.0000	0.1908E-15	-15.7193
koh(aq)	0.7135E-16	-16.1466	0.0000	0.7135E-16	-16.1466
al(oh)3(aq)	0.6169E-16	-16.2098	0.0000	0.6169E-16	-16.2098
mg(h3sio4)2(aq)	0.3756E-19	-19.4253	0.0000	0.3756E-19	-19.4253
fe(oh)4-	0.2340E-19	-19.6308	-0.1662	0.1596E-19	-19.7970
fe(oh)2(aq)	0.7924E-20	-20.1010	0.0000	0.7924E-20	-20.1010
mgh2sio4(aq)	0.3108E-20	-20.5075	0.0000	0.3108E-20	-20.5075
al(oh)4-	0.1155E-20	-20.9375	-0.1662	0.7875E-21	-21.1037
h6(h2sio4)4--	0.7141E-21	-21.1462	-0.8050	0.1119E-21	-21.9512
h2so3(aq)	0.4026E-21	-21.3951	0.0000	0.4026E-21	-21.3951
ca(h3sio4)2(aq)	0.3421E-21	-21.4658	0.0000	0.3421E-21	-21.4658
so2(aq)	0.2952E-21	-21.5298	0.0000	0.2952E-21	-21.5298
hso3-	0.7339E-22	-22.1344	-0.1662	0.5005E-22	-22.3006
cah2sio4(aq)	0.5035E-22	-22.2980	0.0000	0.5035E-22	-22.2980
h2sio4--	0.2725E-22	-22.5647	-0.8050	0.4269E-23	-23.3696
h2(aq)	0.4318E-26	-26.3647	0.0761	0.5145E-26	-26.2886
so3--	0.2491E-27	-27.6036	-0.8050	0.3903E-28	-28.4086
fe(oh)3-	0.5795E-31	-31.2369	-0.1662	0.3952E-31	-31.4032
s2o6--	0.6708E-36	-36.1734	-0.8050	0.1051E-36	-36.9784
h4(h2sio4)4----	0.2161E-38	-38.6654	-3.3878	0.8888E-42	-42.0512
hso5-	0.2966E-39	-39.5279	-0.1662	0.2023E-39	-39.6941
o2(aq)	0.2743E-39	-39.5617	0.0761	0.3269E-39	-39.4856
mg4(oh)4++++	0.2618E-42	-42.5821	-2.8876	0.3190E-45	-45.4697
hs2-	0.1797E-46	-46.7454	-0.1662	0.1226E-46	-46.9117
s2o8--	0.1635E-46	-46.7865	-0.8050	0.2562E-47	-47.5915
s2o5--	0.2460E-48	-48.6090	-0.8050	0.3855E-49	-49.4140
h2s(aq)	0.1215E-55	-55.9154	0.0000	0.1215E-55	-55.9154
s2o3--	0.1209E-58	-58.9177	-0.8050	0.1894E-59	-59.7226
hs2o3-	0.1256E-59	-59.9011	-0.1662	0.8565E-60	-60.0673
hs-	0.2542E-61	-61.5949	-0.2015	0.1598E-61	-61.7963
s2o4--	0.2482E-63	-63.6051	-0.8050	0.3889E-64	-64.4101
s--	0.1542E-72	-72.8120	-0.8050	0.2416E-73	-73.6170
s3o6--	0.5604E-75	-75.2515	-0.8050	0.8780E-76	-76.0565
s4o6--	0.3624E-98	-98.4408	-0.8050	0.5679E-99	-99.2457
al13o4(oh)24(7+)	0.4631E-99	-98.3344	-8.3738	0.1958-107	-107.7081
s2--	0.4771-109	-109.3214	-0.8050	0.7476-110	-110.1263
s3--	0.1232-145	-145.9092	-0.8050	0.1931-146	-146.7142
s5o6--	0.3309-150	-150.4803	-0.8050	0.5185-151	-151.2852
s4--	0.1920-182	-182.7167	-0.8050	0.3008-183	-183.5217
s5--	0.1804-219	-219.7437	-0.8050	0.2827-220	-220.5486

----- major aqueous species contributing to mass balances -----

aqueous species accounting for 99% or more of al+++

species	molal conc	per cent
al(so4)2-	0.2970E-01	57.23
also4+	0.1635E-01	31.52
al+++	0.5836E-02	11.25
total		100.00

aqueous species accounting for 99% or more of ca++

species	molal conc	per cent
ca++	0.2552E-02	59.12
caso4(aq)	0.1764E-02	40.88
total		100.00

aqueous species accounting for 99% or more of fe++

species	molal conc	per cent
fe++	0.8774E-01	54.14
feso4(aq)	0.7431E-01	45.86
total		100.00

aqueous species accounting for 99% or more of k+

species	molal conc	per cent
k+	0.2795E-02	85.38
kso4-	0.4569E-03	13.96

total	99.34

aqueous species accounting for 99% or more of mg++

species	molal conc	per cent
mgso4(aq)	0.1828E-01	64.87
mg++	0.9902E-02	35.13
total		100.00

aqueous species accounting for 99% or more of na+

species	molal conc	per cent
na+	0.3484E-02	86.60
naso4-	0.5392E-03	13.40
total		100.00

aqueous species accounting for 99% or more of sio2(aq)

species	molal conc	per cent
sio2(aq)	0.2164E-02	100.00
total		100.00

aqueous species accounting for 99% or more of so4--

species	molal conc	per cent
hso4-	0.2789E+00	44.65
so4--	0.1582E+00	25.33
feso4(aq)	0.7431E-01	11.90
al(so4)2-	0.2970E-01	9.51
mgso4(aq)	0.1828E-01	2.93
also4+	0.1635E-01	2.62
feso4+	0.8118E-02	1.30
fe(so4)2-	0.3598E-02	1.15
total		99.38

aqueous species accounting for 99% or more of fe+++

species	molal conc	per cent
fe+++	0.3466E-01	73.05
feso4+	0.8118E-02	17.11
fe(so4)2-	0.3598E-02	7.58
fehso4++	0.1052E-02	2.22
total		99.96

----- summary of aqueous redox reactions -----

couple		eh, volts	pe-	log fo2	ah, kcal
default		0.622	0.1050E+02	-36.584	14.345
fe+++	/fe++	0.718	0.1212E+02	-30.109	16.557
h2(aq)	/h2o	0.622	0.1050E+02	-36.584	14.345
hs-	/so4--	0.622	0.1050E+02	-36.584	14.345
o2(aq)	/h2o	0.622	0.1050E+02	-36.584	14.345
so3--	/so4--	0.622	0.1050E+02	-36.584	14.345

----- summary of aqueous non-equilibrium non-redox reactions -----

couple	affinity, kcal
none	

----- summary of stoichiometric mineral saturation states -----

(minerals with affinities .lt. -10 kcal are not listed)

mineral	log q/k	aff, kcal	state	mineral	log q/k	aff, kcal	state
alum-k	-3.712	-5.073		alunite	-6.615	-9.040	
anhydrite	-0.554	-0.758		arcanite	-5.374	-7.344	
bassanite	-1.202	-1.643		caso4:0.5h2o(beta)	-1.369	-1.871	
chalcedony	1.054	1.440	ssatd	coesite	0.516	0.705	ssatd
cristobalite	0.775	1.059	ssatd	cristobalite-a	0.775	1.059	ssatd

```
cristobalite-b          0.333      0.455  satd      epsomite          -2.233     -3.052
fe(oh)3                -4.960     -6.779           feso4             -5.965     -8.151
gibbsite               -6.873     -9.392           glauberite        -6.303     -8.613
goethite                0.160      0.219  satd      gypsum            -0.395     -0.539
hematite                1.288      1.760  ssatd     hexahydrite       -2.463     -3.366
ice                    -0.146     -0.200  satd      jarosite           2.163      2.956  ssatd
jarosite-na            -1.684     -2.302           kieserite         -3.894     -5.322
leonite                -7.234     -9.885           melanterite       -1.024     -1.399
mercallite             -4.048     -5.532           mirabilite        -5.046     -7.989
montronite-ca           3.239      4.426  ssatd     montronite-h       4.369      5.971  ssatd
montronite-k            3.138      4.288  ssatd     montronite-mg      3.395      4.639  ssatd
montronite-na           2.844      3.887  ssatd     pentahydrite      -2.797     -3.822
picromerite            -6.918     -9.453           polyhalite        -6.754     -9.230
quartz                  1.324      1.810  ssatd     sio2(am)           0.043      0.058  satd
starkeyite             -3.178     -4.343           syngenite         -4.439     -6.066
thenardite             -6.596     -9.014           tridymite          1.153      1.576  ssatd

    4 approx. saturated pure minerals
    0 approx. saturated end-members of specified solid solutions
    0 saturated end-members of hypothetical solid solutions


   13 supersaturated pure minerals
    0 supersatd. end-members of specified solid solutions
    0 supersatd. hypothetical solid solution phases



                  ----- summary of gases -----

     gas                          fugacity    log fugacity

  al(g)                        0.131176-169   -169.88215
  ca(g)                        0.771569-146   -146.11263
  h2(g)                        0.657122E-23    -23.18235
  h2o(g)                       0.264152E-01     -1.57815
  h2s(g)                       0.119732E-54    -54.92179
  k(g)                         0.551809E-73    -73.25821
  mg(g)                        0.248070-122   -122.60543
  na(g)                        0.391009E-72    -72.40781
  o2(g)                        0.260829E-36    -36.58364
  s2(g)                        0.830317E-89    -89.08076
  si(g)                        0.177981-182   -182.74963
  so2(g)                       0.203265E-21    -21.69194

----- end of output -----
--- pickup file has been successfully written ---

--- reading the input file ---

--- no further input found ---

        start time = 09:24       3Dec91
          end time = 09:24       3Dec91

        user time =       5.630
         cpu time =       1.180
normal exit
```

7.8. The Dead Sea Brine Test Case

Marcus (1977) attempted to calculate the activity of water and the mean activity of potassium chloride in Dead Sea brine. He was forced to use less accurate means than are presently available. Here, we repeat his work using Pitzer's equations and the **hmw** data file (based on Harvie, Møller, and Weare, 1984). As no meaningful *pH* measurement can be made for such a concentrated solution, we have here attempted to obtain an estimate of the *pH* and the *pHCl* function proposed by Knauss, Jackson, and Wolery (1990) by constraining the hydrogen ion to satisfy equilibrium with carbon dioxide in the atmosphere (log fugacity of CO_2 = -3.5). The dissolution reaction for CO_2 gas can be written as:

$$CO_{2(g)} + H_2O_{(l)} = H^+ + HCO_3^- \qquad (225)$$

total	99.34

aqueous species accounting for 99% or more of mg++

species	molal conc	per cent
mgso4(aq)	0.1828E-01	64.87
mg++	0.9902E-02	35.13
total		100.00

aqueous species accounting for 99% or more of na+

species	molal conc	per cent
na+	0.3484E-02	86.60
naso4-	0.5392E-03	13.40
total		100.00

aqueous species accounting for 99% or more of sio2(aq)

species	molal conc	per cent
sio2(aq)	*0.2164E-02*	*100.00*
total		100.00

aqueous species accounting for 99% or more of so4--

species	molal conc	per cent
hso4-	0.2789E+00	44.65
so4--	0.1582E+00	25.33
feso4(aq)	0.7431E-01	11.90
al(so4)2-	0.2970E-01	9.51
mgso4(aq)	0.1828E-01	2.93
also4+	0.1635E-01	2.62
feso4+	0.8118E-02	1.30
fe(so4)2-	0.3598E-02	1.15
total		99.38

aqueous species accounting for 99% or more of fe+++

species	molal conc	per cent
fe+++	0.3466E-01	73.05
feso4+	0.8118E-02	17.11
fe(so4)2-	0.3598E-02	7.58
fehso4++	0.1052E-02	2.22
total		99.96

----- summary of aqueous redox reactions -----

couple		eh, volts	pe-	log fo2	ah, kcal
default		0.622	0.1050E+02	-36.584	14.345
fe+++	/fe++	0.718	0.1212E+02	-30.109	16.557
h2(aq)	/h2o	0.622	0.1050E+02	-36.584	14.345
hs-	/so4--	0.622	0.1050E+02	-36.584	14.345
o2(aq)	/h2o	0.622	0.1050E+02	-36.584	14.345
so3--	/so4--	0.622	0.1050E+02	-36.584	14.345

----- summary of aqueous *non-equilibrium* non-redox reactions -----

couple affinity, kcal

none

----- summary of stoichiometric mineral saturation states -----

(minerals with affinities .lt. -10 kcal are not listed)

mineral	log q/k	aff, kcal	state	mineral	log q/k	aff, kcal	state
alum-k	-3.712	-5.073		alunite	-6.615	-9.040	
anhydrite	-0.554	-0.758		arcanite	-5.374	-7.344	
bassanite	-1.202	-1.643		caso4:0.5h2o(beta)	-1.369	-1.871	
chalcedony	1.054	1.440	ssatd	coesite	0.516	0.705	ssatd
cristobalite	0.775	1.059	ssatd	cristobalite-a	0.775	1.059	ssatd

```
cristobalite-b          0.333      0.455  satd      epsomite          -2.233     -3.052
fe(oh)3                -4.960     -6.779           fes04             -5.965     -8.151
gibbsite               -6.873     -9.392           glauberite        -6.303     -8.613
goethite                0.160      0.219  satd      gypsum            -0.395     -0.539
hematite                1.288      1.760  ssatd     hexahydrite       -2.463     -3.366
ice                    -0.146     -0.200  satd      jarosite           2.163      2.956  ssatd
jarosite-na            -1.684     -2.302           kieserite         -3.894     -5.322
leonite                -7.234     -9.885           melanterite       -1.024     -1.399
mercallite             -4.048     -5.532           mirabilite        -5.846     -7.989
nontronite-ca           3.239      4.426  ssatd     nontronite-h       4.369      5.971  ssatd
nontronite-k            3.138      4.288  ssatd     nontronite-mg      3.395      4.639  ssatd
nontronite-na           2.844      3.887  ssatd     pentahydrite      -2.797     -3.822
picromerite            -6.918     -9.453           polyhalite        -6.754     -9.230
quartz                  1.324      1.810  ssatd     sio2(am)          0.043      0.050  satd
starkeyite             -3.178     -4.343           syngenite         -4.439     -6.066
thenardite             -6.596     -9.014           tridymite          1.153      1.576  ssatd

    4 approx. saturated pure minerals
    0 approx. saturated end-members of specified solid solutions
    0 saturated end-members of hypothetical solid solutions

   13 supersaturated pure minerals
    0 supersatd. end-members of specified solid solutions
    0 supersatd. hypothetical solid solution phases


                  ----- summary of gases -----

      gas                         fugacity    log fugacity

  al(g)                       0.131176-169    -169.88215
  ca(g)                       0.771569-146    -146.11263
  h2(g)                       0.657122E-23     -23.18235
  h2o(g)                      0.264152E-01      -1.57815
  h2s(g)                      0.119732E-54     -54.92179
  k(g)                        0.551809E-73     -73.25821
  mg(g)                       0.248070-122    -122.60543
  na(g)                       0.391009E-72     -72.40781
  o2(g)                       0.260829E-36     -36.58364
  s2(g)                       0.830317E-89     -89.08076
  si(g)                       0.177981-182    -182.74963
  so2(g)                      0.203265E-21     -21.69194

----- end of output -----
--- pickup file has been successfully written ---

--- reading the input file ---

--- no further input found ---

        start time = 09:24       3Dec91
          end time = 09:24       3Dec91

        user time =       5.630
         cpu time =       1.180
normal exit
```

7.8. The Dead Sea Brine Test Case

Marcus (1977) attempted to calculate the activity of water and the mean activity of potassium chloride in Dead Sea brine. He was forced to use less accurate means than are presently available. Here, we repeat his work using Pitzer's equations and the **hmw** data file (based on Harvie, Møller, and Weare, 1984). As no meaningful *pH* measurement can be made for such a concentrated solution, we have here attempted to obtain an estimate of the *pH* and the *pHCl* function proposed by Knauss, Jackson, and Wolery (1990) by constraining the hydrogen ion to satisfy equilibrium with carbon dioxide in the atmosphere (log fugacity of CO_2 = -3.5). The dissolution reaction for CO_2 gas can be written as:

$$CO_{2(g)} + H_2O_{(l)} = H^+ + HCO_3^- \qquad (225)$$

The analytical data include a measurement of bicarbonate, which frees this reaction to be used as a constraint on the hydrogen ion. The **input** file is presented in both formats, and the **output** file is given starting with the message announcing the end of Newton-Raphson iteration.

Marcus (1977) concluded that the activity of water in Dead Sea brine was 0.754±0.004. The EQ3NR calculation gives a value of 0.750, in good agreement. Marcus also concluded that the mean ionic activity of KCl was in the range 0.876-1.199. The value calculated by EQ3NR is somewhat lower, 0.827. This is taken from the table of mean ionic properties, which was written on the **output** file because the option switch **iopr6** was set to 1 on the **input** file. The *pH* calculated by EQ3NR is 7.43 on the NBS scale and 8.50 on the scale on which $log\gamma_{H^+} = 0$. The *pHCl* is 6.94. Although the *pH* of Dead Sea brine can not be measured in the usual way, it should be possible to measure the *pHCl* using the method proposed by Knauss, Jackson, and Wolery (1990).

Ben-Yaakov and Sass (1977) attempted to measure the *pH* of artificial Dead Sea brine using a procedure that was conceptually very similar to the recommended *pHCl* method. Their artificial brine is very similar to that of Marcus (1977), but not identical. Using an electrochemical cell that in theory should respond linearly to *pHCl*, they took emf measurements on the artificial brine and three *HCl* solutions. In order to obtain the *pH* from their results, they had to estimate the activity coefficient of the chloride ion. They did this by first calculating the mean activity coefficient of potassium chloride in the brine using the Harned rule (Harned and Owen, 1958). This is an older approach to estimating activity coefficients in electrolyte mixtures which does not possess the accuracy of Pitzer's equations. They then estimated the activity coefficient of the chloride ion using the MacInnes (1919) convention:

$$\gamma_{Cl^-} = \gamma_{K^+} = \gamma_{\pm,KCl} \tag{226}$$

Using this approach, they determined that the *pH* of the artificial brine was 5.86. They compared this to the result of a conventional *pH* measurement, which gave a value of 6.22. Their value of 5.86 certainly differs from the value of 7.43 that we obtained by assuming equilibrium with atmospheric carbon dioxide. What does this mean?

Ben-Yaakov and Sass (1977) obtained a value of 0.757 for the mean activity coefficient of *KCl*. The value obtained in our test (see the **output** file) is 0.823. The corresponding values for the logarithm of this quantity are -0.1209 and -0.0846, respectively. This means that their estimate of *pH* should be lower than ours by only 0.036 unit. A more significant problem is that Ben-Yaakov and Sass' use of the MacInnes convention has put their result on a "MacInnes" *pH* scale. On the NBS scale used in our example, we obtained the following results:

Species	$log\,\gamma_i$	
K^+	0.1055	
Cl^-	-0.2750	
	-0.0846	(on the "MacInnes" scale)
Cl^-	-0.1209	(Ben-Yaakov and Sass, 1977)

If we were to correct our result to the "MacInnes" scale, we would have to add 0.190 *pH* unit to our result, which would give us a *pH* of 7.62. This moves us even farther away from agreement with Ben-Yaakov and Sass.

It seems most likely that the *pH* of Dead Sea water must be closer to the value estimated by Ben-Yaakov and Sass (1977). Therefore, the equilibrium fugacity of carbon dioxide must be higher than the atmospheric value used in our test case. Without conducting new measurements, the best approach to estimating the *pH* of Dead Sea brine is probably to update Ben-Yaakov and Sass' calculation using Pitzer's equations and expressing the results on the NBS scale. This can be done by finding the *pHCl* corresponding to Ben-Yaakov and Sass' emf measurements and using this as an input to EQ3NR (how to use *pHCl* as an input is demonstrated in the following example in this chapter). In order to find this *pHCl*, one could use EQ3NR to calculate the *pHCl* of the three *HCl* solutions that Ben-Yaakov and Sass used as standards. One could then plot their measured emf results against these *pHCl* values. This plot could then be used as a calibration plot to determine the *pHCl* of the artificial brine from Ben-Yaakov and Sass' emf measurement. The resulting calculation would give the equilibrium fugacity of carbon dioxide, which could be compared against the atmospheric value. This would be a good exercise for the user who is particularly interested in brine chemistry.

The EQ3NR **input** file (**deadseabr.3i**), the Dead Sea brine test case ("W" format):

```
EQ3NR input file name= deadseabr.3i
Description= "Dead Sea brine"
Version number= 3245    Stage number= 01
Created 10/29/90    Creator= T.J. Wolery
Revised 10/29/90    Revisor= T.J. Wolery

  Dead Sea brine, after Marcus (1977). Because no pH data are
available, the pH is calculated from the assumption that the
brine is in equilibrium with CO2 in the atmosphere (log PCO2 =
-3.5). According to Marcus, the activity of water in this brine
should be 0.754 +/- 0.004, and the mean ionic activity of KCl
should be in the range 0.876 - 1.199.

  Purpose: to test the code on a small problem involving a very
concentrated brine, using Pitzer's equations to calculate the
activity coefficients of the aqueous species.

  This problem is best addressed using the thermodynamic data base of
Harvie, Moller, and Weare (1984).

  The print option switch iopr6 is set to 1 to direct the code to
print a table of mean ionic properties.

  Because Br- in not part of the Harvie-Moller-Weare model, the
reported 0.0602 m Br- is ignored on this input file.

                        References

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-SO4-
  OH-HCO3-CO3-CO2-H2O system to high ionic strengths at 25 C:
  Geochimica et Cosmochimica Acta, v. 48, p. 723-751.

Marcus, Y., 1977, The activities of potassium chloride and of water
  in Dead Sea brine: Geochimica et Cosmochimica Acta, v. 41, p.
  1739-1744.

endit
      tempc=          25.
        rho=       1.207     tdspkg=          0.     tdspl=          0.
        fep=      -0.760     uredox=
     noibt=           0      toldl=          0     tolsat=          0
     itermx=  0
*                1    2    3    4    5    6    7    8    9   10
  iopt1-10=      0    0    0    0    0    0    0    0    0    0
  iopg1-10=      1    0    0    0    0    0    0    0    0    0
  iopr1-10=      0    0    0    0    0    1    0    0    0    0
 iopr11-20=      0    0    0    0    0    0    0    0    0    0
  iodb1-10=      0    0    0    0    0    0    0    0    0    0
```

```
    uebal= none
    nxmod=  0
*
data file master species= na+
   switch with species=
    jflag=  0   csp= 1.7519
data file master species= k+
   switch with species=
   jflag=  0   csp= 0.1739
data file master species= mg++
   switch with species=
   jflag=  0   csp= 1.5552
data file master species= ca++
   switch with species=
   jflag=  0   csp= 0.4274
data file master species= cl-
   switch with species=
   jflag=  0   csp= 5.8098
data file master species= hco3-
   switch with species=
   jflag=  0   csp= 0.00392
data file mastor species= so4--
   switch with species=
   jflag=  0   csp= 0.0063
data file master species= h+
   switch with species=
   jflag= 21   csp= -3.5
    gas= co2(g)
endit.
```

The EQ3NR input file (deadseabr.3i), the Dead Sea brine test case ("D" format):

```
------------------------------------------------------------------------
EQ3NR input file name= deadseabr.3i
Description= "Dead Sea brine"
Version number= 3245   Stage number= 01
Created 10/29/90    Creator= T.J. Wolery
Revised 10/29/90    Revisor= T.J. Wolery

  Dead Sea brine, after Marcus (1977). Because no pH data are
available, the pH is calculated from the assumption that the
brine is in equilibrium with CO2 in the atmosphere (log PCO2 =
-3.5). According to Marcus, the activity of water in this brine
should be 0.754 +/- 0.004, and the mean ionic activity of KCl
should be in the range 0.876 - 1.199.

  Purpose: to test the code on a small problem involving a very
concentrated brine, using Pitzer's equations to calculate the
activity coefficients of the aqueous species.

  This problem is best addressed using the thermodynamic data base of
Harvie, Moller, and Weare (1984).

  The print option switch ioprt is set to 1 to direct the code to
print a table of mean ionic properties.

  Because Br- in not part of the Harvie-Moller-Weare model, the
reported 0.0602 m Br- is ignored on this input file.

                          References

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-SO4-
  OH-HCO3-CO3-CO2-H2O system to high ionic strengths at 25 C:
  Geochimica et Cosmochimica Acta, v. 48, p. 723-751.

Marcus, Y., 1977, The activities of potassium chloride and of water
  in Dead Sea brine: Geochimica et Cosmochimica Acta, v. 41, p.
  1739-1744.

------------------------------------------------------------------------
Temperature (C)          | 25.00        |Density(gm/cm3)|  1.20700
------------------------------------------------------------------------
Total Dissolved Salts    |              | mg/kg | mg/l  |*not used
------------------------------------------------------------------------
Electrical Balancing on  |              | code selects  |*not performed
------------------------------------------------------------------------
SPECIES    | BASIS SWITCH/CONSTRAINT | CONCENTRATION| UNITS OR TYPE
------------------------------------------------------------------------
redox                                  -.7000         logfo2
na+                                     1.7519        molality
k+                                     0.17390        molality
mg++                                    1.5552        molality
ca++                                   0.42740        molality
cl-                                     5.8098        molality
hco3-                                0.39200E-02      molality
so4--                                0.63000E-02      molality
```

h+	co2(g)	-3.5000	log fugacity

Input Solid Solutions

none			

SUPPRESSED SPECIES (suppress,replace,augmentk,augmentg)	value
none	

OPTIONS

- SOLID SOLUTIONS -
  - * ignore solid solutions
    process hypothetical solid solutions
    process input and hypothetical solid solutions
- LOADING OF SPECIES INTO MEMORY -
  - * does nothing
    lists species loaded into memory
- ECHO DATABASE INFORMATION -
  - * does nothing
    lists all reactions
    lists reactions and log K values
    lists reactions, log K values and polynomial coef.
- LIST OF AQUEOUS SPECIES (ordering) -
  - * in order of decreasing concentration
    in same order as input file
- LIST OF AQUEOUS SPECIES (concentration limit) -
  - * all species
    only species > 10**-20 molal
    only species > 10**-12 molal
    not printed
- LIST OF AQUEOUS SPECIES (by element) -
  - * print major species
    print all species
    don't print
- MINERAL SATURATION STATES -
  - * print if affinity > -10 kcals
    print all
    don't print
- PH SCALE CONVENTION -
  - * modified NBS
    internal
    rational
- ACTIVITY COEFFICIENT OPTIONS -
    use B-dot equation
    Davies' equation
  - * Pitzer's equations
- AQUO BASIS SWITCHING -
  - * off
    on
- PITZER DATABASE INFORMATION -
  - * print only warnings
    print species in model and number of Pitzer coefficients
    print species in model and names of Pitzer coefficients
- PICKUP FILE -
  - * write pickup file
    don't write pickup file
- LIST MEAN IONIC PROPERTIES -
    don't print
  - * print
- LIST AQUEOUS SPECIES, ION SIZES, AND HYDRATION NUMBERS -
  - * print
    don't print
- CONVERGENCE CRITERIA -
  - * test both residual functions and correction terms
    test only residual functions

DEBUGGING SWITCHES (0-off, 1,2-on, default is off)

0	generic debugging information	2
0	print details of pre-Newton-Raphson iteration	2
0	print details of Newton-Raphson iteration	
0	print details of stoichiometric factors	2
0	print details of stoichiometric factors calculation	
0	write reactions on RLIST	
0	list stoichiometric concentrations of master species	
0	request iteration variables to be killed	

DEVELOPMENT OPTIONS (used for code development)

none

TOLERANCES	(desired values)	(defaults)
residual functions		1.e-10
correction terms		1.e-10
saturation state		0.5
number of N-R iterations		30

The EQ3NR output file for the Dead Sea brine test case (beginning with the message announcing the end of Newton-Raphson iteration):

```
.
.
.
(Material deleted)
.
.
.
Hybrid newton-raphson iteration converged in   4 steps.


                ----- Summary of the Aqueous Phase -----

        ----- Elemental composition of the aqueous phase -----

  element       mg/l          mg/kg          moles/kg

    o       0.10727E+07    0.88870E+06    0.5554602070E+02
    ca        20675.         17129.       0.4274000000E+00
    cl      0.24861E+06    0.20597E+06    0.5809800000E+01
    h       0.13506E+06    0.11190E+06    0.1110180110E+03
    c         56.829         47.083       0.3920000001E-02
    k       8206.6         6799.2         0.1739000000E+00
    mg      45624.         37799.         0.1555200000E+01
    na      48613.         40276.         0.1751900000E+01
    s       243.83         202.02         0.6300000000E-02

        ----- elemental composition as strict basis species -----

  species          mg/l          mg/kg          moles/kg

h2o           0.12078E+07    0.10007E+07    0.5554602070E+02
ca++           20675.         17129.        0.4274000000E+00
cl-           0.24861E+06    0.20597E+06    0.5809800000E+01
h+            0.13506E+06    0.11190E+06    0.1110180110E+03
hco3-          288.70         239.19        0.3920000001E-02
k+             8206.6         6799.2        0.1739000000E+00
mg++           45624.         37799.        0.1555200000E+01
na+            48613.         40276.        0.1751900000E+01
so4--          730.48         605.20        0.6300000000E-02

        --- equivalent composition of the aqueous phase (ete balances) ---

        original basis                 existing basis

  species          moles/kg h2o        species          moles/kg h2o

h2o           0.5554602070E+02    h2o           0.5554602070E+02
ca++          0.4274000000E+00    ca++          0.4274000000E+00
cl-           0.5809800000E+01    cl-           0.5809800000E+01
h+            0.1110180110E+03    h+            0.1110180110E+03
hco3-         0.3920000001E-02    hco3-         0.3920000001E-02
k+            0.1739000000E+00    k+            0.1739000000E+00
mg++          0.1555200000E+01    mg++          0.1555200000E+01
na+           0.1751900000E+01    na+           0.1751900000E+01
so4--         0.6300000000E-02    so4--         0.6300000000E-02

single ion activities and activity coefficients are here defined
with respect to the modified nbs ph scale


                                      ph          eh          pe

  modified nbs ph scale             7.4270      0.7830    1.3237E+01
  rational ph scale                 0.5048      0.7193    1.2159E+01

phei =      6.9378


                    activity of water =  0.75020
                log activity of water =  -0.12482

              true osmotic coefficient=     1.64043
      stoichiometric osmotic coefficient=   1.63985

                  sum of true molalities=   9.7255522592305
      sum of stoichiometric molalities=     9.7209644495673

                  true ionic strength=      7.8402407646366
```

stoichiometric ionic strength= 7.8486287476994

----- electrical balance totals -----

	equiv/kg
sigma(mz) cations =	0.5884630387E+01
sigma(mz) anions =	-0.5823980803E+01
total charge =	0.1170861119E+02
mean charge =	0.5854305595E+01
charge imbalance =	0.6064958353E-01

total charge = sigma(mz) cations + abs (sigma(mz) anions)
mean charge = 1/2 total charge

the electrical imbalance is

0.518	per cent of the total charge
1.04	per cent of the mean charge
1.03	per cent of sigma(mz) cations
1.04	per cent of abs (sigma(mz) anions)

----- activity ratios of ions -----

log (act(ca++	) / act(h+)xx 2) =	15.3905
log (act(cl-	) x act(h+)xx 1) =	-6.9378
log (act(hco3-	) x act(h+)xx 1) =	-11.4440
log (act(k+	) / act(h+)xx 1) =	6.7726
log (act(mg++	) / act(h+)xx 2) =	16.2165
log (act(na+	) / act(h+)xx 1) =	8.1563
log (act(so4--	) x act(h+)xx 2) =	-19.7229
log (act(co2(aq)	) =	-4.9818
log (act(co3--	) x act(h+)xx 2) =	-21.7833
log (act(oh-	) x act(h+)xx 1) =	-14.1215

----- distribution of aqueous species -----

species	molal conc	log conc	log g	activity	log act
cl-	0.5810E+01	0.7642	-0.2750	0.3084E+01	0.4892
na+	0.1752E+01	0.2435	0.4858	0.5362E+01	0.7293
mg++	0.1552E+01	0.1910	1.1716	0.2304E+02	1.3625
ca++	0.4268E+00	-0.3697	0.9063	0.3440E+01	0.5366
k+	0.1739E+00	-0.7597	0.1055	0.2217E+00	-0.6542
so4--	0.6300E-02	-2.2007	-2.6683	0.1352E-04	-4.8690
mgco3(aq)	0.2299E-02	-2.6384	0.0000	0.2299E-02	-2.6384
mgoh+	0.6245E-03	-3.2044	0.0599	0.7169E-03	-3.1445
caco3(aq)	0.5735E-03	-3.2415	0.0000	0.5735E-03	-3.2415
co3--	0.5322E-03	-3.2739	-3.6555	0.1177E-06	-6.9294
hco3-	0.5115E-03	-3.2911	-0.7259	0.9615E-04	-4.0170
oh-	0.4793E-05	-5.3194	-1.3751	0.2020E-06	-6.6945
co2(aq)	0.3702E-05	-5.4316	0.4498	0.1043E-04	-4.9818
h+	0.3128E-08	-8.5048	1.0778	0.3741E-07	-7.4270
hso4-	0.795BE-10	-10.0992	-0.2182	0.4816E-10	-10.3173

----- mean ionic properties -----

			true (a)			stoichiometric (b)	
species	species	log a(+/-)	a(+/-)	m(+/-)	g(+/-)	m(+/-)	g(+/-)
ca++	cl-	0.50496	3.199E+00	2.433E+00	1.315E+00	2.434E+00	1.314E+00
ca++	hco3-	-2.49917	3.168E-03	4.816E-03	6.579E-01	1.873E-02	1.692E-01
ca++	so4--	-2.16619	6.820E-03	5.186E-02	1.315E-01	5.189E-02	1.314E-01
ca++	co3--	-3.19639	6.362E-04	1.507E-02	4.221E-02		
ca++	oh-	-4.28417	5.198E-05	2.140E-04	2.429E-01	8.452E-02	6.150E-04
h+	cl-	-3.46891	3.397E-04	1.348E-04	2.520E+00	1.796E+01	1.892E-05
h+	hco3-	-5.72201	1.897E-06	1.265E-06	1.500E+00	4.665E-01	4.066E-06
h+	so4--	-6.57430	2.665E-07	3.950E-07	6.747E-01	2.688E+00	9.916E-08
h+	co3--	-7.26111	5.481E-08	1.733E-07	3.163E-01		
k+	cl-	-0.08253	8.269E-01	1.005E+00	8.227E-01	1.005E+00	8.227E-01
k+	hco3-	-2.33563	4.617E-03	9.431E-03	4.895E-01	2.611E-02	1.768E-01
k+	so4--	-2.05913	8.727E-03	5.754E-02	1.517E-01	5.754E-02	1.517E-01
k+	co3--	-2.74593	1.795E-03	2.525E-02	7.109E-02		
k+	oh-	-3.67438	2.117E-04	9.130E-04	2.318E-01	8.085E-02	2.618E-03
mg++	cl-	0.78028	6.029E+00	3.742E+00	1.611E+00	3.744E+00	1.610E+00
mg++	hco3-	-2.22386	5.972E-03	7.406E-03	8.065E-01	2.880E-02	2.073E-01
mg++	so4--	-1.75322	1.765E-02	9.889E-02	1.785E-01	9.898E-02	1.783E-01
mg++	co3--	-2.78342	1.647E-03	2.874E-02	5.728E-02		
mg++	oh-	-4.00806	9.798E-05	3.291E-04	2.977E-01	1.300E-01	7.537E-04
na+	cl-	0.60925	4.067E+00	3.190E+00	1.275E+00	3.190E+00	1.275E+00
na+	hco3-	-1.64385	2.271E-02	2.994E-02	7.585E-01	8.287E-02	2.768E-01
na+	so4--	-1.13676	7.299E-02	2.684E-01	2.719E-01	2.684E-01	2.719E-01
na+	co3--	-1.82357	1.501E-02	1.178E-01	1.275E-01		
na+	oh-	-2.98260	1.041E-03	2.098E-03	3.592E-01	2.566E-01	4.056E-03

(a) true quantities consistent with the speciation model
(b) stoichiometric quantities consistent with the cte mass

balance lumpings, except that
1. effective cte(h+) = cte(h+) - conc(h2o)
2. effective cte(oh-) = cte(h2o) - conc(h2o)

----- major aqueous species contributing to mass balances -----

aqueous species accounting for 99% or more of ca++

species	molal conc	per cent
ca++	0.4268E+00	99.87
total		99.87

aqueous species accounting for 99% or more of cl-

species	molal conc	per cent
cl-	0.5810E+01	100.00
total		100.00

aqueous species accounting for 99% or more of hco3-

species	molal conc	per cent
mgco3(aq)	0.2299E-02	58.65
caco3(aq)	0.5735E-03	14.63
co3--	0.5322E-03	13.58
hco3-	0.5115E-03	13.05
total		99.91

aqueous species accounting for 99% or more of k+

species	molal conc	per cent
k+	0.1739E+00	100.00
total		100.00

aqueous species accounting for 99% or more of mg++

species	molal conc	per cent
mg++	0.1552E+01	99.81
total		99.81

aqueous species accounting for 99% or more of na+

species	molal conc	per cent
na+	0.1752E+01	100.00
total		100.00

aqueous species accounting for 99% or more of so4--

species	molal conc	per cent
so4--	0.6300E-02	100.00
total		100.00

----- summary of aqueous redox reactions -----

couple	eh, volts	pe-	log fo2	ah, kcal
default	0.783	0.1324E+02	-0.700	18.059

----- summary of aqueous non-equilibrium non-redox reactions -----

couple	affinity, kcal
none	

----- summary of stoichiometric mineral saturation states -----

(minerals with affinities .lt. -10 kcal are not listed)

```
  mineral          log q/k    aff, kcal  state        mineral          log q/k    aff, kcal  state

anhydrite            0.030        0.041  satd     antarcticite        -3.378       -4.608
aphthitalite        -7.169       -9.780           aragonite            1.827        2.492  ssatd
arcanite            -4.401       -6.004           bischofite          -2.064       -3.907
bloedite            -5.069       -6.916           brucite             -1.142       -1.558
cacl2:4h2o          -4.701       -6.414           calcite              2.014        2.747  ssatd
carnallite          -2.904       -3.961           dolomite             5.123        6.989  ssatd
epsomite            -2.499       -3.409           gaylussite          -3.067       -4.184
glauberite          -2.498       -3.408           gypsum              -0.002       -0.002  satd
halite              -0.352       -0.480  satd     hexahydrite         -2.620       -3.575
kainite             -3.853       -5.257           kalicinite          -4.953       -6.757
kieserite           -3.509       -4.787           leonite             -6.204       -8.464
magnesite            2.267        3.093  ssatd     mirabilite          -3.431       -4.680
na2co3:7h2o         -5.884       -8.028           na4ca(so4)3:2h2o    -5.730       -7.818
nahcolite           -2.885       -3.936           natron              -5.894       -8.041
nesquehonite        -0.774       -1.056           oxychloride-mg      -1.408       -1.921
picromerite         -6.105       -8.329           pirssonite          -2.870       -3.915
polyhalite          -4.854       -6.622           sylvite             -1.065       -1.453
syngenite           -3.186       -4.347           thenardite          -3.123       -4.260
thermonatrite       -6.077       -8.291

    3 approx. saturated pure minerals
    0 approx. saturated end-members of specified solid solutions
    0 saturated end-members of hypothetical solid solutions


    4 supersaturated pure minerals
    0 supersatd. end-members of specified solid solutions
    0 supersatd. hypothetical solid solution phases



                  ----- summary of gases -----

     gas                          fugacity     log fugacity

  co2(g)                       0.316228E-03      -3.50000
  h2(g)                        0.477613E-41     -41.32092
  o2(g)                        0.199526E+00      -0.70000

----- end of output -----
--- pickup file has been successfully written ---

--- reading the input file ---

--- no further input found ---

         start time =  17:12       5Dec91
           end time =  17:12       5Dec91

         user time =       1.640
          cpu time =       0.820
normal exit
```

7.9. Using *pHCl* as an Input: An Example

In the earlier example treating sea water with Pitzer's equations, we found that the *pHCl* was computed to be 8.6722. The *pH* value that was used as an input in this example was 8.22 on the NBS scale. The purpose of the present example is to show how *pHCl* can be used in place of *pH*. Here we partially invert the earlier problem by entering the *pHCl*. We should then get back the original *pH*. Other results (i.e., the calculated electrical imbalance, values of the mineral saturation indices) should also match the results from the previous example. This problem was run using the same **hmw** data file. The **input** file is presented in both formats. The **output** file is given starting with the message announcing the end of Newton-Raphson iteration.

The computed *pH* on the NBS scale is printed on the **output** file as "8.2200." The correctness of the inversion is therefore verified. One may compare other computed results to verify that the computed model is otherwise identical to that obtained previously.

The EQ3NR **input** file (**swphcl.3i**), the *pHCl* test case ("W" format):

```
EQ3NR input file name= swphcl.3i
Description= "Sea water, using pHCl as input in place of pH"
Version number= 3245   Stage number= 01
Created 10/07/90     Creator= T.J. Wolery
Revised 10/07/90     Revisor= T.J. Wolery

  Sea water, including only the major ions. This is a considerably
pared-down version of swtst.3i, which contains the full benchmark sea
water test case of Nordstrom et al. (1979, Table III). This input
file is a variant of swmajp.inp, which uses Pitzer's equations.
Here the input for pH (8.22) has been replaced by one for pHCl
(8.6717), the value obtained as output from running swmajp.3i and
using the thermodynamic data base of Harvie, Moller, and Weare (1984).
The output of running the present input file should include a pH of
8.22.

  Purpose: to test the pHCl-type (log activity combination) input
option.

                        References

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-SO4-
  OH-HCO3-CO3-CO2-H2O system to high ionic strengths at 25 C:
  Geochimica et Cosmochimica Acta, v. 48, p. 723-751.

Nordstrom, D. K., et al., 1979, A comparison of computerized chemical
  models for equilibrium calculations in aqueous systems, in Jenne,
  E. A., editor, Chemical Modeling in Aqueous Systems, ACS Symposium
  Series, v. 93, American Chemical Society, Washington, D.C.,
  p. 857-892.

endit.
      tempc=          25.
        rho=     1.02336    tdspkg=          0.    tdspl=          0.
        fep=       0.500    uredox=
     tolbt=          0.     toldl=          0.    tolsat=         0.
    itermx=  0
*              1    2    3    4    5    6    7    8    9   10
  iopt1-10=   -1    0    0    0    0    0    0    0    0    0
  iopg1-10=    1    0    0    0    0    0    0    0    0    0
  iopr1-10=    0    0    0    0    0    0    0    0    0    0
 iopr11-20=    0    0    0    0    0    0    0    0    0    0
  ic bl-10=    0    0    0    0    0    0    0    0    0    0
     uebal= none
     nxmod=  0
*
data file master species= na+
   switch with species=
   jflag=  3   csp= 10760.
data file master species= k+
   switch with species=
   jflag=  3   csp= 399.1
data file master species= ca++
   switch with species=
   jflag=  3   csp= 412.3
data file master species= mg++
   switch with species=
   jflag=  3   csp= 1291.8
data file master species= h+
   switch with species=
   jflag= 17   csp= -8.6722
     ion= cl-
data file master species= hco3-
   switch with species=
   jflag=  0   csp= .002022
data file master species= cl-
   switch with species=
   jflag=  3   csp= 19353.
data file master species= so4--
   switch with species=
   jflag=  3   csp= 2712.
endit.
```

The EQ3NR **input** file (**swphcl.3i**), the *pHCl* test case ("W" format):

```
EQ3NR input file name= swphcl.3i
Description= "Sea water, using pHCl as input in place of pH"
Version number= 3245   Stage number= 01
Created 10/07/90     Creator= T.J. Wolery
Revised 10/07/90     Revisor= T.J. Wolery
```

Sea water, including only the major ions. This is a considerably pared-down version of swtst.3i, which contains the full benchmark sea water test case of Nordstrom et al. (1979, Table III). This input file is a variant of swmajp.inp, which uses Pitzer's equations. Here the input for pH (8.22) has been replaced by one for pHCl (8.6717), the value obtained as output from running swmajp.3i and using the thermodynamic data base of Harvie, Moller, and Weare (1984). The output of running the present input file should include a pH of 8.22.

Purpose: to test the pHCl-type (log activity combination) input option.

References

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-SO4-OH-HCO3-CO3-CO2-H2O system to high ionic strengths at 25 C: Geochimica et Cosmochimica Acta, v. 48, p. 723-751.

Nordstrom, D. K., et al., 1979, A comparison of computerized chemical models for equilibrium calculations in aqueous systems, in Jenne, E. A., editor, Chemical Modeling in Aqueous Systems, ACS Symposium Series, v. 93, American Chemical Society, Washington, D.C., p. 857-892.

Temperature (C)	25.00	Density(gm/cm3)	1.02335

Total Dissolved Salts		mg/kg	mg/l	*not used

Electrical Balancing On		code selects	*not performed

SPECIES	BASIS SWITCH/CONSTRAINT	CONCENTRATION	UNITS OR TYPE
redox		0.5000	eh
na+		10768.	mg/kg
k+		399.10	mg/kg
ca++		412.30	mg/kg
mg++		1291.8	mg/kg
h+	cl-	-8.6722	log activity combo
hco3-		0.20220E-02	molality
cl-		19353.	mg/kg
so4--		2712.0	mg/kg

Input Solid Solutions

none			

SUPPRESSED SPECIES	(suppress,replace,augmentk,augmentg)	value

none		

OPTIONS

- SOLID SOLUTIONS -
  - * ignore solid solutions
    process hypothetical solid solutions
    process input and hypothetical solid solutions
- LOADING OF SPECIES INTO MEMORY -
  - * does nothing
    lists species loaded into memory
- ECHO DATABASE INFORMATION -
  - * does nothing
    lists all reactions
    lists reactions and log K values
    lists reactions, log K values and polynomial coef.
- LIST OF AQUEOUS SPECIES (ordering) -
  - * in order of decreasing concentration
    in same order as input file
- LIST OF AQUEOUS SPECIES (concentration limit) -
  - * all species
    only species > 10**-20 molal
    only species > 10**-12 molal
    not printed
- LIST OF AQUEOUS SPECIES (by element) -
  - * print major species
    print all species
    don't print
- MINERAL SATURATION STATES -
  - * print if affinity > -10 kcals
    print all
    don't print
- pH SCALE CONVENTION -
  - * modified NBS
    internal
    rational
- ACTIVITY COEFFICIENT OPTIONS -
  - use H-dot equation
    Davies' equation
  - * Pitzer's equations
- AUTO BASIS SWITCHING -

```
   * off
     on
 - PITZER DATABASE INFORMATION -
   * print only warnings
     print species in model and number of Pitzer coefficients
     print species in model and names of Pitzer coefficients
 - PICKUP FILE -
   * write pickup file
     don't write pickup file
 - LIST MEAN IONIC PROPERTIES -
   * don't print
     print
 - LIST AQUEOUS SPECIES, ION SIZES, AND HYDRATION NUMBERS -
   * print
     don't print
 - CONVERGENCE CRITERIA -
   * test both residual functions and correction terms
     test only residual functions
-------------------------------------------------------------------------
DEBUGGING SWITCHES (0-off, 1,2-on, default is off)
-------------------------------------------------------------------------
0  generic debugging information                                        2
0  print details of pre-Newton-Raphson iteration                        2
0  print details of Newton-Raphson iteration
0  print details of stoichiometric factors                              2
0  print details of stoichiometric factors calculation
0  write reactions on RLIST
0  list stoichiometric concentrations of master species
0  request iteration variables to be killed
-------------------------------------------------------------------------
DEVELOPMENT OPTIONS  (used for code development)
-------------------------------------------------------------------------
 none
-------------------------------------------------------------------------
TOLERANCES                    (desired values)       (defaults)
-------------------------------------------------------------------------
      residual functions     |                     |1.e-10
        correction terms     |                     |1.e-10
        saturation state     |                     |0.5
number of N-R iterations     |                     |30
-------------------------------------------------------------------------
```

The EQ3NR **output** file (**swphcl.3o**), the *pHCl* test case (beginning with the message announcing the end of Newton-Raphson iteration):

```
                    .
                    .
                    .
          (Material deleted)
                    .
                    .
                    .
Hybrid newton-raphson iteration converged in   4 steps.


                  ----- Summary of the Aqueous Phase -----

          ----- Elemental composition of the aqueous phase -----

    element        mg/l           mg/kg          moles/kg

      o       0.91080E+06    0.89001E+06    0.5562742040E+02
      ca       421.93         412.30         0.1028743949E-01
      cl       19805.         19353.         0.5458822600E+00
      h       0.11451E+06    0.11190E+06    0.1110186924E+03
      c        24.854         24.286         0.2022000000E-02
      k        408.42         399.10         0.1020760493E-01
      mg       1322.0         1291.8         0.5314955770E-01
      na       11020.         10768.         0.4683822413E+00
      s        926.41         905.26         0.2823129677E-01

          ----- elemental composition as strict basis species -----

    species             mg/l           mg/kg          moles/kg

  h2o               0.10256E+07    0.10021E+07    0.5562742040E+02
  ca++               421.93         412.30         0.1028743949E-01
  cl-                19805.         19353.         0.5458822600E+00
  h+                0.11451E+06    0.11190E+06    0.1110186924E+03
  hco3-              126.26         123.38         0.2022000000E-02
```

k+	408.42	399.10	0.1020760493E-01
mg++	1322.0	1291.8	0.5314955770E-01
na+	11020.	10768.	0.4683822413E+00
so4--	2775.4	2712.0	0.2823129677E-01

--- equivalent composition of the aqueous phase (cte balances) ---

original basis		existing basis	
species	moles/kg h2o	species	moles/kg h2o
h2o	0.5562742040E+02	h2o	0.5562742040E+02
ca++	0.1028743949E-01	ca++	0.1028743949E-01
cl-	0.5458822600E+00	cl-	0.5458822600E+00
h+	0.1110186924E+03	h+	0.1110186924E+03
hco3-	0.2022000000E-02	hco3-	0.2022000000E-02
k+	0.1020760493E-01	k+	0.1020760493E-01
mg++	0.5314955770E-01	mg++	0.5314955770E-01
na+	0.4683822413E+00	na+	0.4683822413E+00
so4--	0.2823129677E-01	so4--	0.2823129677E-01

single ion activities and activity coefficients are here defined
with respect to the modified nbs ph scale

	ph	eh	pe
modified nbs ph scale	8.2200	0.5000	8.4522E+00
rational ph scale	8.1132	0.5063	8.5590E+00

phcl = 8.6722

activity of water = 0.98198
log activity of water = -0.00790

true osmotic coefficient= 0.90273
stoichiometric osmotic coefficient= 0.90255

sum of true molalities= 1.1180579484932
sum of stoichiometric molalities= 1.1182763364436

true ionic strength= 0.6964348861048
stoichiometric ionic strength= 0.6967641025473

----- electrical balance totals -----

	equiv/kg
sigma(mz) cations =	0.6052455102E+00
sigma(mz) anions =	-0.6043365583E+00
total charge =	0.1209582068E+01
mean charge =	0.6047910342E+00
charge imbalance =	0.9089519534E-03

total charge = sigma(mz) cations + abs (sigma(mz) anions)
mean charge = 1/2 total charge

the electrical imbalance is

0.751E-01 per cent of the total charge
0.150 per cent of the mean charge
0.150 per cent of sigma(mz) cations
0.150 per cent of abs (sigma(mz) anions)

----- activity ratios of ions -----

log (act(ca++) / act(h+)xx 2) =	13.7845
log (act(cl-) x act(h+)xx 1) =	-8.6722
log (act(hco3-) x act(h+)xx 1) =	-11.2062
log (act(k+) / act(h+)xx 1) =	6.0292
log (act(mg++) / act(h+)xx 2) =	14.5358
log (act(na+) / act(h+)xx 1) =	7.7243
log (act(so4--) x act(h+)xx 2) =	-19.0097
log (act(co2(aq)) =	-4.8609
log (act(co3--) x act(h+)xx 2) =	-21.5455
log (act(oh-) x act(h+)xx 1) =	-14.0046

----- distribution of aqueous species -----

species	molal conc	log conc	log g	activity	log act
cl-	0.5459E+00	-0.2629	-0.1893	0.3530E+00	-0.4522
na+	0.4604E+00	-0.3294	-0.1663	0.3194E+00	-0.4957
mg++	0.5306E-01	-1.2752	-0.6291	0.1247E-01	-1.9043
so4--	0.2823E-01	-1.5493	-1.0204	0.2694E-02	-2.5696

ca++	0.1026E-01	-1.9887	-0.6668	0.2210E-02	-2.6555
k+	0.1021E-01	-1.9911	-0.1998	0.6444E-02	-2.1909
hco3-	0.1816E-02	-2.7409	-0.2452	0.1032E-02	-2.9862
co3--	0.8641E-04	-4.0634	-1.0420	0.7844E-05	-5.1054
mgco3(aq)	0.8292E-04	-4.0813	0.0000	0.8292E-04	-4.0813
caco3(aq)	0.2456E-04	-4.6097	0.0000	0.2456E-04	-4.6097
co2(aq)	0.1224E-04	-4.9121	0.0512	0.1378E-04	-4.8609
mgoh+	0.3362E-05	-5.4734	-0.0279	0.3152E-05	-5.5014
oh-	0.3029E-05	-5.5187	-0.2658	0.1642E-05	-5.7846
h+	0.7706E-08	-8.1132	-0.1068	0.6025E-08	-8.2200
hso4-	0.2332E-08	-8.6322	-0.1788	0.1545E-08	-8.8111

----- major aqueous species contributing to mass balances -----

aqueous species accounting for 99% or more of ca++

species	molal conc	per cent
ca++	0.1026E-01	99.76
total		99.76

aqueous species accounting for 99% or more of cl-

species	molal conc	per cent
cl-	0.5459E+00	100.00
total		100.00

aqueous species accounting for 99% or more of hco3-

species	molal conc	per cent
hco3-	0.1816E-02	89.81
co3--	0.8641E-04	4.27
mgco3(aq)	0.8292E-04	4.10
caco3(aq)	0.2456E-04	1.21
total		99.39

aqueous species accounting for 99% or more of k+

species	molal conc	per cent
k+	0.1021E-01	100.00
total		100.00

aqueous species accounting for 99% or more of mg++

species	molal conc	per cent
mg++	0.5306E-01	99.84
total		99.84

aqueous species accounting for 99% or more of na+

species	molal conc	per cent
na+	0.4684E+00	100.00
total		100.00

aqueous species accounting for 99% or more of so4--

species	molal conc	per cent
so4--	0.2823E-01	100.00
total		100.00

----- summary of aqueous redox reactions -----

couple	eh, volts	pe-	log fo2	ah, kcal
default	0.500	0.8452E+01	-16.432	11.531

----- summary of aqueous non-equilibrium non-redox reactions -----

couple	affinity, kcal

none

----- summary of stoichiometric mineral saturation states -----

(minerals with affinities .lt. -10 kcal are not listed)

mineral	log q/k	aff, kcal	state	mineral	log q/k	aff, kcal	state
anhydrite	-0.863	-1.177		aragonite	0.459	0.626	ssatd
arcanite	-5.175	-7.060		*bischofite*	-7.311	-9.975	
bloedite	-5.720	-7.803		brucite	-2.589	-3.532	
calcite	0.645	0.880	ssatd	dolomite	2.312	3.154	ssatd
epsomite	-2.648	-1.613		gaylussite	-4.476	-6.107	
glauberite	-3.542	-4.832		gypsum	-0.660	-0.901	
halite	-2.518	-3.436		*hexahydrite*	-2.886	-3.938	
kainite	-6.948	-9.479		kalicinite	-5.458	-7.447	
kieserite	-4.359	-5.947		magnesite	0.824	1.124	ssatd
mirabilite	-2.412	-3.291		*na2co3:7h2o*	-5.692	-7.765	
na4ca(so4)3:2h2o	-6.691	-9.128		nahcolite	-3.079	-4.200	
natron	-5.351	-7.300		*nesquehonite*	-1.866	-2.546	
oxychloride-mg	-5.686	-7.757		picromerite	-7.145	-9.748	
pirssonite	-4.630	-6.317		sylvite	-3.543	-4.834	
syngenite	-4.736	-6.461		thenardite	-3.274	-4.466	
thermonatrite	-6.587	-8.986					

0 approx. saturated pure minerals
0 approx. saturated end-members of specified solid solutions
0 saturated end-members of hypothetical solid solutions

4 supersaturated pure minerals
0 supersatd. end-members of specified solid solutions
0 supersatd. hypothetical solid solution phases

----- summary of gases -----

gas	fugacity	log fugacity
co2(g)	0.417753E-03	-3.37908
h2(g)	0.450200E-31	-31.13800
o2(g)	0.369830E-16	-16.43200

----- end of output -----
--- pickup file has been successfully written ---

--- reading the input file ---

--- no further input found ---

start time = 11:11 17Dec91
end time = 11:12 17Dec91

user time = 1.580
cpu time = 1.060
normal exit

8. The EQ3NR to EQ6 Connection: The Pickup File

EQ3NR creates a model of an aqueous solution. The EQ6 code may then be used to calculate models of geochemical processes in which this solution is an initial component. The two codes are constructed so that the initial model constraints on the aqueous solution are input only to EQ3NR. The user does not repeat this information in the user-created portion of the EQ6 **input** file. Instead, EQ3NR writes the necessary data on a small file called **pickup**.

EQ6 normally reads the information from the EQ3NR **pickup** file as the bottom part of the EQ6 **input** file. It can also read it directly under an option that can be specified on the EQ6 **input** file. The contents of the EQ3NR **pickup** file will be discussed in more detail in the EQ6 Theoretical Manual and User's Guide (Wolery and Daveler, 1992b). EQ3NR automatically writes a **pickup** file unless the user specifies otherwise (by setting **iopt3** = -1).

The example given in this chapter is the **pickup** file written for the sea water example presented in Chapter 7. This is given in both "W" and "D" formats; the code writes the **pickup** file in the format corresponding to the **input** file used. Discussion here will focus primarily on the example in "W" format.

The following discussion assumes that the reader is familiar with the parameters that appear on the EQ3NR **input** file (Chapter 6). The **pickup** file begins with a repetition of the original title (**utitl**). This is useful because it carries any documentation identifying the particular aqueous solution model.

The following line contains the temperature (°C) of the EQ3NR **run**, which is specified in the variable **tempci**. The EQ6 run that uses this input may actually start at a different temperature. If so, there is said to be a temperature jump. The **tempci** input allows EQ6 to recognize when this is the case and to issue a warning to the user.

The alter/suppress options follow in the same format as on the EQ3NR **input** file. In this example, no such options are specified, hence **nxmod** = 0. If any had been specified, two lines would have followed for each option, one for the **uxmod** parameter and another for the **jxmod, kxmod,** and **xlkmod** parameters.

The next five inputs (**kct, kmt, kxt, kdim,** and **kprs**) are key parameters in the scheme by which EQ6 organizes important variables. Here **kct** is the number of chemical elements in the system. The variable **kdim** is the number of total master species, which in EQ6 includes not only the aqueous master species discussed in this report, but also any mineral species in equilibrium with the aqueous solution. The variables **kmt** and **kxt** mark, respectively, the last pure mineral and last solid-solution end-member that are in the master species set. These species must have corresponding masses present in the geochemical model. This is never the case in EQ3NR, and consequently no such species can appear among the master species written on an EQ3NR **pickup** file. This condition is marked by **kmt** and **kxt** having values equal to that of **kdim**. The variable **kprs** is the number of pure mineral and solid-solution end-member species in the "physically removed system," a concept relating only to the flow-through open system model in EQ6. Hence, **kprs** must always be zero on the EQ3NR **pickup** file.

Following the parameters discussed above are the symbols of the **kct** chemical elements in the system and their masses in moles relative to one kilogram of solvent (**mte**). The calculated electrical imbalance (**electr**) is given in the same format.

The last part of the EQ3NR **pickup** file gives the names of the "run" master iteration variables (the "run" basis species), the corresponding "data file" basis species, and the values of the corresponding variables. These are the base ten logarithms of the molal concentrations of the "run" basis species. If a "run" basis species is not the same as the corresponding "data file" master species, it marks the pair of species for a basis switch.

Example of an EQ3NR **pickup** file in "W" format:

```
* pickup file written by eq3nr.3245R124x
*  supported by eqlib.3245R153
EQ3NR input file name= swmaj.3i
Description= "Sea water, major ions only"
Version number= 3245   Stage number= 01
Created 06/08/90     Creator= T.J. Wolery
Revised 06/08/90     Revisor= T.J. Wolery

  Sea water, including only the major ions. This is a considerably
pared-down version of swtst.3i, which contains the full benchmark
sea water test case of Nordstrom et al. (1979, Table III).

  Purpose: to test the code on a small problem involving a moderately
concentrated solution. The activity coefficients of the aqueous
species are calculated from the B-dot equation and related equations.

                          References

Nordstrom, D. K., et al., 1979, A comparison of computerized chemical
  models for equilibrium calculations in aqueous systems, in Jenne,
  E. A., editor, Chemical Modeling in Aqueous Systems, ACS Symposium
  Series, v. 93, American Chemical Society, Washington, D.C.,
  p. 857-892.

endit.
    tempci=  0.25000E+02
     nxmod=  0
     iopg1=  0              iopg2=  0              iopg3=  0
     iopg4=  0              iopg5=  0              iopg6=  0
     iopg7=  0              iopg8=  0              iopg9=  0
    iopg10=  0
       kct=  9                ksq= 10                kmt= 10
       kxt= 10               kdim= 10               kprs=  0
  o                 0.556274103683972E+02
  ca                0.102874395765819E-01
  cl                0.545882250556139E-00
  h                 0.111018733446243E+03
  c                 0.202200001512726E-02
  k                 0.102076049472423E-01
  mg                0.531495584904613E-01
  na                0.468382242128759E+00
  s                 0.282312979158834E-01
  electr            0.954079658003204E-03
  h2o               h2o                 0.174435898352698E+01
  ca++              ca++               -0.203301523866778E+01
  cl-               cl-                -0.280373596842018E+00
  h+                h+                 -0.820794436314582E+01
  hco3-             hco3-              -0.289243634492398E+01
  k+                k+                 -0.200081837042004E+01
  mg++              mg++               -0.139017351362375E+01
  na+               na+                -0.351762770888127E+00
  so4--             so4--              -0.187880994095832E+01
  o2(g)             o2(g)              -0.164317709391211E+02
```

Example of an EQ3NR **pickup** file in "D" format:

```
c Pickup file written by eq3nr.3245R124
c Supported by eqlib.3245R153
EQ3NR input file name= swmaj.3i
Description= "Sea water, major ions only"
Version number= 3245   Stage number= 01
Created 06/08/90     Creator= T.J. Wolery
Revised 06/08/90     Revisor= T.J. Wolery

  Sea water, including only the major ions. This is a considerably
pared-down version of swtst.3i, which contains the full benchmark
sea water test case of Nordstrom et al. (1979, Table III).

  Purpose: to test the code on a small problem involving a moderately
concentrated solution. The activity coefficients of the aqueous
species are calculated from the B-dot equation and related equations.

                        References

Nordstrom, D. K., et al., 1979, A comparison of computerized chemical
  models for equilibrium calculations in aqueous systems, in Jenne,
  E. A., editor, Chemical Modeling in Aqueous Systems, ACS Symposium
  Series, v. 93, American Chemical Society, Washington, D.C.,
  p. 857-892.

-------------------------------------------------------------------------
temperature (C)                    |      25.000
-------------------------------------------------------------------------
electrical imbalance               |       9.540796580032040E-04
-------------------------------------------------------------------------
number of aqueous master species   |      10
-------------------------------------------------------------------------
position of last pure mineral      |      10
-------------------------------------------------------------------------
position of last solid solution    |      10
-------------------------------------------------------------------------
suppressed species     (suppress,replace,augmentk,augmentg)  value
-------------------------------------------------------------------------
none                               |
-------------------------------------------------------------------------
iopg options
-------------------------------------------------------------------------
 - pH SCALE CONVENTION -
    * modified NBS
      internal
      rational
 - ACTIVITY COEFFICIENT OPTIONS -
    * use B-dot equation
      Davies' equation
      Pitzer's equations
-------------------------------------------------------------------------
elements, moles and moles aqueous
-------------------------------------------------------------------------
o                  |  5.562741836839716E+01  |  0.000000000000000E+00
ca                 |  1.028743957658190E-02  |  0.000000000000000E+00
cl                 |  5.458822605561392E-01  |  0.000000000000000E+00
h                  |  1.110187334462426E+02  |  0.000000000000000E+00
c                  |  2.022000015127257E-03  |  0.000000000000000E+00
k                  |  1.020760494724228E-02  |  0.000000000000000E+00
mg                 |  5.314955849046130E-02  |  0.000000000000000E+00
na                 |  4.683822421287592E-01  |  0.000000000000000E+00
s                  |  2.823129791588344E-02  |  0.000000000000000E+00
-------------------------------------------------------------------------
master species and logarithmic basis variables
-------------------------------------------------------------------------
h2o                |h2o                      |  1.744358983526984E+00
ca++               |ca++                     | -2.033015238667785E+00
cl-                |cl-                      | -2.803735968420180E-01
h+                 |h+                       | -8.107944363145821E+00
hco3-              |hco3-                    | -2.892436349239385E+00
k+                 |k+                       | -2.000818370420036E+00
mg++               |mg++                     | -1.390173513623747E+00
na+                |na+                      | -3.517627708881266E-01
so4--              |so4--                    | -1.878809940958323E+00
o2(g)              |o2(g)                    | -1.643177093912114E+01
-------------------------------------------------------------------------
physically removed subsystem  (solid solution, mineral, moles)
-------------------------------------------------------------------------
none                               |
-------------------------------------------------------------------------
```

9. Solving the Governing Equations

9.1. Introduction

The governing equations that apply to speciation-solubility modeling were introduced in Chapters 2, 3, and 4. The purpose of this chapter is to derive the mathematics necessary to solve them. The approach is to set up the problem in terms of n equations in n unknowns (or "iteration variables") and solve them. Technically, there are a large number of equations and corresponding unknowns to deal with. The unknowns include the concentrations of the all the species appearing in the model and their thermodynamic activity coefficients. The corresponding equations are algebraic, and these must be solved using appropriate methods.

In EQ3NR, the set of unknowns is first reduced to a relatively small set of unknowns, from which the remaining unknowns can be calculated. These are the primary iteration variables. They are defined in this code as the log concentrations of the species in the active basis set. The algebraic equations are solved by a combination of two iterative methods which are applied in sequence. The first method, called pre-Newton-Raphson optimization, has the characteristic of rapid convergence far from the solution, and slow (limiting first order) convergence near the solution. It is used primarily to get all of the primary iteration variables within an order of magnitude of the solution. The second method, a hybrid Newton-Raphson method, has the characteristic of poor convergence behavior far from the solution, and very fast (limiting second order) convergence near the solution. These methods thus complement one another. We will discuss these, as well as supplementary methods designed to aid convergence. Lastly, we will briefly discuss the subject of crash diagnostics.

9.2. The Set of Master Iteration Variables

In the EQ3NR code, the number of equations and unknowns is reduced by substituting all governing aqueous mass action equations into the mass balance and electrical balance equations. The remaining aqueous species giving rise to unknowns then comprise the relatively small active basis set. These master iteration variables reside in the vector z, which has the following structure:

$$z = \begin{bmatrix} \log x_w \\ \cdot \\ \cdot \\ \log m_{s'} \ , \ s' = 1, s_B - 1 \ , \ s \neq w \\ \cdot \\ \cdot \\ \log f_{O_2} \\ \cdot \\ \cdot \\ \log m_{s'} \ , \ s' = s_B + 1, s_Q \end{bmatrix} \tag{227}$$

The first part of this vector contains entries for the s_B strict basis species appearing in a given problem. The second part contains any auxiliary basis variables which appear in the problem and for which **jflag** $\neq 30$ (those with **jflag** $= 30$ are treated as non-basis species). The structure is further simplified if $s_Q = s_B$ (no active auxiliary basis set). For a problem in which this is the case, it is implied that the aqueous solution is in a state of complete internal (homogeneous) equilibrium. If $s_Q = s_B + 1$, complete internal equilibrium is also implied if the s_Q-th species is part of a redox couple used to define the redox state of the fluid (the **iopt1** $= 1$ option). In all other cases, partial internal disequilibrium is implied.

We will first look at the aqueous mass action equations that are to be eliminated. Here s'' denotes a non-basis species and r denotes the corresponding reaction. As index labels, these are related according to:

$$r = s'' - s_B \tag{228}$$

This reflects the fact that strict basis species appear first in the list of all aqueous species and that they have no associated reactions. A mass action equations takes the following logarithmic form:

$$\log K_r = b_{wr}(\log x_w + \log \lambda_w) + b_{s_B r}\log f_{O_2} + b_{s'' r}(\log m_{s''} + \log \gamma_{s''}) + \sum_{\substack{s'=1 \\ s' \neq w, s_B}}^{s_Q} b_{s' r}(\log m_{s'} + \log \gamma_{s'}) \tag{229}$$

where K_r is the thermodynamic equilibrium constant for the reaction b_{sr} is the reaction coefficient for the s-th species, λ_w is the activity coefficient of water, and $\gamma_s (s \neq w)$ is the molal activity coefficient of the s-th species. Note that s' implies a basis species, and that s'' denotes the only non-basis species appearing in the reaction. This equation can be rearranged to give:

$$\log m_{s''} = \frac{\log K_r}{b_{s'' r}} - \log \gamma_{s''} - \frac{b_{wr}}{b_{s'' r}}(\log x_w + \log \lambda_w) - \frac{b_{s_B r}}{b_{s'' r}}\log f_{O_2} - \sum_{\substack{s'=1 \\ s' \neq w, s_B}}^{s_Q} \frac{b_{s' r}}{b_{s'' r}}(\log m_{s'} + \log \gamma_{s'}) \tag{230}$$

Recall the relation:

$$\frac{dx}{d\log x} = 2.303x \tag{231}$$

It follows that:

$$\frac{dm_{s''}}{dx} = 2.303 m_{s''} \frac{dlogm_{s''}}{dx} \tag{232}$$

Treating the activity coefficients as constants, this can be used to show that:

$$\frac{\partial m_{s''}}{\partial log x_w} = -2.303 m_{s''} \frac{b_{wr}}{b_{s''r}} \tag{233}$$

$$\frac{\partial m_{s''}}{\partial log f_{O_2}} = -2.303 m_{s''} \frac{b_{s_B r}}{b_{s''r}} \tag{234}$$

$$\frac{\partial m_{s''}}{\partial log m_{s'}} = -2.303 m_{s''} \frac{b_{s'r}}{b_{s''r}} \quad , \quad s' = 1, s_Q \quad , \quad s' \neq w, \; s_B \tag{235}$$

These relations will be used later in this chapter in deriving the Jacobian matrix elements corresponding to the mass and charge balance residual functions. This matrix is used in Newton-Raphson iteration.

Certain factors will appear repeatedly in some of the derivations below and will be given special symbols. In the EQ3NR code, these parameters are themselves evaluated before the calculation of the Jacobian matrix elements in which they appear. This is done to avoid repetitive arithmetic in the code. These are defined as follows:

$$H_{s'r} = \frac{m_{s''} u_{s''s'}}{b_{s''r}} \quad , \quad s' = 1, s_Q \quad , \quad s' \neq w \tag{236}$$

$$H_{zr} = \frac{m_{s''} z_{s''}}{b_{s''r}} \tag{237}$$

All of the so-called alternative constraints involve equations which are written only in terms of the concentrations (or activities) of species in the active basis set. Therefore, it is not necessary to make any substitutions of the sort noted above in dealing with the mass balance equations and the charge balance equation.

9.3. Expanding the System from the Set of Master Iteration Variables

If one knows the vector $\underline{z}$, one may "expand the system" by computing the concentrations of all non-basis species (all species not in the active basis set; this includes any auxiliary basis species with **jflag** values of 30) and the activity coefficients of all species in solution. The $\underline{z}$ vector uniquely defines all system properties. However, the process of expanding the system is not exactly straightforward. In order to calculate the concentrations of the non-basis species, one must evaluate the corresponding mass action equations. The activity coefficients appear in these equations. Hence, the activity coefficients must be evaluated first. However, the activity coefficients depend in general on the concentrations of all solute species, both basis and non-basis. So to deal with these, one must compute the concentrations of the non-basis species first.

This creates a problem analogous to the old puzzle, "Which came first, the chicken or the egg?" In previous versions of EQ3/6 (e.g., Wolery, 1983), this problem was overcome by treating the ionic strength as a master iteration variable. The equation defining the ionic strength has the same form as a mass balance equation, and this equation was treated in like form. However, this only works if the activity coefficient model depends only on the ionic strength, not the specific composition of the solution. Thus, this approach works for the Davies equation and the B-dot equation, but not for Pitzer's equations or any other set of equations likely to be valid in concentrated solutions.

The concentrations of non-basis species may vary over many orders of magnitude. The activity coefficients of aqueous species generally vary over about two orders of magnitude or less. The approach taken in EQ3/6 is to start by computing a set of reasonable values for the activity coefficients, then hold these constant until the concentrations of the non-basis species become reasonably stable. The activity coefficients are then updated. In the pre-Newton-Raphson optimization algorithm, the computed concentrations of the basis and non-basis species are typically adjusted 3-7 times before the activity coefficients are recalculated. In the hybrid Newton-Raphson method, they are recalculated between each Newton-Raphson step (it is because of this treatment that we refer to our usage of the Newton-Raphson method as a hybrid). In either case, the system is expanded by first calculating the new concentrations of the non-basis species, using the existing values of the activity coefficients. The activity coefficients are then recalculated. An exception to this order occurs when starting values are constructed (see below).

One could view the expansion itself as an iterative process. One could recalculate the concentrations of the non-basis species, recalculate the activity coefficients, and then repeat the process one or more times. In the context of the pre-Newton-Raphson optimization method, this makes little sense because this method is only used to get in the neighborhood of the solution. The merits of a single update, a double update, and a multiple update (repeating the process until a convergence tolerance is satisfied) were examined in the context of the hybrid Newton-Raphson method. Significant differences in performance were only observed in the case of highly concentrated electrolyte solutions. Thus, in the case of less concentrated solutions, the single update method was best because it gave the same performance for the lowest cost. In the case of the more concentrated solutions, it was found that the single update also gave the best performance, followed by the multiple update method. In such solutions, the double update method often led to failure to converge. The method which has been adopted, therefore, is the single update method.

9.4. Beginning the Process: Computing Starting Values

The whole process must begin by assigning starting values. This is done in module **arrset.f**. Initially, this is done as follows. For every basis species with a **jflag** value of less than 15, either the total or free molality is known. The concentration of each such individual basis species is assigned this corresponding value. This value is an upper bound if the corresponding concentration is the total molality, and the actual value if it is a free molality. For species having a **jflag** value of 16, the log activity is known. The concentration of each such species is assigned a value equal to the thermodynamic activity, thus assuming that the activity coefficient has unit value. All other basis species are initially assigned a value of 1×10^{-7} molal. All non-basis species at this point have assigned concentrations of zero.

The charge imbalance is computed. The functions Σm and the ionic strength are then estimated. The estimate of the ionic strength at this point includes a term in which the computed charge imbalance is treated as though it were due to an unmeasured monovalent ion. The mole fraction of water is computed from the value of Σm. Then the activity coefficients are computed. The concentrations of non-basis species are taken as zero until the code enters the pre-Newton-Raphson optimization stage.

Assuming that the concentration of a basis species is equal to its total concentration may or may not be a good approximation. If it turns out to be a very poor approximation, then the first estimate of the concentration of at least one non-basis species will have to be large, typically a few to a few tens of orders of magnitude greater than the limit on its actual concentration imposed by the corresponding mass balance constraint. This first estimate is very often quite large in an absolute as well as a relative sense, often on the order of 10^{+10} to 10^{+60} molal. It is critical not to compute functions such Σm, the ionic strength, and the activity coefficients until the concentrations for such species have been brought down to physically realistic values.

9.5. Methods to Aid Convergence

Several techniques are used in EQ3NR to aid convergence, both in pre-Newton-Raphson optimization and hybrid Newton-Raphson iteration. These are:

- Use of logarithmic iteration variables.
- Under-relaxation techniques.
- Automatic and user-specified basis switching.

We have not found it necessary to employ other methods, such as the "curve-crawler" technique discussed by Crerar (1975).

The physical quantities that correspond to the iteration variables are intrinsically positive. Use of logarithmic iteration variables restricts the generated values to the physically reasonable range. Also, logarithmic corrections are effectively relative corrections to the corresponding physical quantities. Recall that $d\ log\ x/dx = 1/(2.303\ x)$. It follows that:

$$\Delta log\, x \approx \frac{\Delta x}{2.303x} \tag{238}$$

Because of this, effective under-relaxation techniques are especially easy to implement when using logarithmic iteration variables.

Under-relaxation is the technique of judiciously reducing the magnitude of the computed correction terms. Assume that the unmodified method involves adding a correction term vector (δ_k), where k is the iteration number. This is typical in Newton-Raphson iteration. The new vector of master iteration variables is obtained thusly:

$$z_{k+1} = z_k + \delta_k \tag{239}$$

If the new vector of master iteration variables is obtained instead by evaluating some set of corresponding equations not in this format, one can still utilize under-relaxation by defining a correction term vector as follows:

$$\underline{\delta}_k = z_{k+1} - z_k \tag{240}$$

Global under-relaxation is effected by replacing the correction equation given above by:

$$z_{i+1} = z_i + \kappa \underline{\delta}_i \tag{241}$$

where κ is a positive number less than one. Non-global under-relaxation is also possible. This does not involve the use of an under-relaxation factor. Rather it involves truncating the magnitudes of individual correction terms to satisfy specified limits, which may be different depending on the species involved and on the direction of change.

There are several methods of applying global under-relaxation. EQ3NR uses two relatively simple ones in making Newton-Raphson steps. The first of these places a limit on the element of the correction term vector having the largest magnitude:

$$\kappa = \frac{\delta'}{\delta_{max}} \tag{242}$$

where δ' is the imposed limit and δ_{max} is the max norm of δ. In a Newton-Raphson iteration step (which occurs in the EQLIB module **nrstep.f**), this imposed limit is represented by the variable **screwd**. In EQ3NR, this is set in the main program (module **eq3nr.f**) at a value of 2.0. Besides aiding convergence, this method causes divergence, when it does occur, to occur more slowly. In such cases, it helps to yield useful information about the cause of divergence.

The other global under-relaxation method is applied for only the first 8 iterations. The under-relaxation factor is cut in half if the residual vector max norm β_{max} exceeds the value of the variable **screwn**. Initially, κ is set to a value of unity; when the current method of under-relaxation is applied, this factor may have been reduced as a result of applying the method described above. In EQ3NR, **screwn** is set to 0.5 in the main program (module **eq3nr.f**).

Some degree of non-global under-relaxation is also employed in pre-Newton-Raphson optimization. This optimization function takes place in EQ3NR in module **arrset.f**. Here under-relaxation is effected by imposing truncation limits on changes for individual master variables. The master variables for species constrained by mass balance equations are not permitted to decrease by more than 20.0 log units in a given step. A master variable constrained by the charge balance equation may not change by more than 2.0 log units.

Some truncation limits also apply to the activity coefficients and the functions Σm and the ionic strength. These limits are applied during both pre-Newton-Raphson optimization and hybrid Newton-Raphson iteration. These limits are defined in the variable **chgfac**, which is in the calling sequence of the EQLIB module **ngcadv.f**. The value of this variable is set in the calling modules, and is usually scaled inversely with the value of Σm. Values range from 1.3 to 100.

If automatic basis switching is turned on (**iopt2** = 1), EQ3NR (in the module **arrset.f**) will attempt to improve the starting values by means of basis switching. The methodology here is quite simple. Consider the case of dissolved aluminum. The data file basis species is Al^{3+}. At low temperature, typically low values of dissolved aluminum, and moderate to high *pH*, the mass balance is typically very strongly dominated by the species $Al(OH)_4^-$. The concentration of Al^{3+} is many orders of magnitude below the concentration of this species. If one assumes that the concentration of this species is instead essentially equal to the total concentration, the computed concentration of $Al(OH)_4^-$ may be something on the order of 10^{20}-10^{40} molal. The value of the corresponding residual function will be similarly extremely large. On the other hand, when $Al(OH)_4^-$ is in the basis set, the initial assumption is that its concentration is equal to the total concentration, and the computed concentration of Al^{3+} is an appropriately much smaller number.

In the present version of EQ3NR, the total concentration quantity associated with a mass balance is redefined in terms of the new basis species. Assuming this quantity is expressed as molality, this change has no numerical significance in the above example. However, if a species to be switched into the basis set contributes to the original total concentration by a factor different from that of the original basis species, the difference is quite significant. For example, if $Al_{13}O_4(OH)_{24}^{7+}$ is switched into the basis set in place of Al^{3+}, then the associated total concentration must be redefined as:

$$m_{T,Al_{13}O_4(OH)_{24}^{7+}} = \frac{m_{T,Al^{3+}}}{13} \tag{243}$$

Automatic basis switching is accomplished in a loop structure. More than one switch may be done each time through the loop. After this, the activity coefficients are recomputed (again assuming that the concentrations of non-basis species are zero), the residual functions are recomputed and more switches may be made. In the process, some switches may be undone by later ones. For example, $Al_{13}O_4(OH)_{24}^{7+}$ may first be switched into the basis set in place of Al^{3+}, and then $Al(OH)_4^-$ in place of, $Al_{13}O_4(OH)_{24}^{7+}$. When one switch replaces another, the original switch is first undone. In this example, Al^{3+} is switched back into the basis in place of $Al_{13}O_4(OH)_{24}^{7+}$; $Al(OH)_4^-$ is then switched into the basis in place of Al^{3+}. This loop continues until there are no candidates for basis switching or the loop has been passed through **nlopmx** times. This variable is currently set to 12 in a data statement in **arrset.f**.

To be a candidate for automatic basis switching, a species must have a computed concentration ten times that of the corresponding basis species. Furthermore, it can not already be in the basis set. A data file basis species which has been switched out of the basis set can only be brought back into the basis set by undoing an earlier switch. The involvement of a basis species in the input constraint associated with another basis species may prevent switching it out of the basis set. For example, if the input constraint for H^+ is a value for the *pHCl* function, then Cl^- is locked into the basis set. Conflicts may arise in candidate basis switches. For example, the same species

could dominate more than one mass balance. It is then switched into the basis so as to reduce the affected mass balance residual which has the highest value.

The user may specify certain basis switches on the **input** file (see Chapter 5). These switches are executed prior to entry to module **arrset.f** and the construction of the initial starting estimates. If certain switches really need to be made, it is more efficient to make them in this manner. If the software is run on a machine with a small floating point exponent range, such as a VAX system lacking the G_FLOATING option, the code may stop execution because of a floating point overflow (in the computation of one of the elements of the residual function vector β) unless such switches are directed on the **input** file. In such a situation, one may have to guess which switches are necessary to avoid such overflow.

If automatic basis switching is turned off (**iopt2** = 0), the code will proceed directly from having made the initial starting estimates into pre-Newton-Raphson optimization. If it is turned on and the user has directed certain switches to be made on the **input** file, it may undo one or more of those switches, as well as make additional switches.

9.6. The Pre-Newton-Raphson Optimization Algorithm

After any automatic basis switching is completed, **arrset.f** uses an optimization algorithm. This process occurs in a loop structure, the times through which are known as *passes*. At the end of a pass, the activity coefficients are recomputed. If the concentration of an ion is to be adjusted to satisfy electrical balance, this adjustment is also recomputed at this point. Within each pass is another loop structure, the times through which are called *cycles*. Here, adjustments are made to the concentrations of the basis species (other than one which is constrained to satisfy electrical balance). A pass is completed after some number of cycles. The cycles within a pass terminate if some rather rough convergence criteria are satisfied, or if the maximum number of cycles in a pass have been completed. This is determined by the variable **ncylim**, which is currently set to 15 in a data statement in **arrset.f**. The passes terminate if rough convergence criteria applying to both the cycles and passes are satisfied, or if the maximum number of passes has been completed. This is determined by the variable **nplim**, which is currently set to 7 in a data statement in **arrset.f**.

The cycle algorithm is applied only to basis species which are constrained by mass balances. It is an example of what is sometimes called a "continued fraction" method. A variation on this approach (Wolery and Walters, 1975) was in fact the principal method used to solve speciation-solubility problems in a an early version of what is now EQ3NR. The derivation to be given here is different than that previously given elsewhere, and includes an important modification that has apparently not been previously noted.

Consider the case of dissolved aluminum. The total dissolved aluminum is expressed as total Al^{3+}. The normalized mass balance residual is:

$$\beta_{Al^{3+}} = \frac{m_{T,calc,Al^{3+}} - m_{T,Al^{3+}}}{m_{T,Al^{3+}}} \tag{244}$$

where: $m_{T,calc,Al^{3+}}$ is the total concentration of Al^{3+} as calculated from a mass balance expression, using the current estimated values of the concentrations of the basis species and estimates of the concentrations of non-basis species as calculated from the associated mass action equations, using in these the current estimated values of the concentrations of the basis species. In contrast, $m_{T,Al^{3+}}$ is one of the model constraints.

We will assume, for the moment, that Al^{3+} dominates this calculated mass balance. We may express this by writing:

$$m_{Al^{3+}} \approx m_{T,calc,Al^{3+}} \tag{245}$$

We may then write:

$$\beta_{Al^{3+}} \approx \frac{m_{Al^{3+}} - m_{T,Al^{3+}}}{m_{T,Al^{3+}}} \tag{246}$$

We can rearrange this to:

$$\frac{m_{Al^{3+}}}{\beta_{Al^{3+}} + 1} \approx m_{T,Al^{3+}} \tag{247}$$

We take the current iteration in the cycle to be the k-th. Applying the above equation to this iteration, we may write:

$$\frac{m_{Al^{3+},k}}{\beta_{Al^{3+},k} + 1} \approx m_{T,Al^{3+}} \tag{248}$$

Similar, applying it to the next iteration (the $k+1$-th) gives:

$$\frac{m_{Al^{3+},k+1}}{\beta_{Al^{3+},k+1} + 1} \approx m_{T,Al^{3+}} \tag{249}$$

Combining these equations then gives:

$$m_{Al^{3+},k+1} \approx \frac{m_{Al^{3+},k}(\beta_{Al^{3+},k+1} + 1)}{\beta_{Al^{3+},k} + 1} \tag{250}$$

We would like the residual function to approach zero quickly. This desire can be written as:

$$\beta_{Al^{3+},k+1} \approx 0 \tag{251}$$

Substitution of this into the above equation gives the following the iteration equation:

$$m_{Al^{3+},k+1} = \frac{m_{Al^{3+},k}}{\beta_{Al^{3+},k} + 1} \tag{252}$$

In a dilute, acid solution, the species Al^{3+} will indeed dominate its own mass balance. The starting assumption based on this should be a good one. Also, this species should dominate the calculated mass balances during the iteration process. We would expect the above iteration equation to work well. However, in solutions of moderate to high *pH*, the species $Al(OH)_4^-$ dominates the mass balance of Al^{3+}. What happens then? Note that we could do a basis switch, replacing Al^{3+} with $Al(OH)_4^-$. The above equations would then apply to $Al(OH)_4^-$ and all should again work well.

If we do not make this basis switch, we would still assume that $Al(OH)_4^-$ dominates the calculated mass balances. Following the previous approach, we are led to an equation of the form:

$$m_{Al(OH)_4^-,k+1} = \frac{m_{Al(OH)_4^-,k}}{\beta_{Al^{3+},k} + 1} \tag{253}$$

This is almost what we would have if we had made the basis switch. However, in this case, the normalized residual is still defined in terms of Al^{3+}, not $Al(OH)_4^-$. However, we can not directly use this equation, because it is written in terms of molalities of $Al(OH)_4^-$, not Al^{3+}. However, we can convert it into a form in terms of molalities of Al^{3+}. The two species are related by the reaction:

$$Al(OH)_4^- + 4H^+ = Al^{3+} + 4H_2O_{(l)} \tag{254}$$

The corresponding mass action equation is:

$$K_{Al(OH)_4^-} = \frac{m_{Al^{3+}} \gamma_{Al^{3+}} x_w^4 \lambda_w^4}{m_{Al(OH)_4^-} \gamma_{Al(OH)_4^-} m_{H^+}^4 \gamma_{H^+}^4} \tag{255}$$

Let us consider all activity coefficients to be fixed, as well as the concentrations of other basis species appearing in this relation. Then the concentrations of the two aluminum species of interest satisfy the following proportionality:

$$m_{Al^{3+}} \propto m_{Al(OH)_4^-} \tag{256}$$

Presuming that the proportionality constant does not change significantly allows us to use this relation to obtain the same iteration equation that we had before:

$$m_{Al^{3+},k+1} = \frac{m_{Al^{3+},k}}{\beta_{Al^{3+},k}+1} \tag{257}$$

The only significant difference in the iteration process in this case (versus either that in which Al^{3+} is dominant or that in which $Al(OH)_4^-$ is dominant but switched into the basis set) is that the starting estimate is not so good. In fact, it might be very bad, off by many orders of magnitude. However, it turns out that this algorithm is very good for quickly getting to about the right order of magnitude, even if the starting estimate is off (high) by several tens of orders of magnitude. On the other hand, it is not so efficient in a close neighborhood of the solution. This makes it a good complement to Newton-Raphson iteration, which is very efficient near the solution, but which often fails to converge at all if the starting estimates are far from the solution.

Now suppose the complex $Al_{13}O_4(OH)_{24}^{7+}$ dominates the calculated mass balance of Al^{3+}. Again, we do not make a basis switch. This then leads us to a result of the form:

$$m_{Al_{13}O_4(OH)_{24}^{7+},k+1} = \frac{13m_{Al_{13}O_4(OH)_{24}^{7+},k}}{\beta_{Al^{3+},k}+1} \tag{258}$$

The species $Al_{13}O_4(OH)_{24}^{7+}$ and Al^{3+} are related by the reaction:

$$Al_{13}O_4(OH)_{24}^{7+} + 32H^+ = 13Al^{3+} + 28H_2O_{(l)} \tag{259}$$

The corresponding mass action equation is:

$$K_{Al_{13}O_4(OH)_{24}^{7+}} = \frac{m_{Al^{3+}}^{13}\gamma_{Al^{3+}}^{13}x_w^{28}\lambda_w^{28}}{m_{Al_{13}O_4(OH)_{24}^{7+}}\gamma_{Al_{13}O_4(OH)_{24}^{7+}}m_{H^+}^{32}\gamma_{H^+}^{32}} \tag{260}$$

Using the same assumptions as before leads to the following proportionality:

$$m_{Al^{3+}}^{13} \propto m_{Al_{13}O_4(OH)_{24}^{7+}} \tag{261}$$

Using this as before leads to the following iteration equation:

$$m_{Al^{3+},k+1} = \frac{13^{\frac{1}{13}}m_{Al^{3+},k}}{(\beta_{Al^{3+},k}+1)^{\frac{1}{13}}} \tag{262}$$

This differs from the previous results in two ways. First, an exponent now appears on the $(\beta + 1)$ term. This exponent is the factor expressing the stoichiometric equivalence of the basis species corresponding to a mass balance expression to the species which dominates the calculated mass balance. Also, a factor appears in the numerator on the right hand side which is the inverse of the same exponent with a matching exponent This is the general case. These "new" elements of the equation did not explicitly appear in the previous results because the stoichiometric equivalence happened to be unity.

The exponent on the $(\beta + 1)$ term is critical to the success of this method. This is because it has a large effect on the exponent of the resultant calculated concentration of the basis species. When one starts the iteration process, the value of β may initially be something like 10^{+60}. Use of the iteration equation ignoring this exponent would result in the calculated concentration of the basis species being lowered by 60 orders of magnitude. The effect of the exponent on the $(\beta + 1)$ is to cause it to be lowered only about 4.6 orders of magnitude. The effect of ignoring this exponent *on the remainder of the calculation would be wild oscillation.*

The effect of the factor in the numerator is less extreme. It does not affect the resulting order of magnitude. For example, $1^{\frac{1}{1}} = 1$, $2^{\frac{1}{2}} \approx 1.41$, $3^{\frac{1}{3}} \approx 1.44$, $\cdot^{\frac{1}{-}} \approx 1.41$, and $5^{\frac{1}{5}} \approx 1.38$. In the case shown above, $13^{\frac{1}{13}} \approx 1.22$. The absence of this factor doesn't have much effect if all one is trying to do is get within about an order of magnitude of the solution before switching to another algorithm. However, it would cause convergence to fail in a close neighborhood of the solution if one attempted to use this algorithm to obtain a final solution.

Not all basis species are constrained by mass balance relations. If a species is constrained by a value for its log activity, the concentration is simply estimated from this value using the current value for the corresponding activity coefficient:

$$log m_{i,k+1} = log a_i - log \gamma_i \tag{263}$$

The concentration of such a species can not change during a pass, because activity coefficients are only recalculated as the end of a pass.

A basis species whose concentration is to be adjusted to satisfy electrical balance has its concentration recalculated in a cycle after the concentrations of all other basis species have been recalculated. This will be discussed below.

The concentrations of basis species which are constrained by any other types of constraints are calculated simultaneously by solving a matrix equation. This is the case for $O_{2(g)}$ when an *Eh* or *pe* value is input, for cases in which heterogeneous or homogeneous equilibria must be satisfied, and cases in which a combination activity function such as *pHCl* is utilized. It is not always strictly necessary to make these calculations simultaneously, but it is more convenient to always do it this way than to deal otherwise with those cases which would allow complete or partial solution by a sequence of individual calculations (whose order would have to be determined in each individual case).

For example, suppose dissolved calcium is constrained to satisfy equilibrium with calcite and bicarbonate is constrained to satisfy a specified fugacity of carbon dioxide. We will assume that chloride is constrained by a total concentration and that the hydrogen is constrained by an input value of pH. The respective relevant governing equations for Ca^{2+} and H^+ are:

$$\log K_{calcite} = \log m_{Ca^{2+}} + \log \gamma_{Ca^{2+}} + \log m_{HCO_3^-} + \log \gamma_{HCO_3^-} - \log m_{H^+} - \log \gamma_{H^+} \quad (264)$$

$$\log K_{CO_{2(g)}} = \log m_{HCO_3^-} + \log \gamma_{HCO_3^-} + \log m_{H^+} + \log \gamma_{H^+} - \log f_{CO_2} - \log x_w - \log \lambda_w \quad (265)$$

These may be rearranged to give:

$$\log m_{Ca^{2+}} + \log m_{HCO_3^-} = \log K_{calcite} - \log \gamma_{Ca^{2+}} - \log \gamma_{HCO_3^-} + \log m_{H^+} + \log \gamma_{H^+} \quad (266)$$

$$\log m_{HCO_3^-} - \log x_w = \log K_{CO_{2(g)}} - \log \gamma_{HCO_3^-} - \log m_{H^+} - \log \gamma_{H^+} + \log f_{CO_2} + \log \lambda_w \quad (267)$$

where the variables treated as the unknowns are on the left hand sides. As all these concentration values pertain to the $k+1$-th iteration, they could be so marked in the rearranged equations (but we leave them out for clarity). The mole fraction of water may also be adjusted as part of the process, using the approximate relation:

$$\log x_{w,k+1} = \log x_{w,k} + \sum_{s' \in S} \frac{\partial \log x_w}{\partial \log m_{s'}} (\log m_{s',k+1} - \log m_{s',k}) \quad (268)$$

where S is the set of solute basis species whose concentrations must be solved for in this manner. This is a Taylor's series truncated to first order. For notational convenience, we make the following definition:

$$W_{s'} = \frac{\partial \log x_w}{\partial \log m_{s'}} , \qquad s' \neq w, s_B \quad (269)$$

Then the equation in the present example may be written as:

$$\log x_w - W_{Ca^{2+}} \log m_{Ca^{2+}} - W_{HCO_3^-} \log m_{HCO_3^-}$$

$$= \log x_{w,k} - W_{Ca^{2+}} \log m_{Ca^{2+},k} - W_{HCO_3^-} \log m_{HCO_3^-,k} \quad (270)$$

where again all the variables treated as unknowns are on the left hand side. For notational consistency, we have dropped the "$k+1$" subscripts.

We now evaluate the necessary partial derivatives. The mole fraction of water must be expressed as a function of the concentrations of the basis species other than water. We first write this in the form:

$$\log x_w = \log\left(\frac{\Omega}{\Omega + \sum_{\substack{s'=1 \\ s' \neq w}}^{s_Q} m_{s'} + \sum_{r=1}^{r_T} m_{s''}}\right) \tag{271}$$

where s'' denotes the non-basis species associated with the r-th aqueous reaction. Partial differentiation leads to the following intermediate result:

$$W_{s'} = -\frac{x_w}{\Omega}\left(m_{s'} - \sum_{r}^{r_T} \frac{b_{s'r} m_{s''}}{b_{s''r}} - W_{s'} \sum_{r}^{r_T} \frac{b_{wr} m_{s''}}{b_{s''r}}\right) \tag{272}$$

where s' is a basis species other than water. Rearranging then gives the final result:

$$W_{s'} = \frac{-\frac{x_w}{\Omega}\left(m_{s'} - \sum_{r}^{r_T} \frac{b_{s'r} m_{s''}}{b_{s''r}}\right)}{\left(1 - \frac{x_w}{\Omega} \sum_{r}^{r_T} \frac{b_{wr} m_{s''}}{b_{s''r}}\right)} \tag{273}$$

Technically, these partial derivatives should be evaluated using data corresponding to the k-th iteration. However, as the method is not exact anyway and the mole fraction of water does not really vary much in the process, this is not critical.

In the above example, we have three equations in three unknowns. These equations are linear in the four log concentration variables; hence, they can be solved simultaneously by solving a corresponding matrix equation. This takes the form:

$$\begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & -1 \\ -W_{Ca^{2+}} & -W_{HCO_3^-} & 1 \end{bmatrix} \begin{bmatrix} \log m_{Ca^{2+}} \\ \log m_{HCO_3^-} \\ \log x_w \end{bmatrix} = \begin{bmatrix} R_{Ca^{2+}} \\ R_{HCO_3^-} \\ R_w \end{bmatrix} \tag{274}$$

where the elements in the right hand side vector are equal to the expressions on the right hand side of eqs (266), (267), and (270). Matrix equations of this form are evaluated after new values have been calculated for the concentrations of the other basis species.

At the end of a cycle, a full set of residual functions is computed. This includes the β array and its max norm, β_{max}. The only non-zero elements of β at the end of a cycle are those which pertain to mass and charge balance constraints. The elements of this vector corresponding to the former have been previously introduced. In the case of the latter, the relevant equation is:

$$\beta_{s_z} = \frac{\alpha_{s_z}}{\sum_c z_c m_c + \sum_a |z_a| m_a} \tag{275}$$

where s_z is the basis species so constrained, c denotes cations and a anions, and:

$$\alpha_{s_z} = \sum_c z_c m_c + \sum_a z_a m_a \tag{276}$$

The concentration of this species is then adjusted so as to make these residuals zero:

$$m_{s_z, k+1} = m_{s_z, k} - \frac{\alpha_{s_z, k}}{|z_{s_z}|} \tag{277}$$

A pass ends when one of the following occurs:

- The non-zero (mass balance) elements of β satisfy a loose convergence test (all fall in the range -10% to +50%).
- The maximum number of cycles per pass have been completed.
- A convergence function β_{func} (**betfnc**) indicates that iteration in the present cycle is diverging. This convergence function will be discussed later in this chapter.

At the end of a pass, the Σm function and the ionic strength are recalculated, the mole fraction of water is recalculated, and the activity coefficients are recalculated. The code defines residual functions based on the magnitude of the changes in Σm, the ionic strength, and the activity coefficients from the values pertaining to the previous pass. The sequence of passes is stops when one of the following occurs:

- The residuals defined for Σm, the ionic strength, and the activity coefficients satisfy a loose convergence test (all less than or equal to 0.1).
- The maximum number of passes have been completed.

The optimization is deemed successful if both sets of loose convergence tolerances are satisfied. The code will then execute hybrid Newton-Raphson iteration. If optimization is not successful, the code checks to see if any equilibrium constraints appear to imply unrealistically high solute concentrations. If this is the case, the code will not attempt hybrid Newton-Raphson iteration and a message to this effect is written to the screen and **output** files. Otherwise, the code will execute hybrid Newton-Raphson iteration, which often succeeds even when the optimization step fails to satisfy its own convergence tolerances.

The cycle algorithm discussed in this section is an example of a first order method. This means that in a close neighborhood of the solution, one has that:

$$\delta_{i,k+1} = \sum_{j=1}^{n} p_{ij}\delta_{jk} \tag{278}$$

where n is the number of iteration variables and the p_{ij} are constants.

9.7. The Newton-Raphson Method

The Newton-Raphson method is a well-known iterative technique for solving non-linear systems of algebraic equations (see for example Van Zeggeren and Storey, 1970; Carnahan, Luther, and Wilkes, 1969; or any introductory text on numerical methods). We will not discuss the derivation of the method here, only its application. Given a set n governing equations and n unknowns (represented by a vector $\underline{z}$ of iteration variables), one may construct a set of residual functions (represented by the vector $\underline{\alpha}$), each member of which has a value of zero when the n equations are satisfied. Both $\underline{z}$ and $\underline{\alpha}$ are of length n.

A simple example will illustrate this. Suppose we wish to solve the general quadratic equation:

$$ax^2 + bx + c = 0 \tag{279}$$

This is a case in which $n = 1$. Here a, b, and c are given coefficients, and x is the only unknown. The residual function can be defined as:

$$\alpha = ax^2 + bx + c \tag{280}$$

Other definitions are possible, however. The only requirement is that they take on a value of zero when the governing equation is satisfied. We note here that the choice of definition may affect the convergence behavior.

Let k be the number of iterations, such that $\underline{z}_k$ and $\underline{\alpha}_k$ are the iteration variable and residual function vectors on the k-th iteration. Let $\underline{z}_0$ represent the set of starting estimates. An iteration step is made by calculating $\underline{z}_{k+1}$ from $\underline{z}_k$. The Newton-Raphson method does this by computing a vector of correction terms, δ, by solving the matrix equation:

$$\underline{J}\underline{\delta} = -\underline{\alpha} \tag{281}$$

Here $\underline{J}$ is the Jacobian matrix, defined by:

$$\underline{J} = \left(\frac{\partial \alpha_i}{\partial z_j}\right) \tag{282}$$

where i and j are the matrix coordinates. In our example, this becomes:

$$J = 2ax + b \tag{283}$$

The correction term is then applied:

$$z_{k+1} = z_k + \delta_k \tag{284}$$

If the iteration converges, all elements of both α and δ approach zero. It is useful to define another residual function vector β which is identical to α, except that some elements may be normalized to provide a better measure of convergence. It is then convenient to define β_{max} and δ_{max} as the largest absolute values of the elements of β and δ, respectively. Both β_{max} and δ_{max} may then be used in tests to determine if the iteration has converged satisfactorily.

A useful measure of how well convergence is proceeding may also be constructed. The Newton-Raphson method is a second order method.This means that in a close neighborhood of the solution, one has that:

$$\delta_{i,k+1} = \sum_{j=1}^{n} p_{ij}\delta_{jk}^2 \tag{285}$$

where the p_{ij} are constants. In practice, there is usually no attempt to actually evaluate them. The significance of this is that in a close neighborhood of the solution, $\delta_{max,\,k+1}$ should be much less than $\delta_{max,\,k}$. The function δ_{func} (the variable **delfnc**) is defined:

$$\delta_{func,\,k+1} = 1 - \left(\frac{\delta_{max,\,k+1}}{\delta_{max,\,k}}\right) \tag{286}$$

may therefore be expected to approach (from below) a value of unity if the iteration is converging very rapidly (as theory suggests when the cross terms describing the evolution of the correction vector are small; i. e., the p_{ij} are small for the case $i \neq j$). Convergence to a lesser value, say $\approx$ 0.72 instead of $\approx$0.99 is not unknown. This may imply non-negligible cross terms or an error in writing the Jacobian matrix. It also may result from modifications to the basic Newton-Raphson method, such as we have introduced by updating the activity coefficients between Newton-Raphson steps. The function, β_{func} (the variable **betfnc**) is defined similarly:

$$\beta_{func,\,k+1} = 1 - \left(\frac{\beta_{max,\,k+1}}{\beta_{max,\,k}}\right) \tag{287}$$

and has essentially the same properties.

The use of a pure Newton-Raphson method would require the activity coefficients and their associated model equations to be brought directly into the set of n equations and n unknowns solved by the method, either directly or by substitution. In a previous section in this chapter, we noted that there was a problem in expanding the current set of master iteration variables in that the activity coefficients would have to be calculated before the concentrations of the non-basis species, and vice versa. This problem precludes taking care of the activity coefficients by a substitution mechanism that leaves the current set of master iteration variables unchanged. One would have to instead treat the activity coefficients themselves as master iteration variables. These are many, so this would not be very convenient. Alternatively, one could treat the ionic strength as a master iteration variable, but this would only suffice for simple extended Debye-Hückel formalisms that are only valid in dilute solutions. We have instead chosen to hybridize the Newton-Raphson

method by simply updating the activity coefficients between Newton-Raphson steps. In practice, this seems to work quite well, except in some extremely concentrated solutions.

The EQLIB module **newton.f** oversees the Newton-Raphson iteration for EQ3NR. Fellow EQLIB module **nrstep.f** is called to make a single Newton-Raphson step, and fellow EQLIB module **ngcadv.f** recomputes activity coefficients and computes the number of moles of dependent species. The latter module is called between Newton-Raphson steps, in accordance with the single update method that was described earlier in this chapter. The EQ3NR module **betas.f** computes the residual functions, and the EQ3NR module **matrix.f** writes the Jacobian matrix.

The maximum number of iterations in a Newton-Raphson calculation is determined by the **input** file variable **itermx**. This has a default value of 30 in EQ3NR. Convergence is achieved when β_{max} is less than the tolerance parameter **tolbt**, δ_{max} is less than the tolerance parameter **toldl**, and max norms on the changes in the Σm function, the ionic strength, and the activity coefficients are all less than **tolbt**. The tolerance parameters **tolbt** and **toldl** both appear on the **input** file, and both have a default value of 1×10^{-6}.

9.8. Derivation of Residual Functions and the Jacobian Matrix

In this section, we shall derive the residual functions and the Jacobian matrix for the Newton-Raphson iteration procedures used by the EQ3NR code. Given a set of governing equations and an equal number of unknowns, there is no unique way to formulate residuals and Jacobians. The number of equations and unknowns may be reduced by means of substitutions. Furthermore, one may then construct the residual functions in any number of ways. Once the residual functions have been chosen, the form of the Jacobian is determined according to the partial derivatives of these functions.

We will now take each remaining governing equation, construct a corresponding pair of residual functions (α and β), and derive the corresponding row of elements in the Jacobian matrix by partial differentiation. The α residuals are the true Newton-Raphson residual functions and are the subject of partial differentiation to define the Jacobian matrix. The β residuals are better measures of satisfactory convergence.

9.8.1. Mass Balance

This may be applied to any aqueous species in the basis set ($s = 1$ through s_Q) except water (w), H^+, and $O_{2(g)}$ (the s_B-th species). Mass balance is specified as the governing equation by setting the corresponding **jflag** value to 0 and entering a total concentration on the molal scale ($m_{T,s}$). Alternatively, one may enter total concentration in other units using other **jflag** values (molar, **jflag** = 1; mg/L, **jflag** = 2; mg/kg of solution, **jflag** = 3), which EQ3NR will then recompute into molality and set **jflag** to zero. The governing equation can be written as:

$$m_{T,s} = \sum_{s'=1}^{s_Q} u_{s's} m_s + \sum_{s''=s_Q+1}^{s_T} u_{s''s} m_{s''} \qquad (288)$$

The residual functions are defined by:

$$\alpha_s = -m_{T,s} + \sum_{s'=1}^{s_Q} u_{s's} m_s + \sum_{s''=s_Q+1}^{s_T} u_{s''s} m_{s''} \tag{289}$$

$$\beta_s = \frac{\alpha_s}{m_{T,s}} \tag{290}$$

where in the last part of eqs (288) and (289) it is implied that $s'' = s_r$. From this point, we may use the relations developed above to derive the following Jacobian elements:

$$\begin{aligned} J_{sw} &= \frac{\partial \alpha_s}{\partial log x_w} \\ &= \sum_{r=1}^{r_T} u_{s''s} \frac{\partial m_{s''}}{\partial log x_w} \\ &= -2.303 \sum_{r=1}^{r_T} \frac{u_{s''s} m_{s''} b_{wr}}{b_{s''r}} \\ &= -2.303 \sum_{r=1}^{r_T} b_{wr} H_{sr} \end{aligned} \tag{291}$$

$$\begin{aligned} J_{ss_B} &= \frac{\partial \alpha_s}{\partial log f_{O_2}} \\ &= \sum_{r=1}^{r_T} u_{s''s} \frac{\partial m_{s''}}{\partial log f_{O_2}} \\ &= 2.303 \sum_{r=1}^{r_T} \frac{u_{s''s} m_{s''} b_{s_B r}}{b_{s''r}} \\ &= 2.303 \sum_{r=1}^{r_T} b_{s_B r} H_{sr} \end{aligned} \tag{292}$$

and for $s \neq w$, s_B:

$$J_{ss'} = \frac{\partial \alpha_s}{\partial log m_{s'}}$$

$$= \sum_{s''' = 1}^{s_Q} u_{s'''s} \frac{\partial m_{s'''}}{\partial log m_{s'}} + \sum_{s'' = s_Q + 1}^{s_T} u_{s''s} \frac{\partial m_{s''}}{\partial log m_{s'}}$$

$$= -2.303 \left(u_{s's} m_{s'} - \sum_{r=1}^{r_T} \frac{u_{s''s} m_{s''} b_{s'r}}{b_{s''r}} \right)$$

$$= -2.303 \left(u_{s's} m_{s'} - \sum_{r=1}^{r_T} b_{s'r} H_{sr} \right) \tag{293}$$

Here $u_{s'''s'} = 1.0$ if $s' = s'''$, otherwise $u_{s'''s'} = 0$.

9.8.2. Electrical Balance

This governing equation may be applied to one of the ions in the aqueous species basis set, here denoted by s. Apart from the definition of the β residual, the treatment is exactly analogous to that for mass balance. The governing equation can be written as:

$$\sum_{s'=1}^{s_Q} z_{s'} m_s + \sum_{s''=s_Q+1}^{s_T} z_{s''} m_{s''} = 0 \tag{294}$$

The residual functions are defined by:

$$\alpha_s = \sum_{s'=1}^{s_Q} z_{s'} m_s + \sum_{s''=s_Q+1}^{s_T} z_{s''} m_{s''} \tag{295}$$

$$\beta_s = \frac{\alpha_s}{\sum_{s'=1}^{s_Q} |z_{s'}| m_s + \sum_{s''=s_Q+1}^{s_T} |z_{s''}| m_{s''}} \tag{296}$$

The corresponding Jacobian elements are as follows:

$$J_{sw} = \frac{\partial \alpha_s}{\partial log x_w}$$

$$= -2.303 \sum_{r=1}^{r_T} b_{wr} H_{zr} \tag{297}$$

$$J_{ss_B} = \frac{\partial \alpha_s}{\partial log f_{O_2}}$$

$$= 2.303 \sum_{r=1}^{r_T} b_{s_B r} H_{zr} \tag{298}$$

and for $s \neq w$, s_B:

$$J_{ss'} = \frac{\partial \alpha_s}{\partial log m_{s'}}$$

$$= -2.303 \left(z_{s'} m_{s'} - \sum_{r=1}^{r_T} b_{s'r} H_{zr} \right) \tag{299}$$

9.8.3. The Mole Fraction Of Water

The governing equation can be written as:

$$x_w = \frac{\Omega}{\Omega + \sum_{\substack{s=1 \\ s \neq w}}^{s_T} m_s} \tag{300}$$

where Ω is the number of moles of solvent water comprising a mass of 1 kg ($\Omega \approx 55.51$) and s_T is the number of aqueous species in the solution. The corresponding residual functions are defined as:

$$\alpha_w = log\left(\frac{\Omega}{\Omega + \sum_{\substack{s=1 \\ s \neq w}}^{s_T} m_s}\right) - log x_w \tag{301}$$

$$\beta_s = \alpha_s \tag{302}$$

Because it is necessary to distinguish between basis species and non-basis species, it is helpful to write the equation for the first residual function in the slightly expanded form:

$$\alpha_w = log\left(\frac{\Omega}{\Omega + \sum_{\substack{s'=1 \\ s' \neq w}}^{s_Q} m_{s'} + \sum_{r=1}^{r_T} m_{s''}}\right) - log x_w \tag{303}$$

In the following equations, we will take s' and s''' to be basis species other than water. The following Jacobian elements are then obtained:

$$J_{ww} = \left(\frac{1}{\Omega}\right)\left(\frac{\Omega}{\Omega + \sum_{\substack{s'=1 \\ s' \neq w}}^{s_Q} m_{s'} + \sum_{r=1}^{r_T} m_{s''}}\right)\left(\sum_{r=1}^{r_T} \frac{b_{wr} m_{s''}}{b_{s''r}}\right) - 1.0 \tag{304}$$

$$J_{ws'''} = \left(-\frac{1}{\Omega}\right)\left(\frac{\Omega}{\Omega + \sum_{\substack{s'=1 \\ s' \neq w}}^{s_Q} m_{s'} + \sum_{r=1}^{r_T} m_{s''}}\right)\left(m_{s'''} - \sum_{r=1}^{r_T} \frac{b_{s'''r} m_{s''}}{b_{s''r}}\right) \tag{305}$$

Note that these Jacobian elements differ from the corresponding set of partial derivatives used in the truncated Taylor's expansion as part of the pre-Newton-Raphson optimization. In the present case, $log\, x_w$ is treated as an independent variable. In the previous case, it was not.

9.8.4. Specified Free Concentration

This represents the **jflag** = 4 option for $s = 1$ through s_Q, except when s is water (w) or $O_{2(g)}$ (s_B). The free concentration is in molality. If it is initially entered in molarity (**jflag** = 5), EQ3NR will converts it to molality and resets **jflag** to 4 before beginning Newton- Raphson iteration. The governing equation is just the identity:

$$m_s = m_s \tag{306}$$

Hence, the corresponding residual functions are given by:

$$\alpha_s = 0 \tag{307}$$

$$\beta_s = 0 \tag{308}$$

In order to prevent singularity in the Jacobian, we set:

$$J_{ss} = 1.0 \tag{309}$$

9.8.5. Specified Thermodynamic Activity

This represents the **jflag** = 16 option for $s = 1$ through s_Q, except when s is water (w) or $O_{2(g)}$ (s_B). This option is most frequently employed with H^+ in order to specify a pH value ($pH = -log\ a_{H^+}$). The governing equation is:

$$log m_s + log \gamma_s = log a_s \tag{310}$$

The residual functions are:

$$\alpha_s = -log a_s + log m_s + log \gamma_s \tag{311}$$

$$\beta_s = \alpha_s \tag{312}$$

The only non-zero Jacobian element is:

$$J_{ss} = 1.0 \tag{313}$$

9.8.6. Log Activity Combination

Recall that the activity combination parameter is defined by:

$$\aleph_{ij} = |z_j| log a_i - \frac{z_i z_j}{|z_j|} log a_j \tag{314}$$

We will identify i as the basis species s to which this constraint is applied, and s^* as the other basis species involved. The governing equation can then be written as:

$$log m_s = \frac{\aleph_{ss*}}{|z_{s*}|} - log\gamma_s + \frac{z_s}{z_{s*}} log m_{s*} + \frac{z_s}{z_{s*}} log\gamma_{s*} \tag{315}$$

Hence, the corresponding residual functions are given by:

$$\alpha_s = \frac{\aleph_{ss*}}{|z_{s*}|} - log\gamma_s + \frac{z_s}{z_{s*}} log m_{s*} + \frac{z_s}{z_{s*}} log\gamma_{s*} - log m_s \tag{316}$$

$$\beta_s = \alpha_s \tag{317}$$

The corresponding non-zero elements of the Jacobian matrix are then:

$$J_{ss} = -1.0 \tag{318}$$

$$J_{ss*} = \frac{z_s}{z_{s*}} \tag{319}$$

9.8.7. Mean Log Activity

The mean log activity of two oppositely charge ions i and j can be written as:

$$log a_{\pm,ij} = \frac{|z_j| log a_i + |z_i| log a_j}{|z_i| + |z_j|} \tag{320}$$

We will identify i as the basis species s to which this constraint is applied, and s^* as the other basis species involved. The governing equation can then be written as:

$$log m_s = \frac{|z_s| + |z_{s*}|}{|z_{s*}|} log a_{\pm,ss*} - log\gamma_s - \left|\frac{z_s}{z_{s*}}\right| log m_{s*} - \left|\frac{z_s}{z_{s*}}\right| log\gamma_{s*} \tag{321}$$

Hence, the corresponding residual functions are given by:

$$\alpha_s = \frac{|z_s| + |z_{s*}|}{|z_{s*}|} log a_{\pm,ss*} - log\gamma_s - \left|\frac{z_s}{z_{s*}}\right| log m_{s*} - \left|\frac{z_s}{z_{s*}}\right| log\gamma_{s*} - log m_s \tag{322}$$

$$\beta_s = \alpha_s \tag{323}$$

The corresponding non-zero elements of the Jacobian matrix are then:

$$J_{ss} = -1.0 \tag{324}$$

$$J_{ss*} = -\left|\frac{z_s}{z_{s*}}\right| \tag{325}$$

9.8.8. Equilibrium With A Pure Mineral

This option (**jflag** = 19) may be specified for any aqueous species denoted by $s = 1$ through s_Q, except when s is water (w). Let ϕ denote the mineral in question. The governing equation is:

$$logK_\phi = b_{w\phi}(logx_w + log\lambda_w) + b_{s_B\phi} logf_{O_2} + \sum_{\substack{s'=1 \\ s' \neq w, s_B}}^{s_Q} b_{s'\phi}(logm_{s'} + log\gamma_{s'}) \tag{326}$$

The residual functions are defined as:

$$\alpha_s = \frac{1}{b_{s\phi}} \left(logK_\phi - b_{w\phi}(logx_w + log\lambda_w) - b_{s_B\phi} logf_{O_2} - \sum_{\substack{s'=1 \\ s' \neq w, s_B}}^{s_Q} b_{s'\phi}(logm_{s'} + log\gamma_{s'}) \right) \tag{327}$$

$$\beta_s = \alpha_s \tag{328}$$

The corresponding Jacobian elements are then:

$$J_{ss'} = -\frac{b_{s'\phi}}{b_{s\phi}} \tag{329}$$

The residual function α_s defined in eq (327) has in a sense been normalized by dividing by the stoichiometric reaction coefficient $b_{s\phi}$. This makes the residual equivalent to the difference between the calculated and current values of log m_s, independent of how the reaction has been written. This avoids some potential numerical scaling problems. Other options involving mass action equations are treated in the same manner.

9.8.9. Equilibrium With A Solid Solution End-member Component

This option (**jflag** = 20) may be specified for any aqueous species denoted by $s = 1$ through s_Q, except when s is water (w). The treatment is closely analogous to that for equilibrium with a pure mineral. Let σ and ψ denote the end-member and solid solution phase, respectively. The governing equation contains an additional term in the mole fraction and activity coefficient of the solid solution end-member and is given by:

$$logK_{\sigma\psi} = b_{\sigma\psi\sigma\psi}(logx_{\sigma\psi} + log\lambda_{\sigma\psi}) + b_{w\sigma\psi}(logx_w + log\lambda_w) + b_{s_B\sigma\psi}logf_{O_2}$$

$$+ \sum_{\substack{s'=1 \\ s' \neq w, s_B}}^{s_Q} b_{s'\sigma\psi}(logm_{s'} + log\gamma_{s'}) \tag{330}$$

The residual functions are defined as:

$$\alpha_s = \frac{1}{b_{s\sigma\psi}}\left(logK_{\sigma\psi} - b_{\sigma\psi\sigma\psi}(logx_{\sigma\psi} + log\lambda_{\sigma\psi}) - b_{w\sigma\psi}(logx_w + log\lambda_w) - \right.$$

$$\left. - b_{s_B\sigma\psi}logf_{O_2} - \sum_{\substack{s'=1 \\ s' \neq w, s_B}}^{s_Q} b_{s'\sigma\psi}(logm_{s'} + log\gamma_{s'}) \right) \tag{331}$$

$$\beta_s = \alpha_s \tag{332}$$

The corresponding Jacobian elements are then:

$$J_{ss'} = -\frac{b_{s'\sigma\psi}}{b_{s\sigma\psi}} \tag{333}$$

9.8.10. Equilibrium With A Gas

This option (**jflag** = 21) may be specified for any aqueous species denoted by $s = 1$ through s_Q, except when s is water (w). The treatment is closely analogous to that for equilibrium with a pure mineral. Let g denote the gas in question. The governing equation contains an additional term in the fugacity of this gas and is given by:

$$logK_g = b_{gg}logf_g + b_{wg}(logx_w + log\lambda_w) + b_{s_Bg}logf_{O_2} + \sum_{\substack{s'=1 \\ s' \neq w, s_B}}^{s_Q} b_{s'g}(logm_{s'} + log\gamma_{s'}) \tag{334}$$

The residual functions are defined as:

$$\alpha_s = \frac{1}{b_{sg}} \left(logK_g - b_{gg} logf_g - b_{wg} (logx_w + log\lambda_w) \right.$$

$$\left. - b_{s_B g} logf_{O_2} - \sum_{\substack{s' = 1 \\ s' \neq w,\, s_B}}^{s_Q} b_{s'g} (logm_{s'} + log\gamma_{s'}) \right) \tag{335}$$

$$\beta_s = \alpha_s \tag{336}$$

The corresponding Jacobian elements are then:

$$J_{ss'} = -\frac{b_{s'g}}{b_{sg}} \tag{337}$$

9.8.11. Concentration Fixed By Internal Equilibrium

This option (**jflag** = 27) excludes the species to which it is applied, which must be in the auxiliary basis set, and its ion-pairs and complexes from the mass balance of the corresponding basis species to which it is linked by its own associated reaction (usually a strict basis species). This is a good choice for dissolved gas species such as $O_{2(aq)}$ and $H_{2(aq)}$. If HS^- is an auxiliary basis species with **jflag** = 27 and it is linked to SO_4^{2-}, then HS^- and its "complexes" (other species whose reactions link them to this species) are not included in calculating the SO_4^{2-} mass balance. If instead one chooses **jflag** = 30 for HS^-, they are included.

Let s be the auxiliary basis species constrained by **jflag** = 27, and let r denote its associated reaction. The governing is then:

$$logK_r = b_{sr} (logm_s + log\gamma_s) + b_{wr} (logx_w + log\lambda_w)$$

$$+ b_{s_B r} logf_{O_2} + \sum_{\substack{s' = 1 \\ s' \neq w,\, s_B,\, s}}^{s_Q} b_{s'r} (logm_{s'} + log\gamma_{s'}) \tag{338}$$

The residual functions are defined as:

$$\alpha_s = \frac{1}{b_{sr}} \left(log K_r - b_{sr} (log m_s + log \gamma_s) - b_{wr} (log x_w + log \lambda_w) \right.$$

$$\left. - b_{s_B r} log f_{O_2} - \sum_{\substack{s' = 1 \\ s' \neq w, s_B, s}}^{s_Q} b_{s'r} (log m_{s'} + log \gamma_{s'}) \right) \tag{339}$$

$$\beta_s = \alpha_s \tag{340}$$

The corresponding Jacobian elements are then:

$$J_{ss'} = -\frac{b_{s'r}}{b_{sr}} \tag{341}$$

9.8.12. Specified Log Oxygen Fugacity

This option (**iopt1** = 0) allows direct specification of the log oxygen fugacity. The governing equation is just the identity:

$$log f_{O_2} = log f_{O_2} \tag{342}$$

The residuals are:

$$\alpha_s = 0 \tag{343}$$

$$\beta_s = 0 \tag{344}$$

The only non-zero Jacobian element in the corresponding row is:

$$J_{ss} = 1.0 \tag{345}$$

9.8.13. Specified *Eh*

This option (**iopt1** = -1) allows indirect specification of the log oxygen fugacity. If *pe* is specified (**iopt1** = -2), EQ3NR converts it to *Eh* before it does the Newton-Raphson iteration. Letting F be the Faraday constant, R the gas constant, and T the absolute temperature, the governing equation can be written as:

$$log f_{O_2} = \frac{4FEh}{2.303RT} + log K_{Eh} + 2 (log x_w + log \lambda_w) - 4 (log m_{H^+} + log \gamma_{H^+}) \tag{346}$$

The residual functions are:

$$\alpha_{s_B} = \frac{4FEh}{2.303RT} + logK_{Eh} + 2\,(logx_w + log\lambda_w) - 4\,(logm_{H^+} + log\gamma_{H^+}) - logf_{O_2} \quad (347)$$

$$\beta_s = \alpha_s \quad (348)$$

The non-zero Jacobian elements in the corresponding row are:

$$J_{s_B w} = 2 \quad (349)$$

$$J_{s_B H^+} = -4 \quad (350)$$

$$J_{s_B s_B} = -1 \quad (351)$$

9.8.14. Oxygen Fugacity Fixed by An Aqueous Redox Couple

.This represents the **iopt1** = 1 option. Here s is restricted to s_B. The couple is specified on the **input** file by setting **uredox** to the name of the auxiliary basis species which comprises half of the couple (the other half is automatically the corresponding strict basis species). Let r denote the reaction associated with the auxiliary basis species (s_r) in the desired couple. The governing is then:

$$logK_r = b_{s_r r}\,(logm_{s_r} + log\gamma_{s_r}) + b_{wr}\,(logx_w + log\lambda_w)$$

$$+ b_{s_B r} logf_{O_2} + \sum_{\substack{s' = 1 \\ s' \neq w, s_B, s_r}}^{s_Q} b_{s' r}\,(logm_{s'} + log\gamma_{s'}) \quad (352)$$

The residual functions are defined as:

$$\alpha_s = \frac{1}{b_{s_B r}} \left(logK_r - b_{s_r r}\,(logm_{s_r} + log\gamma_{s_r}) - b_{wr}\,(logx_w + log\lambda_w) \right.$$

$$\left. - b_{s_B r} logf_{O_2} - \sum_{\substack{s' = 1 \\ s' \neq w, s_B, s_r}}^{s_Q} b_{s' r}\,(logm_{s'} + log\gamma_{s'}) \right) \quad (353)$$

$$\beta_s = \alpha_s \quad (354)$$

The corresponding Jacobian elements are then:

$$J_{ss'} = -\frac{b_{s'r}}{b_{sr}} \tag{355}$$

9.9. Methods for Crash Diagnostics

The iteration procedure is good enough that divergence almost always results from the input of a bad set of input constraints. EQ3NR screens the input prior to Newton-Raphson iteration, but some cases are not sufficiently obvious to be caught at this stage. When the iteration diverges, EQ3NR calls the module **ndiagx.f**, which examines final state of the iteration process in an attempt to generate diagnostics to write on the screen and **output** files. Barring the generation of a useful diagnostic, the user should examine the iteration summary on the **output** file for clues to the cause of the situation.

An iteration crash is generally the result of an iteration variable "blowing up" or, more commonly, "blowing down." "Blowing up" means that the value of an iteration variable or its corresponding residual function, usually a mass balance residual, is increasing to the point where the matrix equation can no longer be solved by the computer and iteration must terminate. Sometimes in this situation the operating system may terminate the code execution due to an overflow condition. If a variable "blows down," it probably means that no physical solution exists for the problem as posed. Because the primary iteration variables are logarithmic, underflow can only occur when these variables are exponentiated. When this happens, the code is usually stopped by encountering what appears to be a singular Jacobian matrix.

"Blow down" occurs for example when the problem calls for the concentration of a species to be adjusted to satisfy electrical balance, but this balance can only be achieved if the resulting concentration takes a negative value. An iteration process which only allows adjustments to log concentrations will never allow the generation of concentration value which is non-positive. What happens instead is that the calculated adjustments to the log concentration become large and negative. These are truncated to a value of -δ' (**-screwd**). As this adjustment truncation parameter has a magnitude of 2.0, the largest adjustment (element of the **del** array) takes on a value of -2.0. This can be seen in the iteration summary that is printed on the **output** file as the code executes Newton-Raphson iteration.

An iteration block printed during this iteration is exemplified by the following:

```
iter=   2
      del(conc     so4--    )= -1.77931E-03, delfnc= 9.34442E-01
     beta(conc     so4--    )= 8.87762E-05, betfnc= 9.79484E-01
       bbig= 8.87762E-05, ubbig= so4--
       bneg= 0.00000E+00, ubneg= none
       bgamx= 1.61604E-04, ubgamx= mg4(oh)4++++
       bsigmm= -1.18619E-04
       bxi= -3.19106E-04
       btfcnr= 9.97838E-01
```

This is taken from the sea water test case that was presented as the first example in Chapter 7. The **del** element with the largest magnitude and the **beta** element with the largest magnitude are printed first. We see that in both cases, these are the elements corresponding to the concentration of the basis species SO_4^{2-}. "Blow down" would be evident if the **del** output in this block were to

take on a value of -2.0 (usually for the last several iterations before the process is terminated). "Blow up" would be evident if the **beta** output in this block were to take on very high positive values. The variables **delfnc** and **betfnc** are convergence functions for the **del** and **beta** arrays, respectively. They usually start at small positive values less than unity and then approach unity if the iteration process converges. If one or the other takes on sustained negative values, the iteration process usually diverges. Here **bbig** is the largest positive mass balance residual (**ubbig** identifying the corresponding species) and **bneg** is the negative mass balance residual (if any) with the largest magnitude (**ubneg** identifying the corresponding species). The variable **bgamx** is an aqueous species activity coefficient residual function (the max norm on the absolute values of the differences between current and previous values of the activity coefficients of the aqueous species); **ubgamx** identifies the corresponding species. Also, **bsigm** is a residual on the Σm function (difference between the current and previous values), and **bxi** is a similar residual on the ionic strength. The variable **btfncr** is similar to **betfnc**, but measures the convergence in a pure Newton-Raphson step. Usually **betfnc** has a smaller value than **btfncr** because any adjustment of activity coefficient values in between Newton-Raphson steps tends to reduce the overall reduction in the residual functions.

10. Code Architecture and Flow of Execution

The purpose of the present chapter is to describe the structure of the software itself. This material is primarily included because it is required as part of the documentation to satisfy NUREG-0856 (Silling, 1983). It does not provide anything necessary for the typical code user. It may be helpful to those few users who desire to modify the code for whatever purpose.

In the present description, we will not make it a point to describe the role and function of every module in the source code. For such descriptions, the reader is referred to the relevant glossaries of modules. For EQ3NR modules, see Appendix B of the present report. For EQLIB modules, see Appendix A of the EQ3/6 Package Overview and Installation Guide (Wolery, 1992). The purpose here is to describe the main features and essential aspects of the structure of the code. The purpose is not to provide detailed design documentation. Readers who want more detailed information are invited to examine the source code itself, which is reasonably well-documented internally.

The main program is the module **eq3nr.f**. It directs the overall process of code execution. This is illustrated in Figure 4, which is a simplified flow diagram of the code from the point of view of this module. The first function of **eq3nr.f** is to get the time and date, which it does by calling the EQLIB module **timdat.f**. This module contains UNIX-dependent code. The main program then writes the code name and version identification, the copyright notice, and the time and date information to the screen and **output** files. Its next step is to initialize the dimensioning variables which correspond to the dimensioning parameters. Dimensioning variables are used to pass dimensioning data in the calling sequences of called modules; FORTRAN does not generally allow parameters (in the special FORTRAN sense) to be passed in calling sequences.

Recall that EQ3NR will run multiple problems stacked on a single **input** file. The line "20 continue" is a return point in the main program to which execution flows after a single problem has been solved. After this point, the code initializes the relevant arrays and variables, setting their contents to zero or some other appropriate null value equivalent. The purpose of this is to provide a clean slate for the solution of the next problem. The main program then calls module **rdinp.f** to read the first or a following problem on the **input** file. If no new problem is found, the problem execution stage is presumed to be finished. The main program then gets the current time and date and writes the starting and ending times and dates to the screen and **output** files. After that, it makes a "normal exit" and execution stops.

Module **rdinp.f** does the following. If the **input** file has not been opened on a previous call, it opens it. Using an EQLIB module called **stripl.f**, it copies the **input** file to a file called **inputs**, deleting any comment lines. The original **input** file is then closed. The code then subsequently reads the stripped input file instead. Module **rdinp.f** looks at the first line of this file to see if it is in "W" or "D" format. It then repositions the file pointer at the top of the file.

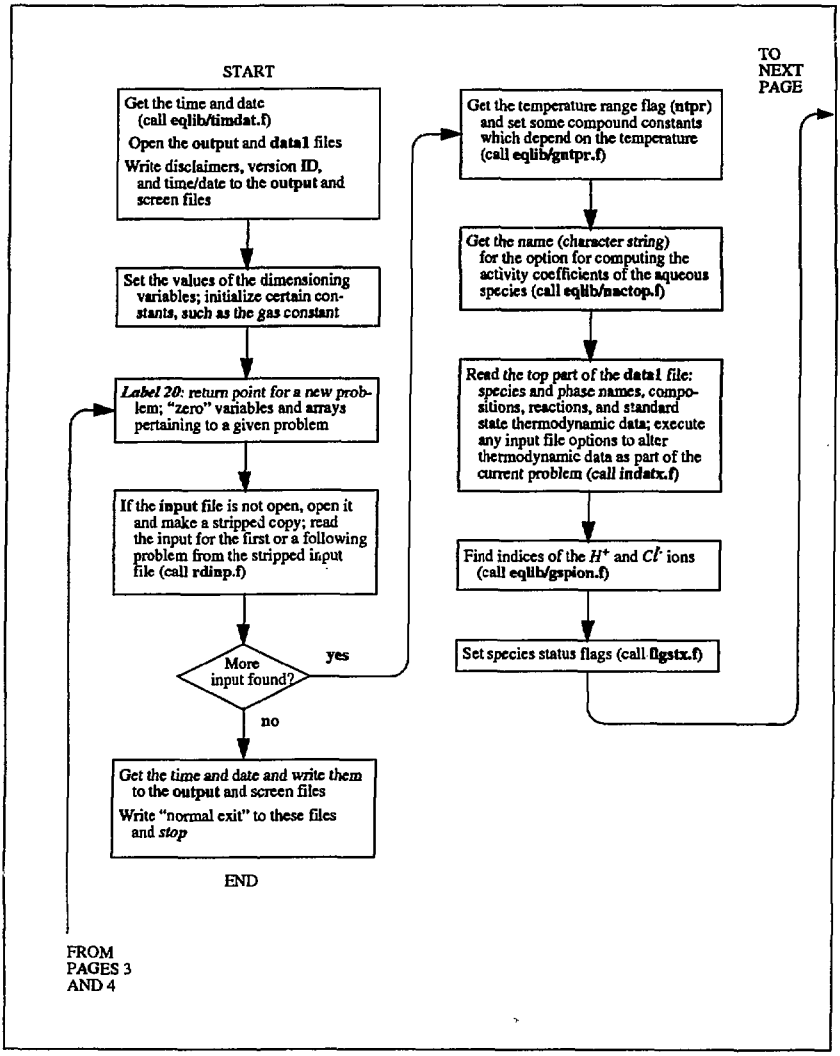

Figure 4 (page 1 of 4). Simplified flow diagram of the EQ3NR main program (eq3nr.f).

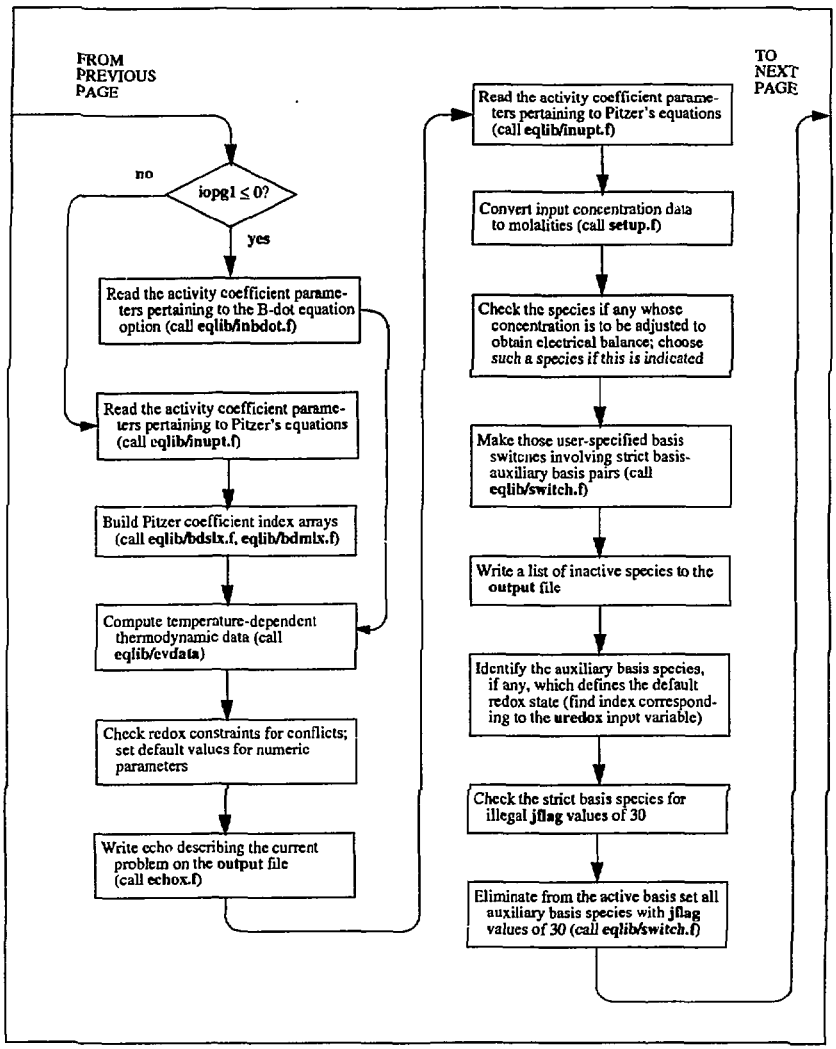

Figure 4 (continued, page 2 of 4). Simplified flow diagram of the EQ3NR main program (eq3nr.f).

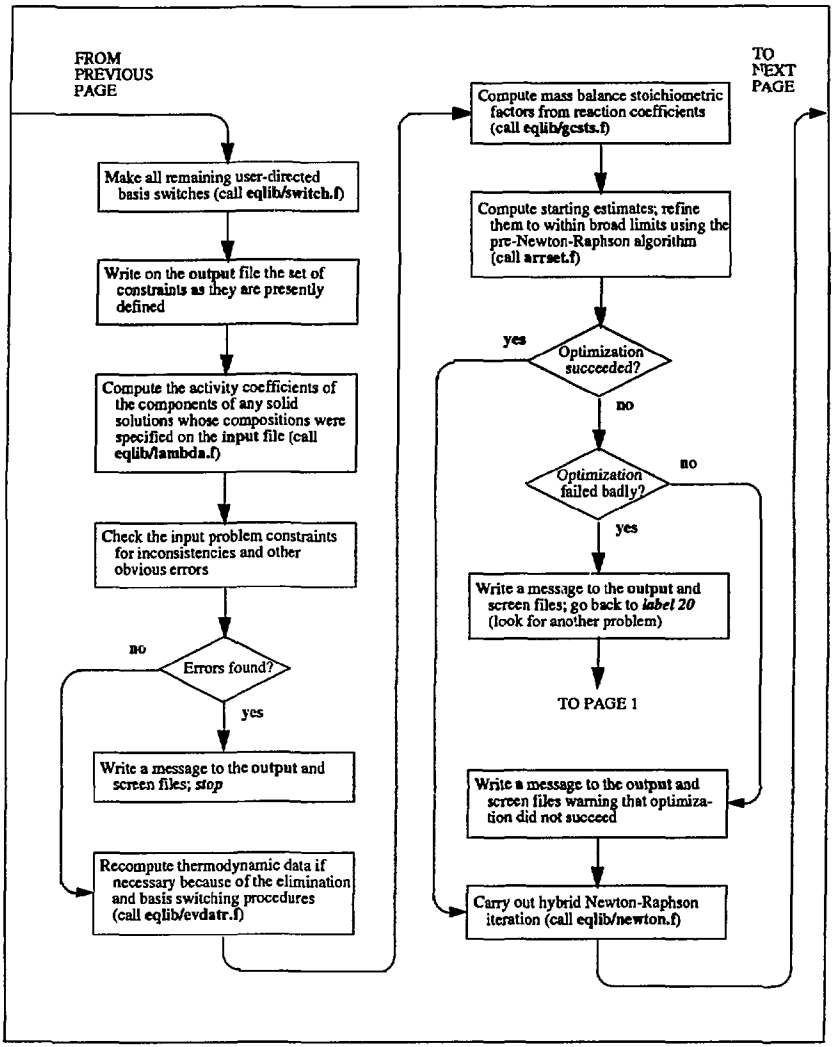

Figure 4 (continued, page 3 of 4). Simplified flow diagram of the EQ3NR main program (**eq3nr.f**).

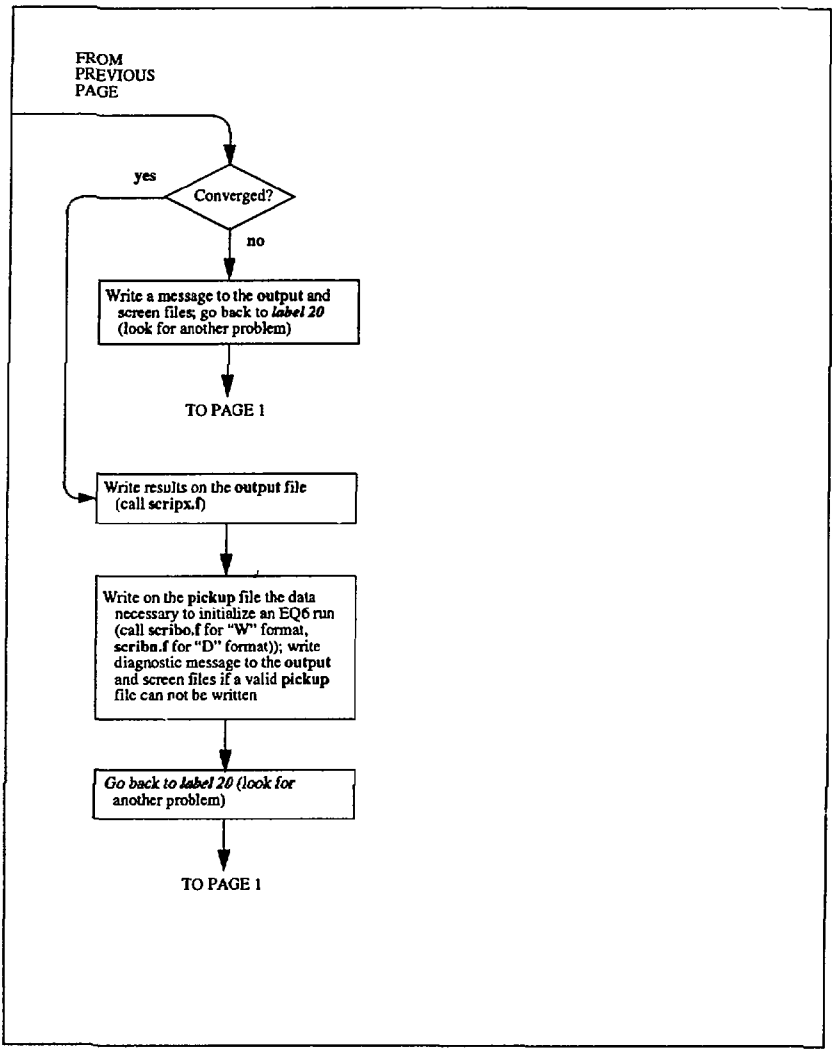

Figure 4 (continued, page 4 of 4). Simplified flow diagram of the EQ3NR main program (**eq3nr.f**).

If the **input** file is in "W" format, module **rdinp.f** calls module **readx.f** to read the next problem. If it is in "D" format, it calls module **rdninp.f** to oversee the reading of the next problem. Unlike **readx.f**, **rdninp.f** carries out its function by calling a fairly large number of other modules, all of which have names beginning with "**rd**" or "**get.**" Calls to either **readx.f** or **rdinp.f** result in the writing of an instant echo of the **input** file to the **output** file. If no problem is found at the current position of the file pointer, the main program is notified and it then proceeds as described above to make a normal exit.

If an input problem description is found on the **input** file, **eq3nr.f** sets the temperature range flag **ntpr** = 1 calling the EQLIB module **gntpr.f**. It then gets the name of the chosen activity coefficient option corresponding to the **iopg1** activity coefficient option switch. This is a descriptive character string that will be used subsequently. The code then calls module **indatx.f** to read the supporting data file, **data1**. This module checks a flag contained on the data file to insure that the kind of data file provided is consistent with the activity coefficient option selected on the **input** file. If it is not, this module writes an error message to the screen and **output** files and execution stops. Otherwise, it proceeds to read the standard state thermodynamic data on the data file. All basis species are loaded into memory (even if they are not needed). All other types of species are loaded into memory only if they are relevant to the current problem.

The primary method of indexing data pertinent to species, reactions, etc., follows from the reading of the data file. Corresponding data read from the **input** file are stored in holding arrays. For example, the names of the basis species to include in the model and the corresponding "**csp**" inputs are initially stored in holding arrays called **uspecb** and **cspb**, respectively. The names of the aqueous species read from the data file are kept in the **uspec** array. The actual **csp** array is constructed by name matching between the **uspecb** and **uspec** arrays. This is done by the module **gcsp.f**, which is called by **indatx.f**. The module **indatx.f** also calls the EQLIB module **alters.f** to execute any **nxmod** "alter" options that may have been included on the **input** file.

The main program then calls the EQLIB module **gspion.f** to find the indices of the hydrogen and chloride ions, and follows this by calling the module **flgstx.f** to set the species status flags. The latter module may in turn call the EQLIB module **supprs.f** to execute any **nxmod** "suppress" options that may have been included on the **input** file.

Execution then returns to the main program. If **iopg1** ≤ 0, **eq3nr.f** calls the module **inbdot.f** to read from the data file the hard core diameter (**azero**) and **insgfl** flag data required for the B-dot activity coefficient model. This is done even if **iopg1** = -1 (the Davies equation option, in which case these data are not used). If **iopg1** = 1, **eq3nr.f** calls the module **inupt.f** to read from the data file the relevant interaction parameters needed for Pitzer's equations. This is followed by calls to the EQLIB modules **bdslx.f** and **bdmlx.f**, which build index arrays used in the evaluation of Pitzer's equations. Only the data needed for the current problem are loaded into memory by either **inbdot.f** or **inupt.f**.

The thermodynamic data that have been read in at this point are in various forms which do not generally correspond to the data required at the temperature for the given problem. For example, the equilibrium constants as read from the data file are only represented as the coefficients of interpolating polynomials. The main program then calls the EQLIB module **evdata.f** to compute the needed data for the temperature specified for the current problem.

The main program then checks the redox constraints for the current problem to see if there are any conflicts. If any are found, it resolves them and writes appropriate warning messages to the screen and **output** files. It then sets default values as required for some of the **input** file parameters. The next action of the main program is to call module **echox.f**. This writes an echo of the current problem to the **output** file. This echo includes the values of any default parameters that have been assigned.

The next action of **eq3nr.f** is to call module **setup.f**. This converts concentration data which are not in terms of molalities to molalities. The main program then identifies the species to be adjusted for electrical balance, or finds one which is suitable for this purpose if that is what the user requested on the **input** file. A choice made by the user is then checked for suitability. Some warnings and notes may be written to the screen and **output** files if any problems are apparent, such as the specified species not having any electrical charge (in which case the calculation may well fail, but not necessarily, as the concentrations of charged non-basis species will change in response to adjustments to the concentration of the species defined in the balance constraint).

The main program then executes those basis switches that were specified on the **input** file and involve switching the roles of strict and auxiliary basis species (other basis switches specified on the **input** file are done later). In doing this, it makes use of the EQLIB module **switch.f**. Then it writes a list of inactive species to the **output** file. These include species which have been suppressed by user options on the **input** file and species which were loaded from the data file, but for which the requisite thermodynamic data are lacking.

Module **eq3nr.f** then identifies the auxiliary basis species which defines the redox couple to be used to calculate the default redox state, if this option has been selected on the **input** file. It then checks to insure that there are no cases of a **jflag** value of 30 being assigned to a strict basis species. It then calls the EQLIB module **elim.f** to eliminate all auxiliary basis species with **jflag** = 30 from the active basis set. Then it executes (again by calling the EQLIB module **switch.f**) those basis switches which were specified on the **input** file and bring non-basis species into the basis set. The main program then writes to the **output** file a table describing the input constraints as it presently understands them. If solid solution compositions have been entered on the **input** file, the main program now calls the EQLIB module **lambda.f** to evaluate the activity coefficients of the components of these phases.

The main program then makes a rather extensive set of tests on the current problem inputs, looking for inconsistencies and other obvious errors. If problems are found, error messages are written to the screen and **output** files. At the end of this process, if one or more errors have been detected, the code stops execution.

The next action of **eq3nr.f** is to call the EQLIB module **evdatr.f** to recompute those thermodynamic data which have to be changed in response to the rewriting of reactions due to elimination of auxiliary basis species from the active basis set and basis switching. During the actual rewriting of reactions associated with such activities, the associated interpolating polynomials are recalculated, but the corresponding equilibrium constants are not immediately recalculated. The equilibrium constants that were evaluated previously by **evdata.f** (which itself called **evdatr.f** to do this) are presently not used in any code function. In the past, they have been used to verify that the data were being properly reconstructed in the code prior to the steps in which the reactions

were rewritten. Depending on the choice of the **input** file option switch **iopr2**, the reactions and associated data may be written to the **output** file after these actions have taken place. Module **eq3nr.f** then calls the EQLIB module **gcsts.f**, which computes the stoichiometric mass balance factors, which are kept in the **stor1** array.

At this point, the code is ready to solve the system of algebraic equations posed by the current problem. The main program calls module **arrset.f**, which sets up the appropriate computational structures (e.g., the array defining the master iteration variables), assigns starting values, and attempts to refine them using the pre-Newton-Raphson optimization algorithm discussed in Chapter 9. The flow of execution in this module is illustrated in the simplified flow diagram in Figure 5. Note that there is an initial estimation procedure, followed by a fairly complex optimization process. This consists of an outer loop structure (*loops*) in which automatic basis switching is carried out (this happens only if **iopt2** is set to 1 on the **input** file). Inside this is a middle loop structure (*passes*) in which Σm, I, and the activity coefficients of aqueous species are updated. Inside this in turn is an inner loop (*cycles*) in which the primary optimization (adjustment of concentrations of the basis species) is carried out. In applying the optimization algorithm, **arrset.f** calls module **arrsim.f** to compute refined values for those basis species concentrations which must be determined simultaneously (instead of by successively evaluating individual equations for the basis species concentrations, as is the case in the rest of the algorithm).

The goal of the optimization procedure in **arrset.f** is only to get the iterated values to within about an order of magnitude of the solution, so that subsequent Newton-Raphson iteration will have sufficiently good values to start with.This optimization process may succeed or not. If not, this module checks to see if any computed basis species concentrations are outrageously high. If this is the case, the problem as stated is almost assuredly ill-posed (has no realistic solution).

If the problem appears to be ill-posed, **eq3nr.f** writes a note to the screen and **output** files and gives up on the current problem and loops back to see if there is another problem on the **input** file. Otherwise, whether or not the optimization step succeeded, it continues by calling the EQLIB module **newton.f** to compute the final solution. This executes the hybrid Newton-Raphson algorithm that was also discussed in Chapter 9. The flow of execution in **newton.f** is illustrated in the simplified flow diagram in Figure 6. This module in turn calls the module **betas.f** and **matrix.f** to compute, respectively, the residual functions and the Jacobian matrix. These are known to **newton.f** as "**betae**" and "**matrxe**," respectively. Module **newton.f** also calls the module **ncmpx.f** to expand the system. This routine is known to it as "**ncmpe**." Module **newton.f** calls the EQLIB module **ngcadv.f** to recompute the activity coefficients. It also calls the EQLIB module **nrstep.f** to execute a single Newton-Raphson step. The flow of execution in **nrstep.f** is illustrated in the simplified flow diagram in Figure 7. Note that this is actually a fairly complex piece of coding.

If hybrid Newton-Raphson iteration fails, the main program writes an error message to the screen and **output** files. It then calls module **ndiagx.f** to examine the results of the failed calculation in an attempt to generate useful diagnostics. If any are generated, they are written to the screen and **output** files. Most of the possible messages that might be written identify the relevant aspects of ill-posed input. The code then loops back to look for another problem on the **input** file.

If hybrid Newton-Raphson iteration succeeds, the code continues processing the current problem. The next action of the main program is to call the module **scripx.f**. This writes a description of all relevant results to the **output** file. As necessary, it calculates various secondary parameters before writing them. What this module writes is essentially all of what the user sees as the results of the calculation.

The next action of **eq3nr.f** is to see if a valid **pickup** file can be written. To be valid for transmittal to EQ6, the aqueous solution model must include each strict basis species which is linked to each auxiliary basis species in the model. If a valid **pickup** file can not be written, the main program writes a note to the screen and **output** files. Otherwise, a **pickup** file is then written. If the **input** file was in "W" format, the main program calls the module **scribo.f**, which writes the **pickup** file in the corresponding format. If the **input** file was in "D" format, it calls the module **scribx.f**, which writes the **pickup** file in the corresponding format.

This terminates all code activity on the current problem. The main program then loops back to look for another problem on the ***input*** file.

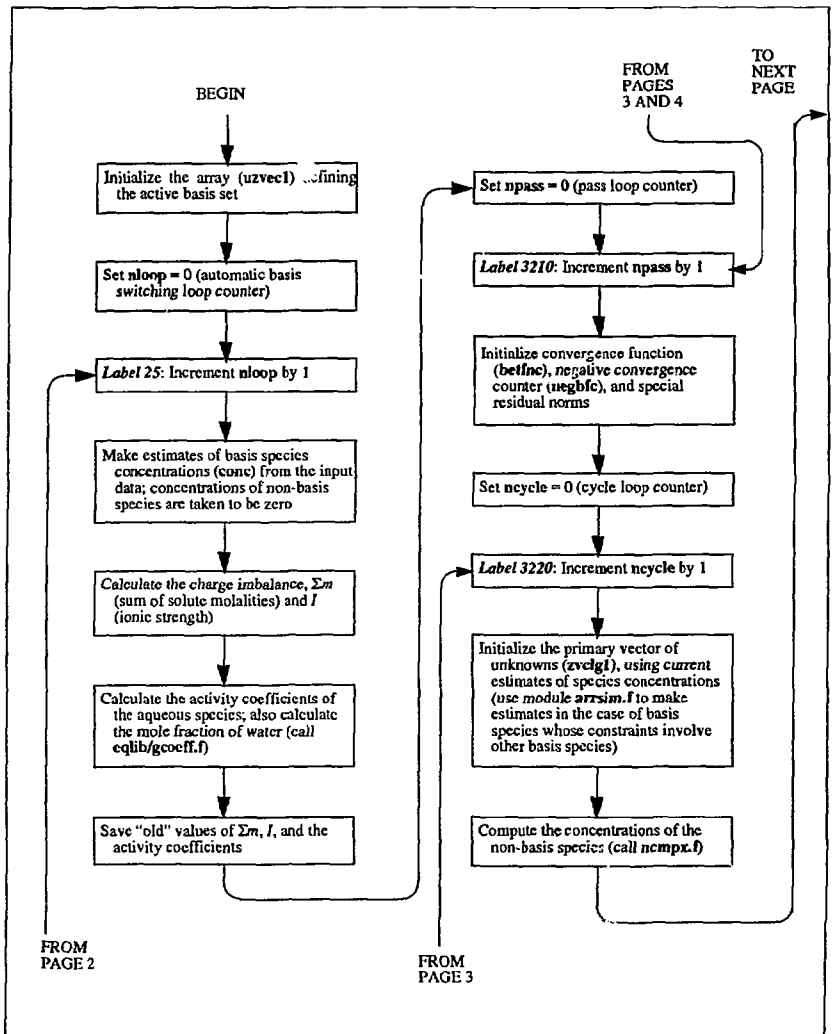

Figure 5 (page 1 of 4). Simplified flow diagram for creation of starting values and pre-Newton-Raphson optimization (from the viewpoint of module **arrset.f**).

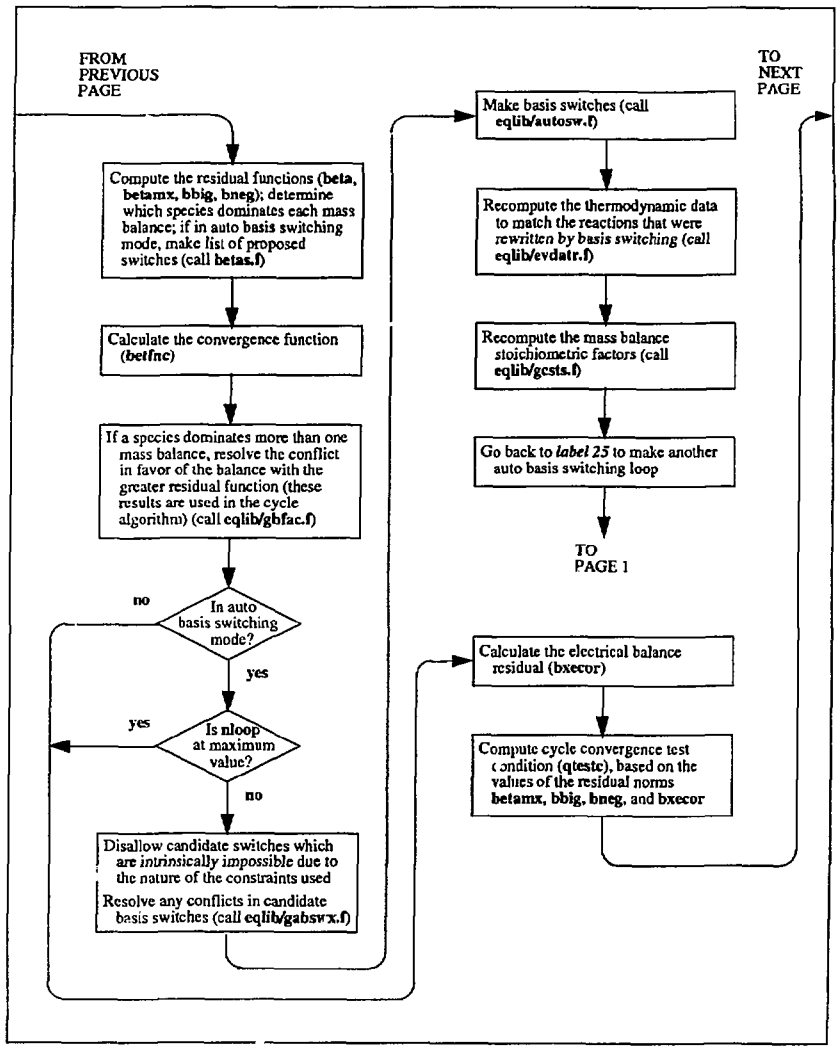

Figure 5 (continued, page 2 of 4). Simplified flow diagram for creation of starting values and pre-Newton-Raphson optimization (from the viewpoint of module arrset.f).

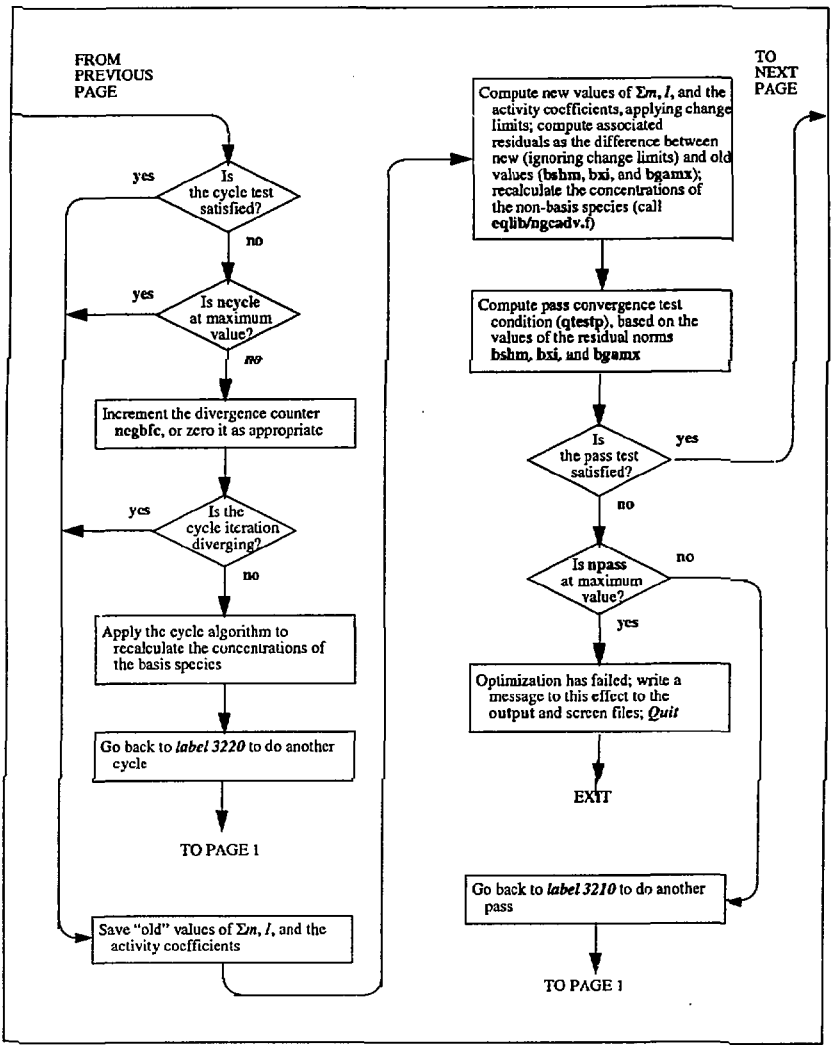

Figure 5 (continued, page 3 of 4). Simplified flow diagram for creation of starting values and pre-Newton-Raphson optimization (from the viewpoint of module **arrset.f**).

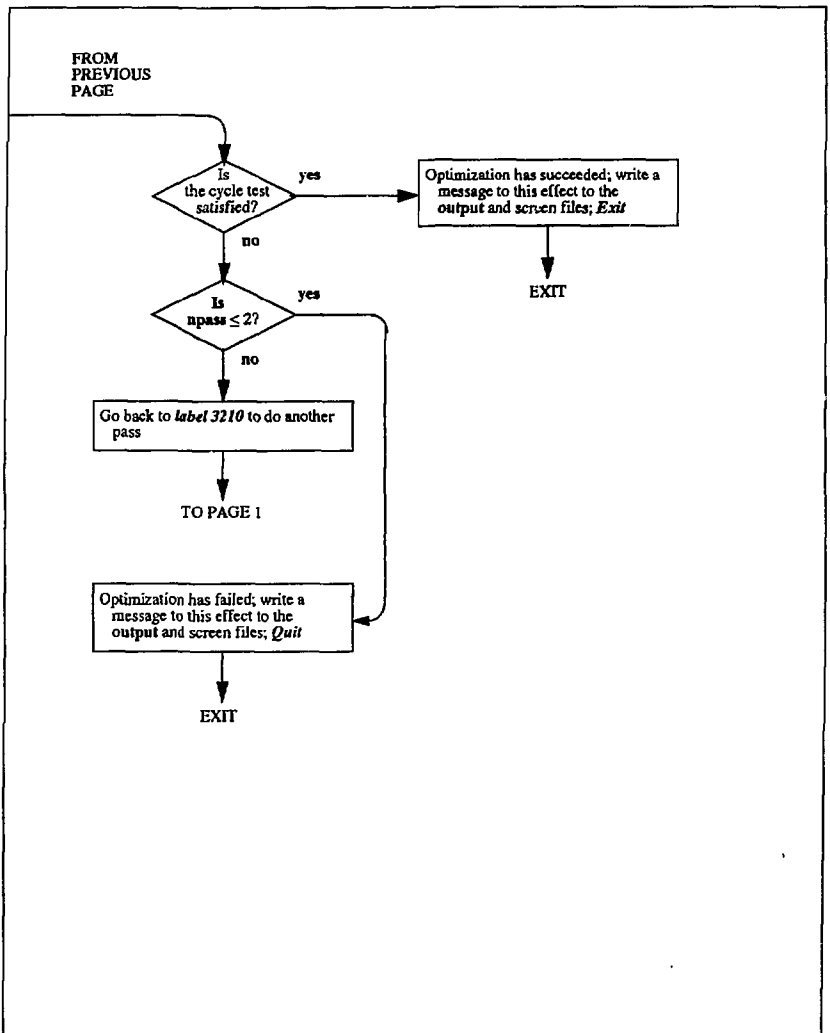

Figure 5 (continued, page 4 of 4). Simplified flow diagram for creation of starting values and pre-Newton-Raphson optimization (from the viewpoint of module **arrset.f**).

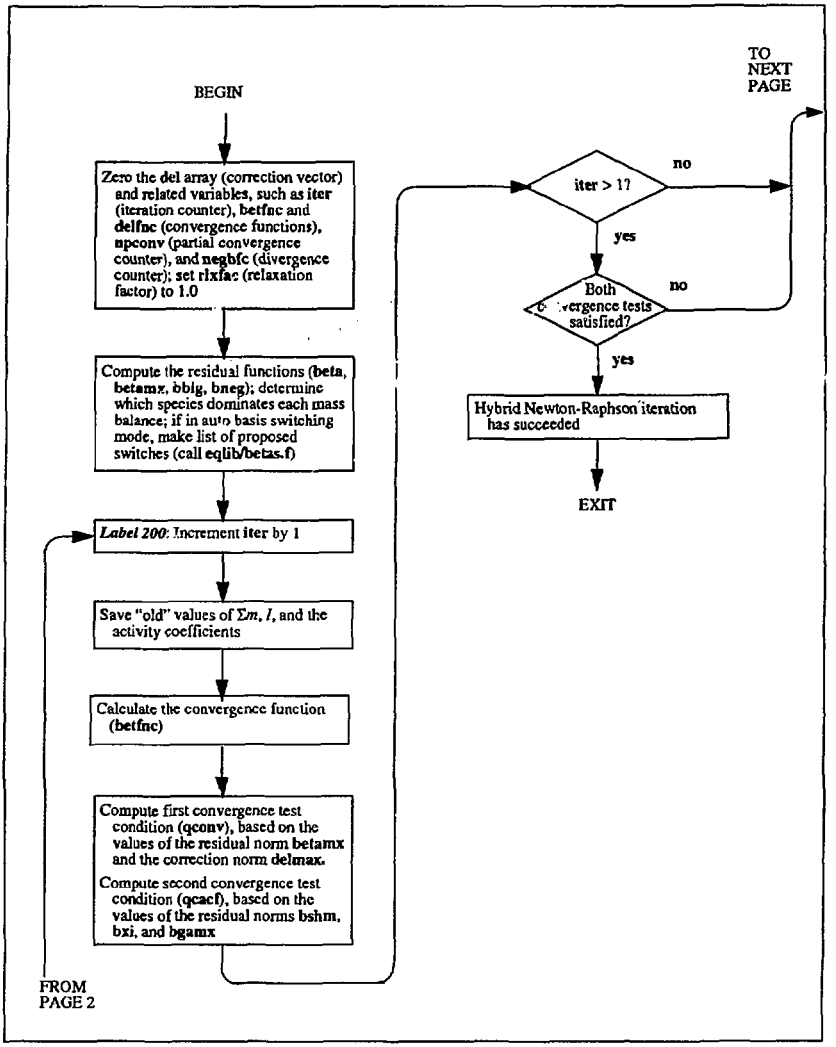

Figure 6 (page 1 of 2). Simplified flow diagram for hybrid Newton-Raphson iteration (from the viewpoint of the EQLIB module newton.f).

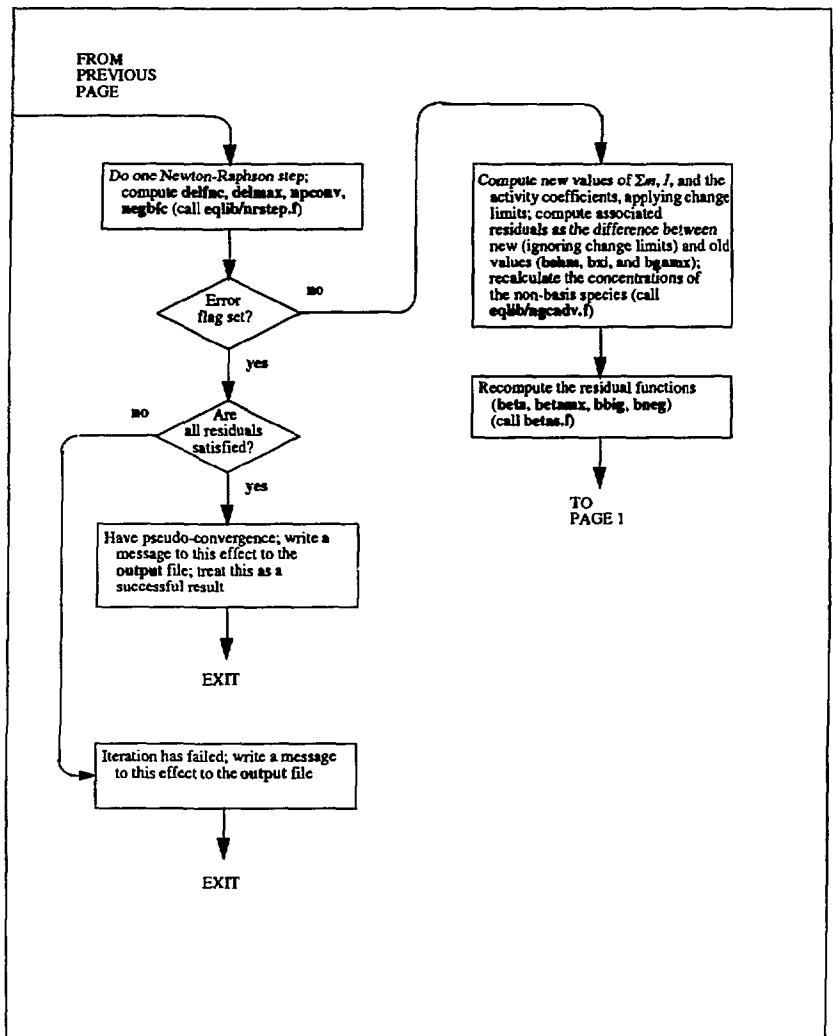

Figure 6 (continued, page 2 of 2). Simplified flow diagram for hybrid Newton-Raphson iteration (from the viewpoint of the EQLIB module **newton.f**).

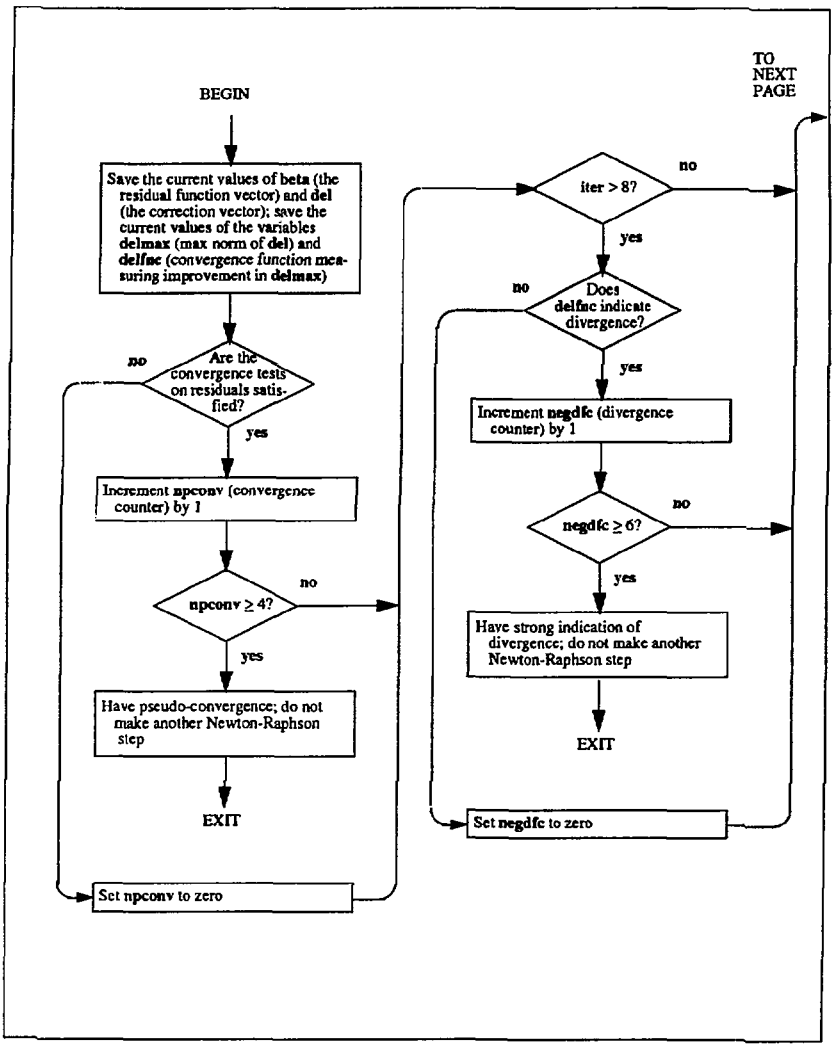

Figure 6 (page 1 of 4). Simplified flow diagram for a Newton-Raphson step (from the viewpoint of the EQLIB module nrstep.f).

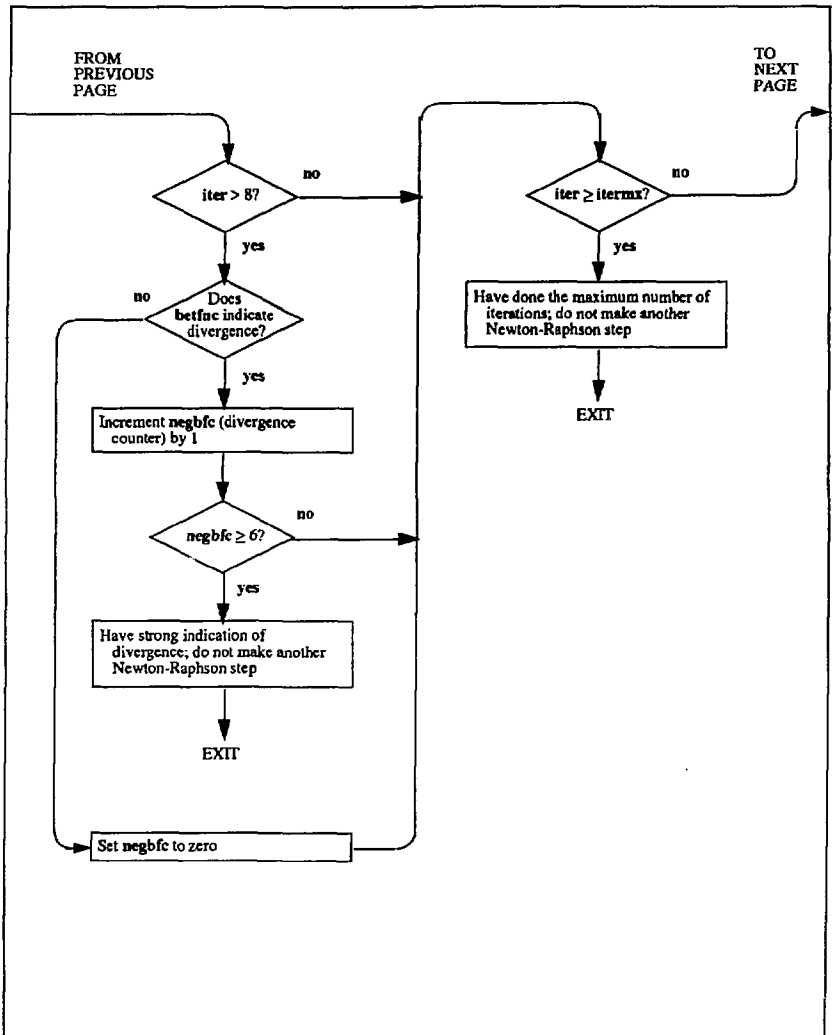

Figure 6 (page 2 of 4). Simplified flow diagram for a Newton-Raphson step (from the viewpoint of the EQLIB module **nrstep.f**).

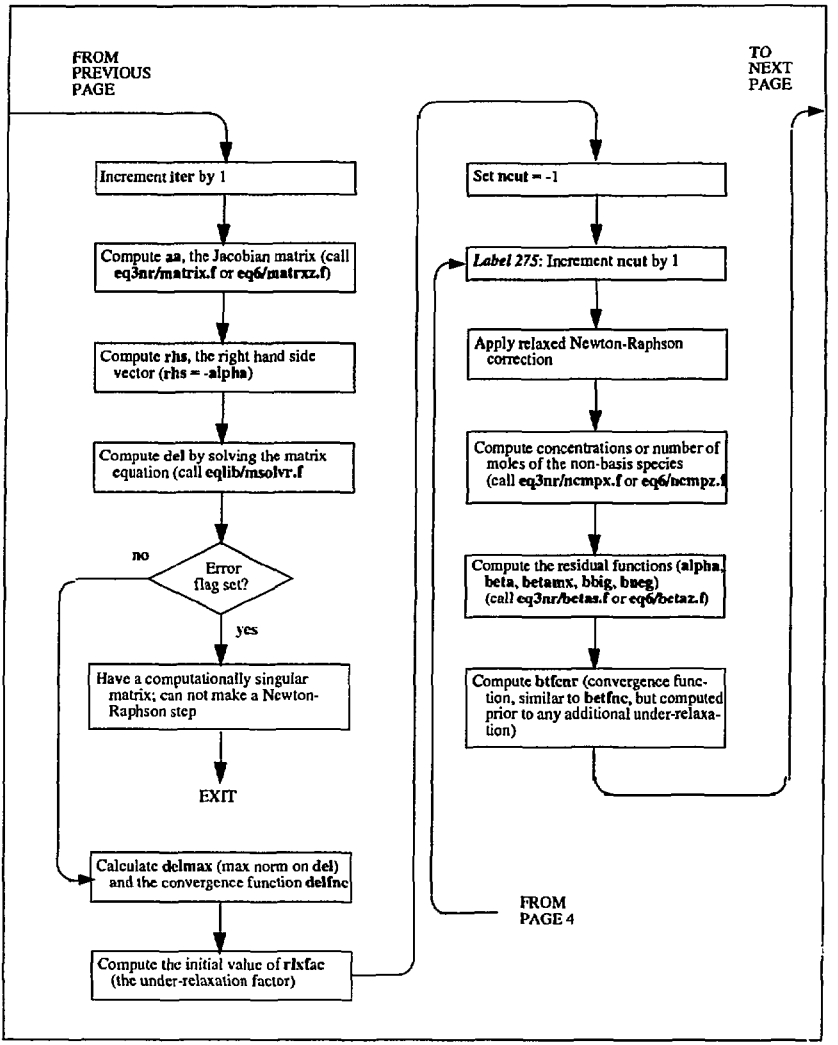

Figure 7 (page 3 of 4). Simplified flow diagram for a Newton-Raphson step (from the viewpoint of the EQLIB module **nrstep.f**).

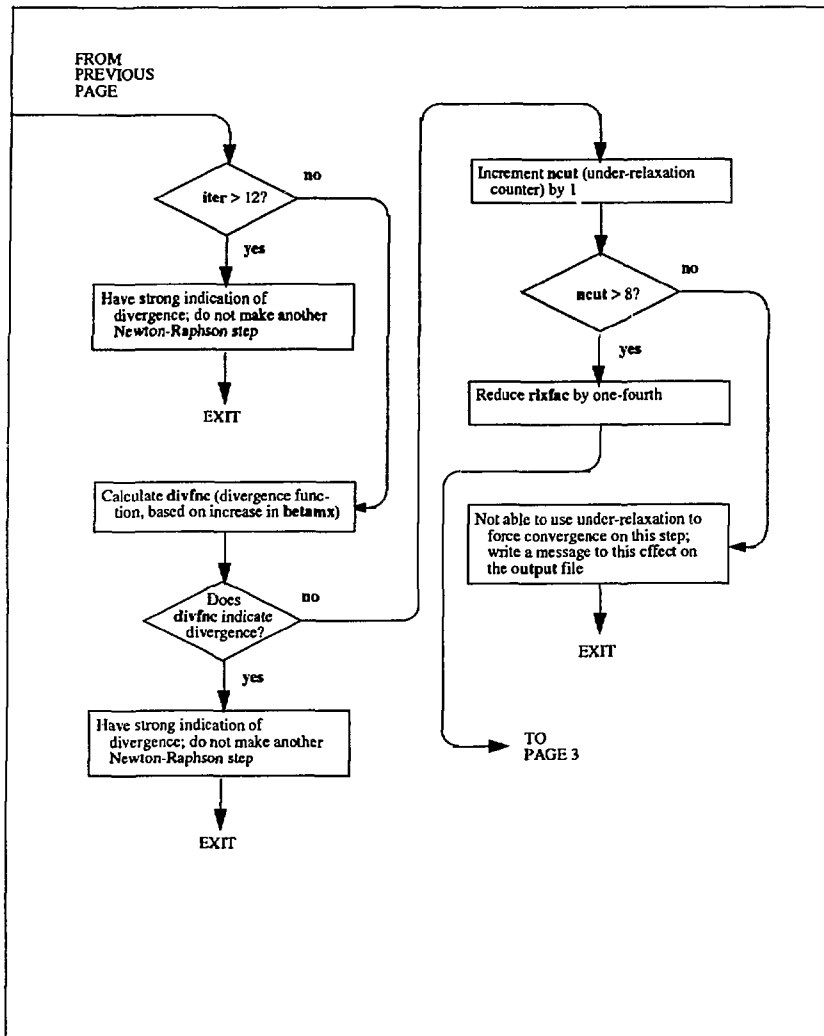

Figure 7 (page 4 of 4). Simplified flow diagram for a Newton-Raphson step (from the viewpoint of the EQLIB module nrstep.f).

Acknowledgments

Thanks are due to many individuals for their contributions in one form or another to the development of EQ3/6. Among these are Roger Aines, Carol Bruton, Bill Bourcier, Manny Clinnick, Paul Cloke, Stephanie Daveler, Joan Delany, Don Emerson, Mandy Goldner, Bob Herrick, Dana Isherwood, Ken Jackson, Jim Johnson, Suzanne Lundeen, Bill McKenzie, Judith Moody, Miki Moore, Ignasi Puigdomenech, Larry Ramspott, Terry Steinborn, and Brian Viani. I thank Brian Viani for his technical review of this report, which led to a number of improvements. I also thank Jim Blink for his comments on the manuscript.

References

Ananthaswamy J., and Atkinson, G., 1984, Thermodynamics of concentrated electrolyte mixtures. 4. Pitzer-Debye-Hückel limiting slopes for water from 0 to 100°C and from 1 atm to 1 kbar: Journal of Chemical and Engineering Data, v. 29, p. 81-87.

Archer, D. G., 1990, Effect of revisions of Debye-Hückel limiting law coefficients on the thermodynamic parameters for strong-electrolyte solutions: Journal of Chemical and Engineering Data, v. 35, p. 340-344.

Baes, C. F., Jr., and Mesmer, R. E., 1976, The Hydrolysis of Cations: Wiley Interscience, New York.

Ball, J. W., Jenne, E. A., and Nordstrom, D. K., 1979, WATEQ2: A computerized chemical model for trace and major element speciation and mineral equilibria for natural water, *in* Jenne, E. A., editor, Chemical Modeling in Aqueous Systems, American Chemical Society Symposium Series, v. 93, American Chemical Society, Washington, D. C., p. 815-835.

Barta, L., and Bradley, D. J., 1985, Extension of the specific interaction model to include gas solubilities in high temperature brines: Geochimica et Cosmochimica Acta, v. 49, p. 195-203.

Bassett, R. L., and Melchior, D. C., 1990, Chemical modeling of aqueous systems: An overview, *in* Melchior, D. C., and Bassett, R. L., editors, Chemical Modeling of Aqueous Systems II, American Chemical Society Symposium Series, v. 416, American Chemical Society, Washington, D. C., p. 1-14.

Bates, R. G., 1964, determination of pH: Wiley and Sons, New York.

Bates, R. G., 1973, Ion activity scales for use with ion-selective electrodes: Pure and Applied Chemistry, v. 36, p. 407-420.

Bates, R. G., and Alfenaar, M., 1969, Activity standards for ion-selective electrodes, *in* Durst, R. A., editor, Ion-Selective Electrodes, Proceedings of an NBS Symposium, NBS Special Publication 314, p. 191-214.

Bates, R. G., and Robinson, R. A., 1974, An approach to conventional scales of ionic activity for the standardization of ion-selective electrodes: Pure and Applied Chemistry, v. 37, p. 575-577.

Ben-Yaakov, S., and Sass, E., 1977, Independent estimate of the pH of Dead Sea brine: Limnology and Oceanography, v. 22, p. 374-376.

Berner, R. A., 1971, Principles of Chemical Sedimentology: McGraw-Hill, New York.

Bourcier, W. L., 1985, Improvements in the Solid Solution Modeling Capabilities of the EQ3/6 Geochemical Code: UCID-20587, Lawrence Livermore National Laboratory, Livermore, California.

Bourcier, W. L., 1989, Improvements in the solid solution modeling capabilities of EQ3/6, *in* Jackson, K. J., and Bourcier, W. L, editors, Proceedings of the Workshop on Geochemical Modeling, CONF-8609134, Lawrence Livermore National Laboratory, Livermore, California p. 41-48.

Busenberg, E., 1978, The products of the interaction of feldspars with aqueous solutions at 25°C. Geochimica et Cosmochimica Acta, v. 42, p. 1679-1686.

Carnahan, B., Luther, H. A., and Wilkes, J. O., 1969, Applied Numerical Methods: John Wiley & Sones, New York.

Carothers, W. W., and Kharaka, Y. K., 1978, Aliphatic acid anions in oil-field waters- Implications for origin of natural gas: The American Association of Petroleum Geologists Bulletin, v. 62, p. 2441-2453.

Chen, C.-T. A., and Marshall, W. L., 1981, Amorphous silica solubilities IV. Behavior in pure water and aqueous sodium chloride, sodium sulfate, magnesium chloride, and magnesium sulfate solutions up to 350°C: Geochimica et Cosmochimica Acta, v. 46, p. 279-287.

Clegg, S. L., and Brimblecombe, P., 1989, Solubility of ammonia in pure aqueous and multicomponent solutions: Journal of Physical Chemistry, v. 93, p. 7237-7248.

Clegg, S. L., and Brimblecombe, P., 1990, Solubility of volatile electrolytes in multicomponent solutions with atmospheric applications, *in* Melchior, D. C., and Bassett, R. L., editors, Chemical Modeling of Aqueous Systems II, American Chemical Society Symposium Series, v. 416, American Chemical Society, Washington, D. C., p. 58-73.

Connaughton, L. M., Millero, F. J., and Pitzer, K. S., 1989, Volume changes for mixing the major sea salts: equations valid to ionic strength 3.0 and temperature 95°C: Journal of Solution Chemistry, v. 18, p. 1007-1017.

Crerar, D. A., 1975, A method for computing multicomponent equilibria based on equilibrium constants, Geochimica et Cosmochimica Acta, v. 39, p. 1375-1384.

Daveler, S. A., and Wolery, T. J., 1992, EQPT, A Data File Preprocessor for the EQ3/6 Software Package: User's Guide and Related Documentation (Version 7.0): UCRL-MA-110662-PT-II, Lawrence Livermore National Laboratory, Livermore, California.

Davies, C. W., 1962, Ion Association: Butterworths, London.

Delany, J. M., and Lundeen, S. R., 1991, The LLNL Thermochemical Data Base- Revised Data and File Format for the EQ3/6 Package: UCID-21658, Lawrence Livermore National Laboratory, Livermore, California.

Drummond, S. E., Jr., 1981, Boiling and Mixing of Hydrothermal Fluids: Chemical Effects on Mineral Precipitation: Ph.D. thesis, The Pennsylvania State University, University Park, Pennsylvania.

Felmy, A. R., and Weare, J. H., 1986, The prediction of borate mineral equilibria in natural waters: Application to Searles Lake, California: Geochimica et Cosmochimica Acta, v. 50, p. 2771-2783.

Freeze, R. A., and Cherry, J. A., 1979, Groundwater: Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Garrels, R. M., and Christ, C. L., 1965, Solutions, Minerals and Equilibria: Freeman, Cooper and Company, San Francisco.

Garrels, R. M., and Thompson, M. E., 1962, A chemical model for sea water at 25°C and one atmosphere total pressure: American Journal of Science, v. 260, p. 57-66.

Greenberg, J. P., and Møller, N., 1989, The prediction of mineral solubilities in natural waters: A chemical equilibrium model for the Na-K-Ca-Cl-SO_4-H_2O system to high concentration from 0 to 250°C: Geochimica et Cosmochimica Acta, v. 53, p. 2503-2518.

Grenthe, I., Lemire, R. J., Muller, A. B., Nguyen-Trung, C., and Wanner, H., 1989, NEA-TDB Chemical Thermodynamics of Uranium, OECD-Nuclear Energy Agency, Saclay, France (draft manuscript; NNA.900815.0013).

Grenthe, I., Fuger, J., Lemire, R. J., Muller, A. B., Nguyen-Trung, C., and Wanner, H., editors, 1992, Chemical Thermodynamics of Uranium: Elsevier Science Publishing Company, New York.

Harned, H. S., and Owen, B. B., 1958, The Physical Chemistry of Electrolyte Solutions, 3rd Edition: ACS Monograph 137, Reinhold, New York.

Harvie, C. E., 1981, Theoretical Investigations in Geochemistry and Atom Surface Scattering: Ph.D. thesis, University of California, San Diego.

Harvie, C. E., Møller, 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 concentrations at 25°C: Geochimica et Cosmochimica Acta, v. 44, p. 981-997.

Helgeson, H. C., 1967, Thermodynamics of complex dissociation in aqueous solutions at elevated temperatures: Journal of Physical Chemistry, v. 71, p. 3121-3136.

Helgeson, H. C., 1969, Thermodynamics of hydrothermal systems at elevated temperatures and pressures: American Journal of Science, v. 267, p. 729-804.

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. 569-592-877.

Helgeson, H. C., Delaney, J. M., Nesbitt, H. W., and Bird, D. K., 1978, Summary and critique of the thermodynamic properties of rock-forming minerals: American Journal of Science, v. 278-A, p. 1-229.

Helgeson, H. C., and Kirkham, D. H., 1974a, Theoretical prediction of the thermodynamic behavior of aqueous electrolytes at high pressures and temperatures: I. Summary of the thermodynamic/electrostatic properties of the solvent: American Journal of Science, v. 274, p. 1089-1198.

Helgeson, H. C., and Kirkham, D. H., 1974b, Theoretical prediction of the thermodynamic behavior of aqueous electrolytes at high pressures and temperatures: II. Debye-Hückel parameters for activity coefficients and relative partial-molal properties: American Journal of Science, v. 274, p. 1199-1261.

Helgeson, H. C., and Kirkham, D. H., 1976, Theoretical prediction of the thermodynamic behavior of aqueous electrolytes at high pressures and temperatures: IV. Calculation of activity coefficients, osmotic coefficients, and apparent molal and standard and relative partial molal properties to 600°C and 5 kb: American Journal of Science, v. 281, p. 1249-1516.

Helgeson, H. C., Kirkham, D. H., and Flowers, G. C., 1981, Theoretical prediction of the thermodynamic behavior of aqueous electrolytes at high pressures and temperatures: III. Equation of state for aqueous species at infinite dilution: American Journal of Science, v. 276, p. 97-240.

Henley, R. W., Truesdell, A. H., Barton, P. B., Jr., and Whitney, J. A., 1984, Fluid-Mineral Equilibria in Hydrothermal Systems: Reviews in Economic Geology, v. 1, Society of Economic Geologists, The Economic Geology Publishing Company, El Paso, Texas.

Holmes, H. F., and Mesmer, R. E., 1986a, Isopiestic studies of aqueous solutions at elevated temperatures VIII. The alkali-metal sulfates: Journal of Chemical Thermodynamics, v. 18, p. 263-275.

Holmes, H. F., and Mesmer, R. E., 1986b, Thermodynamics of aqueous solutions of the alkali metal sulfates: Journal of Solution Chemistry, v. 15, p. 495-518.

Hostetler, J. D., 1984, Electrode electrons, aqueous electrons, and redox potentials in natural waters: American Journal of Science, v. 284, p. 734-759.

Huang, H.-H., 1989, Estimation of Pitzer's ion interaction parameters for electrolytes involved in complex formation using a chemical equilibrium model: Journal of Solution Chemistry, v. 18, p. 1069-1084.

INTERA Environmental Consultants, Inc., 1983, *Geochemical Models Suitable for Performance Assessment of Nuclear Waste Storage: Comparison of PHREEQE and EQ3/EQ6*: ONWI-473, Office of Nuclear Waste Isolation, Battelle Project Management Division, Columbus, Ohio.

Jackson, K. J., 1988, Verification and Validation Studies of the Addition of Pitzer's Equations to the EQ3/6 Brine Model: UCRL-53841, Lawrence Livermore National Laboratory, Livermore, California.

Jenne, E. A., 1981, Geochemical Modeling: A Review: PNL-3574, Battelle Pacific Northwest Laboratory, Richland, Washington.

Johnson, J. W., and Norton, D., 1991, Critical phenomena in hydrothermal systems: State, thermodynamic, electrostatic, and transport properties of H_2O in the critical region: American Journal of Science, v. 291, p. 541-648.

Johnson, J. W., Oelkers, E. H., and Helgeson, *H. C., 1992, SUPCRT92: A software package for calculating the standard molal thermodynamic properties of minerals, gases, aqueous species, and reactions from 1 to 5000 bars and 0° to 1000°C: Computers and Geosciences, v. 18, p. 899-947.*

Kharaka, Y. K., and Barnes, I., 1973, SOLMNEQ: Solution-Mineral Equilibrium Computations: United States Geological Survey Computer Contributions Publication 215-899.

Knauss, K. G., and Wolery, T. J., 1986, Dependence of albite dissolution kinetics on pH and time at 25°C and 70°C: Geochimica et Cosmochimica Acta, v. 50, p. 2481-2497.

Knauss, K. G., Wolery, T. J., and Jackson, K. J., 1990, A new approach to measuring pH in brines and other concentrated electrolytes: Geochimica et Cosmochimica Acta, v. 54, p. 1519-1523.

Knauss, K. G., Wolery, T. J., and Jackson, K. J., 1991, Reply to comment by R. E. Mesmer on "A new approach to measuring pH in brines and other concentrated electrolytes": Geochimica et Cosmochimica Acta, v. 55, p. 1177-1179.

Kodytek, V., and Dolejs, V., 1986, On the Pitzer equation for the osmotic coefficient of aqueous solutions of 3:1 rare earth electrolytes: Zeitschrift für Physikalische Chemie (Leipzig), v. 267, p. 743-746.

Krauskopf, K., 1967, Introduction to Geochemistry: McGraw-Hill, New York.

Kumar, A., 1986, Prediction of densities of concentrated brines by Pitzer theory: Journal of Chemical and Engineering Data, v. 31, p. 19-20.

Laxen, D. P. H, and Chandler, I. M., 1982, Comparison of filtration techniques for size distribution in freshwaters: Analytical Chemistry, v. 54, p. 1350-1355.

Langmuir, D., 1979, Techniques of estimating thermodynamic properties for some aqueous complexes of geochemical interest, *in* Jenne, E. A., editor, Chemical Modeling in Aqueous Systems, American Chemical Society Symposium Series, v. 93, American Chemical Society, Washington, D. C., p. 353-387.

Lindberg, R. D., and Runnells, D. D., 1984, Ground water redox reactions: An analysis of equilibrium state applied to Eh measurements and geochemical modeling: Science, v. 225, p. 925-927.

MacInnes, D. A., 1919, The activities of the ions of strong electrolytes: Journal of the Chemical Society, v. 41, p. 1086-1092.

Marcus, Y., 1977, The activities of potassium chloride and of water in Dead Sea brine: Geochimica et Cosmochimica Acta, v. 41, p. 1739-1744.

Merino, E., 1979, Internal consistency of a water analysis and uncertainty of the calculated distribution of aqueous species at 25°C: Geochimica et Cosmochimica Acta, v. 43, p. 1533-1542.

Mesmer, R. E., 1991, Comments on "A new approach to measuring pH in brines and other concentrated electrolytes" by K. G. Knauss, T. J. Wolery, and K. J. Jackson: Geochimica et Cosmochimica Acta, v. 55, p. 1175-1176.

Millero, F. J., 1977, The use of the specific interaction model to estimate the partial molal volumes of electrolytes in seawater: Geochimica et Cosmochimica Acta, v. 41, p. 215-223.

Millero, F. J., and Byrne, R. H., 1984, Use of Pitzer's equations to determine the media effect on the formation of lead chloro complexes: Geochimica et Cosmochimica Acta, v. 48, p. 1145-1150.

Monnin, C., 1989, An ion interaction model for the volumetric properties of natural waters: Density of the solution and partial molal volumes of electrolytes to high concentrations at 25°C: Geochimica et Cosmochimica Acta, v. 53, p. 1177-1188.

Monnin, C., and Schott, J., 1984, Determination of the solubility products of sodium carbonate minerals and an application to trona deposition in Lake Magadi (Kenya): Geochimica et Cosmochimica Acta, v. 48, p. 571-581.

Morris, J. C., and Stumm, W., 1967, Redox equilibria and measurements of potentials in the aquatic environment, *in* Gould, R. F., editor, Equilibrium Concepts in Natural Water Systems, Advances in Chemistry Series, v. 67, American Chemical Society, Washington, D. C., p. 270-285.

Nordstrom, D. K., and Jenne, E. A., 1977, Fluorite solubility equilibria in selected geothermal waters: Geochimica et Cosmochimica Acta, v. 41, p. 175-188.

Nordstrom, D. K., Jenne, E. A., and Ball, J. W., 1979, Redox equilibria of iron in acid mine waters, *in* Jenne, E. A., editor, Chemical Modeling in Aqueous Systems, American Chemical Society Symposium Series, v. 93, American Chemical Society, Washington, D. C., p. 51-79.

Nordstrom, D. K. and Munoz, J. L., 1985, Geochemical Thermodynamics: Benjamin/Cummings Publishing Company, Inc., Menlo Park, California, 477 p.

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* Jenne, E. A., editor, Chemical Modeling in Aqueous Systems, American Chemical Society Symposium Series, v. 93, American Chemical Society, Washington, D. C. p. 857-892.

Pabalan, R. T., and Pitzer, K. S., 1987, Thermodynamics of concentrated electrolyte mixtures and the prediction of mineral solubilities to high temperatures for mixtures in the system Na-K-Ca-Mg-Cl-SO_4-OH-H_2O: Geochimica et Cosmochimica Acta, v. 51, p. 2429-2443.

Pabalan, R. T., and Pitzer, K. S., 1988, Heat capacity and other thermodynamic properties of $Na_2SO_{4(aq)}$ in hydrothermal solutions and the solubilities of sodium sulfate minerals in the system Na-Cl-SO_4-OH-H_2O: Geochimica et Cosmochimica Acta, v. 52, p. 2393-2904.

Pabalan, R. T., and Pitzer, K. S., 1990, Models for aqueous electrolyte mixtures for systems extending from dilute solutions to fused salts, *in* Melchior, D. C., and Bassett, R. L., editors, Chemical Modeling of Aqueous Systems

II, American Chemical Society Symposium Series, v. 416, American Chemical Society, Washington, D. C., p. 44-57.

Parkhurst, D. L., Thorstenson, D. C., and Plummer, L. N., 1980, PHREEQE - A Computer Program for Geochemical Calculations: United States Geological Survey Water Resources Investigations 80-96.

Pitzer, K. S., 1973, Thermodynamics of electrolytes - I. Theoretical basis and general equations: Journal of Physical Chemistry, v. 77, p. 268-277.

Pitzer, K. S., 1975, Thermodynamics of electrolytes. V. Effects of higher-order electrostatic terms: Journal of Solution Chemistry, v. 4, p. 249-265.

Pitzer, K. S., 1979, Theory: ion interaction approach, *in* Pytkowicz, R. M., editor, Activity Coefficients in Electrolyte Solutions, CRC Press, Boca Raton, Florida, p. 157-208.

Pitzer, K. S., 1984, A consideration of Pitzer's equations for activity and osmotic coefficients in mixed electrolytes: Journal of the Chemical Society, Faraday Transactions I, v. 80, p. 3451-3454.

Pitzer, K. S., 1987, Thermodynamic model for aqueous solutions of liquid-like density, *in* Carmichael, I. S. E., and Eugster, H. P., editors, Thermodynamic Modeling of Geological Materials: Minerals, Fluids and Melts, Reviews in Mineralogy, v. 17, Mineralogical Society of America, Washington, D. C., p. 97-142

Pitzer, K. S., 1992, Ion interaction approach: Theory and Data Correlation, *in* Pitzer, K. S., editor, Activity Coefficients in Electrolyte Solutions, 2nd edition, CRC Press, Boca Raton, Florida, p. 75-153.

Pitzer, K. S., and Kim, J. J., 1974, Thermodynamics of electrolytes. IV. Activity and osmotic coefficients for mixed electrolytes: Journal of the American Chemical Society, v. 96, p. 5701-5707.

Pitzer, K. S., and Mayorga, G., 1974, Thermodynamics of electrolytes. III. Activity and osmotic coefficients for 2-2 electrolytes: Journal of Solution Chemistry, v. 3, p. 539-546.

Pitzer, K. S., Roy, R. N., and Silvester, L. F., 1977, Thermodynamics of electrolytes. 7. Sulfuric acid: Journal of the American Chemical Society, v. 99, p. 4930-4936.

Pitzer, K. S., and Silvester, L. F., 1976, Thermodynamics of electrolytes. VI. Weak electrolytes including H_3PO_4: Journal of Solution Chemistry, v. 5, p. 269-278.

Plummer, L. N., and Parkhurst, D. L., 1990, Application of the Pitzer equations to the PHREEQE geochemical model, *in* Melchior, D. C., and Bassett, R. L., editors, Chemical Modeling of Aqueous Systems II, American Chemical Society Symposium Series, v. 416, American Chemical Society, Washington, D. C., p. 128-137.

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: United States Geological Survey Water-Resources Investigations Report 88-4153, 310 p.

Prigogine, I., and Defay, R., 1954, Chemical Thermodynamics: Longmans, Green, and Company, London.

Reardon, E. J., and Langmuir, D., 1976, Activity coefficients of $MgSO_4^o$ and $CaSO_4^o$ ion pairs as a function of ionic strength: Geochimica et Cosmochimica Acta, v. 40, p. 549-554.

Saxena, S. R., 1973, Thermodynamics of Rock-Forming Crystalline Solutions: Springer-Verlag, New York.

Shock, E. L., and Helgeson, H. C., 1988, Calculation of the thermodynamic and transport properties of aqueous species at high pressures and temperatures: Correlation algorithms for ionic species and equation of state predictions to 5 kb and 1000°C: Geochimica et Cosmochimica Acta, v. 52, p. 2009-2036.

Shock, E. L., and Helgeson, H. C., 1989, Corrections to Shock and Helgeson (1988) *Geochimica et Cosmochimica Acta* 52, 2009-2036: Geochimica et Cosmochimica Acta, v. 53, p. 215.

Shock, E. L., and Helgeson, H. C., 1990, Calculation of the thermodynamic and transport properties of aqueous species at high pressures and temperatures: Standard partial molal properties of organic species: Geochimica et Cosmochimica Acta, v. 54, p. 915-945.

Shock, E. L., Helgeson, H. C., and Sverjensky, D. A., 1989, Calculation of the thermodynamic and transport properties of aqueous species at high pressures and temperatures: Standard partial molal properties of inorganic species: Geochimica et Cosmochimica Acta, v. 53, p. 2157-2183.

Shock, E. L., Oelkers, E. H., Johnson, J. W., Sverjensky, D. A., and Helgeson, H. C., 1992, Calculation of the thermodynamic properties of aqueous species at high pressures and temperatures: Effective electrostatic radii, dissociation constants and standard partial molal properties to 1000°C and 5 kbar: Journal of the Chemical Society Faraday Transactions, v. 88, p. 803-826.

Silling, S. A., 1983, Final Technical Position on Documentation of Computer Codes for High-Level Waste Management: NUREG-0856, United States Nuclear Regulatory Commission, Washington, D. C.

Silvester, L. F., and Pitzer, K. S., 1977, Thermodynamics of electrolytes. 8. Enthalpy and the effect of temperature on the activity coefficients, Journal of Solution Chemistry, v. 7, p. 327-337.

Silvester, L. F., and Pitzer, K. S., 1978, Thermodynamics of electrolytes. X. High-temperature properties, including enthalpy an heat capacity, with application to sodium chloride: Journal of Physical Chemistry, v. 7, p. 327-337.

Simonson, J. M., Roy, R. N., Roy L. N., and Johnson, D. A., 1987, The thermodynamics of aqueous borate solutions I. Mixtures of boric acid with sodium or potassium borate and chloride: Journal of Solution Chemistry, v. 16, p. 791-803.

Spencer, R. J., Møller, N., and Weare, J. H., 1990, The prediction of mineral solubilities in natural waters: A chemical equilibrium model for the Na-K-Ca-Mg-Cl-SO_4-H_2O system at temperatures below 25°C: Geochimica et Cosmochimica Acta, v. 54, p. 575-590.

Standard Methods for the Examination of Water and Wastewater, 14th edition, 1976, prepared and published jointly by the American Public Health Association, American Water Works Association, and Water Pollution Control Federation: American Public Health Association, Washington, D. C.

Stumm, W., and Morgan, J. J., 1981, Aquatic Chemistry, second edition: John Wiley and Sons, New York.

Tanger, J. C., and Helgeson, H. C., 1988, Calculation of the thermodynamic and transport properties of aqueous species at high pressures and temperatures: Revised equations of state for the standard partial molal properties of ions and electrolytes: American Journal of Science, v. 288, p. 19-98.

Tardy, Y., and Garrels, R. M., 1974, A method of estimating the Gibbs energies of formation of layer silicates. Geochimica et Cosmochimica Acta, v. 38, p. 1101-1116.

Thorstenson, D. C., 1970, Equilibrium distribution of small organic molecules in natural waters: Geochimica et Cosmochimica Acta, v. 34, p. 745-770.

Thorstenson, D. C., 1983, The concept of electron activity and its relation to redox potentials in aqueous geochemical systems: United States Geological Survey Open-File Report 84-072 (available from Open-File Services Section, Western Distribution Branch, Box 25425, Federal Center, Denver, CO 80225).

Truesdell, A. H., 1968, The advantage of using pe rather than Eh in redox equilibrium calculations: Journal of Geological Education, v. 16, p. 17-20.

Van Zeggeren, F., and Storey, S. H., 1970, The Computation of Chemical Equilibrium: Cambridge University Press.

Viani, B. E., and Bruton, C. J., 1992, Modeling Fluid-Rock Interaction at Yucca Mountain, Nevada: A Progress Report: UCRL-ID-109921, Lawrence Livermore National Laboratory, Livermore, California.

Willey, L. M., Kharaka, Y. K., Presser, T. S., Rapp, J. B., and Barnes, I., 1975, Short chain aliphatic acid anions in oil field waters and their contribution to the measured alkalinity: Geochimica et Cosmochimica Acta, v. 39, p. 1707-1711.

Wolery, T. J., 1978, Some Chemical Aspects of Hydrothermal Processes at Mid-Ocean Ridges-A Theoretical Study: Ph.D. thesis, Northwestern University, Evanston, Illinois.

Wolery, T. J., 1979, Calculation of Chemical Equilibrium between Aqueous Solutions and Minerals: The EQ3/6 Software Package: UCRL-52658, Lawrence Livermore National Laboratory, Livermore, California.

Wolery, T. J., 1983, EQ3NR, A Computer Program for Geochemical Aqueous Speciation-Solubility Calculations. *User's Guide and Documentation:* UCRL-53414, Lawrence Livermore National Laboratory, Livermore, California.

Wolery, T. J.,1990, On the thermodynamic framework of solutions (with special reference to aqueous electrolyte solutions: American Journal of Science, v. 290, p. 296-320.

Wolery, T. J., 1992, EQ3/6, A Software Package for Geochemical Modeling of Aqueous Systems: Package Overview and Installation Guide (Version 7.0): UCRL-MA-110662-PT-I, Lawrence Livermore National Laboratory, Livermore, California.

Wolery, T. J., and Daveler, S. A., 1992, EQ6, A Computer Code for Reaction-Path Modeling of Aqueous Geochemical Systems (Version 7.0): Theoretical Manual, User's Guide, and Related Documentation: UCRL-MA-110662-PT-IV, Lawrence Livermore National Laboratory, Livermore, California.

Wolery, T. J., Jackson, K. J., Bourcier, W. L., Bruton, C. J., Viani, B. E., Knauss, K. G., and Delany, J. M., 1990, Current status of the EQ3/6 software package for geochemical modeling, *in* Melchior, D. C., and Bassett, R. L., editors, Chemical Modeling of Aqueous Systems II, American Chemical Society Symposium Series, v. 416, American Chemical Society, Washington, D. C., p. 104-116.

Wolery, T. J., and Walters, L. J., Jr., 1975, Calculation of equilibrium distributions of chemical species in aqueous solutions by means of monotone sequences: Mathematical Geology, v. 7, p. 99-115.

Wood, B. J., and Fraser, D. G., 1977, Elementary Thermodynamics for Geologists: Oxford University Press, Oxford, United Kingdom.

Appendix A: Glossary of Major Variables in EQ3NR

This glossary covers the major variables in EQ3NR. Almost all of these are also shared by EQ6 (Wolery and Daveler, 1992). Those which are unique to EQ3NR are so marked. This glossary does not include variables which are of a purely local nature, such as a floating point variable used to accumulate sums in a do loop. Logarithmic quantities are frequently used. All refer to base ten, unless otherwise specified.

The variable names may be preceded by the corresponding algebraic symbols used in this report, if any. Following the conventions used universally in the EQ3/6 package, variables beginning with **i**, **j**, **k** or **n** are integer, those beginning with **q** are logical, and those beginning with **u** are character variables. Variables beginning with any other letters, including **l** and **m**, should be **real*8.** On 32-bit machines, this corresponds to **double precision**; on 64-bit machines, this corresponds to **single precision.**

Some of the entries in this glossary are described as either "data file ordered" or "input file ordered". The former refers to the indexing of arrays that have been filled by reading the data file, **data1.** The latter refers to the indexing of arrays that have been filled by reading the **input** file. Since the two forms of indexing are not equivalent, EQ3NR must correlate them. This is done by mapping "input file ordered" data into a form which corresponds to "data file ordering," which is the ordering which is used in the code calculations. Most data arrays which are "input file ordered" are therefore treated as holding arrays and play no further role in the calculation once the data in them has been mapped to corresponding arrays which are "data file ordered."

Most arrays in EQ3/6 are dimensioned using FORTRAN parameters. The value of a parameter is assigned once in a code (in a PARAMETER statement) and can not be changed elsewhere. Some FORTRAN compilers permit parameters to be passed through calling sequences, but others do not allow this, as a means of protecting the parameter's assigned value. A called module often requires the dimensions of passed arrays. This is handled in EQ3/6 by the use of "dimensioning variables," which are ordinary integer variables whose values have been set equal to the corresponding parameters. The names of the dimensioning parameters in EQ3/6 typically end in **par**, **pa**, or **pa1**. The names of the corresponding dimensioning variables end, respectively, in **max**, **mx**, and **mx1**. Thus, the dimension of the **aa** matrix is given by the pair **kpar/kmax.** The maximum number of aqueous species is given by **nstpar/nstmax.** The parameter suffixes are used exclusively by dimensioning parameters. However, the corresponding variable suffixes are not used exclusively by dimensioning variables. For example, **itermx** is the maximum number of Newton-Raphson iterations.

J	**aa**	The Jacobian matrix. Dimensioning: **aa(kpar,kpar).** Usage: **aa(krow,kcol)** is the element corresponding to the **krow**-th row and the **kcol**-th column. Note: In EQ3NR, this is also used as the matrix that is required for simultaneous solutions in pre-Newton-Raphson optimization.
	aadh	Interpolating polynomial coefficients for computing the Debye-Hückel parameter $A_{\gamma,10}$ (**adh**) as a function of temperature. Dimensioning: **aadh(narxpa,ntprpa).** Usage: **aadh(narx,ntpr)** refers to the **narx**-th coefficient for the **ntpr**-th temperature range.

	aaphi	Interpolating polynomial coefficients for computing the Debye-Hückel parameter A_ϕ (**aphi**) as a function of temperature. Dimensioning: **aaphi(narxpa,ntprpa)**. Usage: **aaphi(narx,ntpr)** refers to the **narx**-th coefficient for the **ntpr**-th temperature range.
	abdh	Interpolating polynomial coefficients for computing the Debye-Hückel parameter B_γ (**bdh**) as a function of temperature. Dimensioning: **abdh(narxpa,ntprpa)**. Usage: **abdh(narx,ntpr)** refers to the **narx**-th coefficient for the **ntpr**-th temperature range.
	abdot	Interpolating polynomial coefficients for computing the extended Debye-Hückel parameter $\dot{B}$ (**abdot**) as a function of temperature. Dimensioning: **abdot(narxpa, ntprpa)**. Usage: **abdot(narx,ntpr)** refers to the **narx**-th coefficient for the **ntpr**-th temperature range.
a_w, a_i	**act**	The activity of an aqueous species. Dimensioning: **act(nstpar)**. Usage: **act(ns)** refers to the activity of the **ns**-th aqueous species.
$log\ a_w$, $log\ a_i$	**actlg**	The log activity of an aqueous species. Dimensioning: **actlg(nstpar)**. Usage: **actlg(ns)** refers to the log activity of the **ns**-th aqueous species.
$A_{\gamma,l0}$	**adh**	Debye-Hückel A_γ parameter for computing the base ten logarithm of the molal activity coefficient of an aqueous species.
	aeh	Interpolating polynomial coefficients for computing the special equilibrium constant $log\ K_{Eh}$ (**xlkeh**) as a function of temperature. Dimensioning: **aeh(narxpa, ntprpa)**. Usage: **aeh(narx,ntpr)** refers to the **narx**-th coefficient for the **ntpr**-th temperature range.
	afcnst	The affinity constant; equal to $0.001(2.303)RT$.
A, A_-	**aff**	Affinity of a pure mineral to precipitate, kcal. Dimensioning: **aff(nmtpar)**. Usage: **aff(nm)** refers to the affinity of the **nm**-th pure mineral.
A, A_-	**affcx**	Affinity of a solid solution phase to precipitate, kcal. Dimensioning: **affcx(nxtpar)**. Usage: **affcx(nx)** refers to the affinity of the **nx**-th solid solution.
A, A_-	**affx**	Affinity of an end-member component of a solid solution. Dimensioning: **affx(iktpar,xtpar)**. Usage: **affx(ik,nx)** refers to the affinity of the **ik**-th end-member component of the **nx**-th solid solution.
	ags	Interpolating polynomial coefficients for computing the array of equilibrium constants **xlkg**. Dimensioning: **ags(narxpa,ntprpa,ngtpar)**. Usage: **ags(narx,ntpr,ng)** refers to the **narx**-th coefficient for the **ntpr**-th temperature range for the **ng**-th gas species.
$\underline{\alpha}$	**alpha**	Residual function vector. Dimensioning: **alpha(kpar)**. Usage: **alpha(kcol)** refers to the **kcol**-th master iteration variable.
2.303	**al10**	*ln* 10; note- the value 2.303 is generally an insufficient approximation; this parameter should be calculated to the full machine precision to avoid problems with numerical consistency.

	amn	Interpolating polynomial coefficients for computing the array of equilibrium constants **xlkm**. Dimensioning: **amn(narxpa,ntprpa,nmtpar)**. Usage: **amn(narx, ntpr,nm)** refers to the **narx**-th coefficient for the **ntpr**-th temperature range for the **nm**-th pure mineral.
A_ϕ	**aphi**	Debye-Hückel A_ϕ parameter.
	apress	Interpolating polynomial coefficients for computing the pressure (**press**) as a function of temperature. Dimensioning: **apress(narxpa,ntprpa)**. Usage: **apress(narx, ntpr)** refers to the **narx**-th coefficient for the **ntpr**-th temperature range.
	ard	Holding array used to read in the interpolating polynomial coefficients used to compute equilibrium constants of various kinds of reactions. Dimensioning: **ard(narxpa,ntprpa)**. Usage: **ard(narx,ntpr)** is the **narx**-th coefficient for the **ntpr**-th temperature range of the current reaction.
$p_{k\psi}$	**apx**	Solid solution activity coefficient model parameters. Dimensioning: **apx(iapxpa,nxtpar)**. Usage: **apx(iapx,nx)** refers to the **iapx**-th coefficient for the **nx**-th solid solution.
	ars	Interpolating polynomial coefficients for computing the array of equilibrium constants **xlks**. Dimensioning: **ars(narxpa,ntprpa,nstpar)**. Usage: **ars(narx,ntpr,ns)** refers to the **narx**-th coefficient for the **ntpr**-th temperature range for the **ns**-th aqueous species.
	atwt	Atomic weight of a chemical element. Dimensioning: **atwt(nctpar)**. Usage: **atwt(nc)** refers to the **nc**-th chemical element.
$\dot{a}_i$	**azero**	Hard core diameter of an aqueous species. Dimensioning: **azero(nstpar)**. Usage: **azero(ns)** refers to the **ns**-th aqueous species.
	bbig	The mass balance residual (an element of the **beta** array) which has the most extreme positive value; if none is positive, this is assigned a value of zero. The name of the associated basis species is given in the variable **ubbig**.
B_γ	**bdh**	Debye-Hückel B_γ parameter.
$\underline{\beta}$	**beta**	Residual function vector, with mass balance elements normalized. Dimensioning: **beta(kpar)**. Usage: **beta(kcol)** refers to the **kcol**-th master iteration variable.
β_{func}	**betfnc**	Convergence function that measures the improvement in **betamx**, defined as (1.0 - (**betamx/betmxo**))
β_{max}	**betamx**	Max norm of $\underline{\beta}$.
	betmxo	The previous value of **betamx**.
$\dot{B}$	**bdot**	Extended Debye-Hückel B-dot parameter.
	bfac	Array of $(\beta+1)^{\frac{1}{n}}$ factors used in pre-Newton-Raphson optimization. Dimensioning: **bfac(kpar)**. Usage: **bfac(kcol)** refers to the **kcol**-th master iteration variable.

	bgamx	The max norm of activity coefficient residual functions; the name of the associated species is given in the variable **ubgamx**.
μ_{ijk}	**bmu**	Array of third order primitive virial coefficients of Pitzer's equations (25°C values). Dimensioning: **bmu(nmupar)**. Usage: **bmu(k)** is the coefficient for the **k**-th species triplet. See **dmu1, dmu2, nmu,** and **pmu.**
	bneg	The mass balance residual (an element of the **beta** array) which has the most extreme negative value; if none is negative, this is assigned a value of zero. The name of the associated basis species is given in the variable **ubneg.**
$s\lambda_{ij}^{(n)}$	**bslm**	Array of second order "short range" primitive virial coefficient parameters of Pitzer's equations (25°C values). Dimensioning: **bslm(3,nslpar).** Usage: **bslm(n,k)** is the ***n-th*** *such parameter* for the **k**-th species pair. See **dslm1, dslm2, nslmx, pslm,** and **pslam.**
b	**bt**	The b factor appearing in the Debye-Hückel term in Pitzer's equations; it is assigned a constant value of 1.2.
	cco2	Parameters of the Drummond (1981) equation for computing $log\gamma_{CO_{2(aq)}}$ as a function of temperature and ionic strength. Dimensioning: **cco2(5).** Usage: **cco2(i)** refers to the **i**-th coefficient.
b_{sg}	**cdrg**	Reaction coefficient for dissolution reactions of gas species. Dimensioning: **cdrg(nsqpa1,ngtpar).** Usage: **cdrg(ns,ng)** is the coefficient of the **ns**-th aqueous basis species appearing in the dissolution reaction for the **ng**-th gas; **cdrg(nsq1,ng)** is the coefficient of the **ng**-th gas itself.
$b_{s\phi}$	**cdrm**	Reaction coefficient for the dissolution reactions of pure minerals. Dimensioning: **cdrm(nsqpa1,nmtpar).** Usage: **cdrm(ns,nm)** is the coefficient of the **ns**-th aqueous basis species appearing in the dissolution reaction of the **nm**-th pure mineral; **cdrm(nsq1,nm)** is the coefficient of the **nm**-th pure mineral itself.
b_{sr}	**cdrs**	Reaction coefficient for aqueous redox and dissociation reactions. Dimensioning: **cdrs(nsqpa1,nrstpa).** Usage: **cdrs(ns,nrs)** is the coefficient of the **ns**-th aqueous basis species appearing in the **nrs**-th aqueous dissociation/destruction reaction; **cdrg(nsq1,nrs)** is the coefficient of the associated species (of index **nsc = nrs + nsb**). If the **ns**-th species is an auxiliary basis species and **nrs** is its own formally associated reaction, **cdrs(ns,nrs) = 0** and **cdrs(nsq1,nrs)** is its coefficient.
	cdrsd	*Holding array used to read in the coefficients of reactions listed on the data file.* Dimensioning: **cdrsd(nsqpa1).** Usage: **cdrsd(n)** is the **n**-th coefficient in the current reaction as it is written on the data file; it corresponds to the species whose name is **udrxd(n).**
c_{es}	**cess**	Array containing the elemental composition coefficients of aqueous species. Dimensioning: **cess(nctpar,nstpar).** Usage: **cess(nc,ns)** is the coefficient of the **nc**-th chemical element for the **ns**-th aqueous species.
	cessd	Holding array used to read in the elemental composition coefficients of species listed on the data file. Dimensioning: **cessd(nsqpa1).** Usage: **cessd(n)** is the ***n-th*** coef-

		ficient for the current species as it is written on the data file; it corresponds to the element whose name is **uelemd(n)**.
	cessi	Temporary storage array used to decode the elemental composition coefficients of species listed on the data file. Dimensioning: **cessi(nsqpa1)**. Usage: **cessi(nc)** is the coefficient for the **nc**-th element. This array is used to transfer data between the **cessd** array and the **cess** array.
m_i	**conc**	Molal concentration of an aqueous species. Dimensioning: **conc(nstpar)**. Usage: **conc(ns)** is the molal concentration of the **ns**-th aqueous species. This is defined as zero for water, even its molality is technically fixed at Ω (≈ 55.51).
$log\ m_i$	**conclg**	Log molal concentration of an aqueous species. Dimensioning: **conclg(nstpar)**. Usage: **conclg(ns)** is the log molal concentration of the **ns**-th aqueous species.
m_i	**csort**	Molal concentrations of the aqueous species, sorted in order of increasing value. Dimensioning: **csort(nstpar)**. Usage: primarily as a storage array required for the sorting; **csort(n) = conc(jsort(n))**. It is unique to EQ3NR.
	csp	Constraining parameter array; the meaning of each member of this array is determined by the corresponding member of the **jflag** array. Dimensioning: **csp(nsqpa1)**. Usage: **csp(ns)** is the constraining parameter for the **ns**-th aqueous basis species. This array is data file ordered. It is unique to EQ3NR.
	cspb	Holding array into which the **csp** values are read from the **input** file. Dimensioning: **cspb(nsqpa1)**. Usage: **cspb(n)** is the constraining parameter for the **n**-th aqueous basis species listed on the **input** file. This array is input file ordered. It is unique to EQ3NR.
	cstor	Stoichiometric mass balance factors of aqueous species for oxygen in the O^{2-} valence form and hydrogen in the H^+ valence form. Dimensioning: **cstor(2,nstpar)**. Usage: **cstor(1,ns)** is the factor for the **ns**-th aqueous species in the mass balance for O^{2-}; **cstor(2,ns)** is the corresponding factor in the mass balance for H^+. This array is used to set up the corresponding parts of the **csts** array. It is unique to EQ3NR.
$u_{s's}$	**csts**	Mass balance coefficient array. Dimensioning: **csts(nsqpa1,nstpar)**. Usage: **csts(nse,ns)** is the coefficient of the **ns**-th aqueous species in the mass balance equation for the **nse**-th basis species. It is unique to EQ3NR.
	cte	Equivalent total molality of a chemical element in a given valence form or combination of valence forms in mutual equilibrium. Dimensioning: **cte(nsqpa1)**. Usage: **cte(ns)** is the equivalent total molality of the **ns**-th aqueous basis species.
$\underline{\delta}$	**del**	Newton-Raphson correction term vector. Dimensioning: **del(kpar)**. Usage: **del(kcol)** is the correction term for the **kcol**-th master iteration variable.
δ_{func}	**delfnc**	Convergence function that measures the improvement in **delmax**, defined as (1.0 - (**delmax/delmxo**)).
δ_{max}	**delmax**	Max norm of $\underline{\delta}$.
	delmxo	The previous value of **delmax**.

$\underline{W}$	**dlogxw**	Array of partial derivatives of the log mole fraction of water with respect to the log molalities of the solute basis species. Dimensioning: **dlogxw(nsqpar)**. Usage: **dlogxw(ns)** is the partial derivative with respect to the **ns**-th basis species.
$\frac{d\mu_{ijk}}{dT}$	**dmu1**	Array of first derivatives with respect to temperature of third order primitive virial coefficients of Pitzer's equations (25°C values). Dimensioning: **dmu1(nmupar)**. Usage: **dmu1(k)** is the derivative for the **k**-th species triplet. See **bmu**, **dmu2**, **nmux**, and **pmu**.
$\frac{d^2\mu_{ijk}}{dT^2}$	**dmu2**	Array of second derivatives with respect to temperature of third order primitive virial coefficients of Pitzer's equations (25°C values). Dimensioning: **dmu2(nmupar)**. Usage: **dmu1(k)** is the derivative for the **k**-th species triplet. See **bmu**, **dmu1**, **nmux**, and **pmu**.
$\frac{d^S\lambda_{ij}^{(n)}}{dT}$	**dslm1**	Array of first derivatives with respect to temperature of second order "short range" primitive virial coefficient parameters of Pitzer's equations (25°C values). Dimensioning: ***dslm1(3,nslpar)***. *Usage:* ***dslm1(n,k)*** *is the* ***n****-th such parameter for the* ***k****-th species pair. See* **bslm**, **dslm2**, **nslmx**, **pslm**, and **pslam**.
$\frac{d^2\,S\lambda_{ij}^{(n)}}{dT^2}$	**dslm2**	Array of second derivatives with respect to temperature of second order "short range" primitive virial coefficient parameters of Pitzer's equations (25°C values). Dimensioning: **dslm2(3,nslpar)**. Usage: **dslm2(n,k)** is the **n**-th such parameter for the **k**-th species pair. See **bslm**, **dslm1**, **nslmx**, **pslm**, and **pslam**.
	ee	A work array used in solving matrix equations. Dimensioning: **ee(kpar)**.
	efac	Array of reciprocal stoichiometric weighting factors used in pre-Newton-Raphson optimization; this is the $\frac{1}{n}$ which appears in the array of $(\beta + 1)^{\frac{1}{n}}$ factors used in this optimization method. Dimensioning: **efac(nsqpa1)**. Usage: **efac(ns)** is the factor for the **ns**-th basis species.
Eh	**eh**	Redox potential, volts.
${}^E\lambda(I)$, ${}^E\lambda'(I)$	**elam**	The higher order electrostatic lambda functions that appear in Pitzer's equations. Dimensioning: **elam(3,10,10)**. Usage: **elam(1,i,j)** = ${}^E\lambda(I)$ for a pair of ions the magnitude of whose charges are **i** and **j**; **elam(2,i,j)** = ${}^E\lambda'(I)$ for the same pair of ions; **elam(3,i,j)** = ${}^E\lambda''(I)$ and is presently not used.
	eps100	One hundred times the **real*8** machine epsilon.
F	**farad**	The Faraday constant, 23062.3 cal/equivalent-volt.
f_{O_2}	**fo2**	Oxygen fugacity, bars.

$log f_{O_2}$	**fo2lg**	Log oxygen fugacity.
f_g	**fug**	Array of fugacities of gas species. Dimensioning: **fug(ngtpar)**. Usage: **fug(ng)** is the fugacity of the **ng**-th gas species.
$log f_g$	**fuglg**	Array of log fugacities of gas species. Dimensioning: **fuglg(ngtpar)**. Usage: **fuglg(ng)** is the log fugacity of the **ng**-th gas species.
$g(\alpha_n\sqrt{I})$	**g**	The function $g(x)$ which appears in Pitzer's equations. Dimensioning: **g(2)**. Usage: **g(1)** = $g(\alpha_1\sqrt{I})$ and **g(2)** = $g(\alpha_2\sqrt{I})$.
$g'(\alpha_n\sqrt{I})$	**gp**	The function $g'(x)$ which appears in Pitzer's equations. Dimensioning: **gp(2)**. Usage: **gp(1)** = $g'(\alpha_1\sqrt{I})$ and **gp(2)** = $g'(\alpha_2\sqrt{I})$.
$log\ \lambda_n, log\gamma_i$	**glg**	Array of log activity coefficients of aqueous species; this is a mole fraction activity coefficient for water and a molal activity coefficient for all solute species. Dimensioning: **glg(nstpar)**. Usage: **glg(ns)** is the log activity coefficient of the **ns**-th aqueous species.
	glgo	Array of previous values of log activity coefficients of aqueous species Dimensioning. **glgo(nstpar)**.
	gm	A work array that is used to store a copy of **aa**, the Jacobian matrix. Dimensioning: **gm(kpar,kpar)**.
	iapxmx	Maximum number of coefficients for determining activity coefficient parameters per solid solution in the **apx** array. This is the variable equivalent to the parameter ***iapxpa***.
	iapxpa	Dimensioning parameter: the maximum number of coefficients for determining activity coefficient parameters per solid solution in the **apx** array. See **iapxmx**.
	ibasis	Array that contains a record of basis switching. Dimensioning: **ibasis(nstpar)**. Usage: if **ibasis(ns)** is not 0, then the **ns**-th species was switched with the species whose index is given by **ibasis(ns)**.
	ibetmx	The index of the element of the **beta** array corresponding to the max norm (**betamx**).
	ibswx	Array that contains the indices of species which are candidates for switching into the basis set. This is a part of the automatic basis switching mechanism. Dimensioning: **ibswx(nsqpar)**. Usage: if **ibswx(ns)** is not 0, then it gives the index of a species to switch with the **ns**-th basis species.
s_z	**iebal**	The index of the aqueous species chosen for electrical balancing.
	ielam	Flag controlling whether or not higher order electrostatic terms are used in Pitzer's equations. It is set by the **uelam** flag on the supporting data file and has the following meanings: = -1 Higher order electrostatic terms are not used = 1 They are used

ier An error flag parameter commonly found in subroutine calling sequences. Values greater than zero mark error conditions. Values less than zero mark warning conditions. The significance of this flag is only that the subroutine did or may have failed to carry out its primary function. For example, a matrix solver may have found a matrix to be computationally singular. This situation may or may not equate to an error in the context of the larger code. For example, the larger code may not have a problem with the fact that a matrix is computationally singular.

iindex Array containing the indices of the basis species for which the starting iteration values must be estimated simultaneously. Dimensioning: **iindex(nsqpar)**. Usage: **iindex(n)** is the **n**-th such species.

iindx1 Array that contains the indices of the species for which the master iteration variables are defined. Dimensioning: **iindx1(kpar)**. Usage: **iindx1(kcol)** is the index of the species corresponding to the **kcol**-th master iteration variable.

iktmax The maximum number of end members in a solid solution. This is the variable which corresponds to the dimensioning parameter **iktpar**.

iktpar Dimensioning parameter: the maximum number of end members in a solid solution. See **iktmax**.

insgfl Flag array for determining how to compute the activity coefficient of a neutral solute species in conjunction with the B-dot equation. Dimensioning: **insgfl(nstpar)**. Usage: **insgfl(ns)** is the flag for the **ns**-th aqueous species:

= 0 Set $log\,\gamma_i = 0$ (polar species)

= -1 Use the Drummond (1981) polynomial (non-polar species).

iodb1, etc. Debugging print option switches; see Chapter 6. Some of these switches differ from those used in EQ6.

iopg1, etc. Activity coefficient option switches; see Chapter 6.

iopr1, etc., Print option switches; see Chapter 6. Some of these switches differ from those used in EQ6.

iopt1, etc. Model option switches; see Chapter 6. Some of these switches differ from those used in EQ6.

ir A work array used in solving matrix equations. Dimensioning: **ir(kpar)**.

irang The **real*8** machine exponent range parameter (the exponent range is $\pm$**irang**).

istack A work array used to sort aqueous species in increasing order of concentration. Dimensioning: **istack(nstpar)**.

itermx Limit on the number of Newton-Raphson iterations.

izm The max norm on the electrical charges of the aqueous species.

jcflag Status flag array for chemical elements. Dimensioning: **jcflag(nctpar)**. Usage: **jcflag(nc)** is the flag for the **nc**-th element:

= 0 An element does not appear in the current model

= 1 It does appear

jflag Switch array that defines the type of constraint (governing equation) applied to each basis species. Dimensioning: **jflag(nsqpar)**. Usage: **jflag(ns)** is the flag for the **ns**-th basis species. See Chapter 6.

jflagb Array used to read in **jflag** values from the **input** file. Dimensioning: **jflagb(nsqpar)**. Usage: **jflagb(n)** is the **jflag** value for the **n**-th basis species read from the **input** file. This array is thus input file ordered. It is unique to EQ3NR.

jgflag Status switch array for gas species. Dimensioning: **jgflag(ngtpar)**. Usage: **jgflag(ng)** is the flag for the **ng**-th gas:
- = 0 The gas species appears in the current model
- = 2 It is thermodynamically suppressed

jkflag Status switch array for solid solution end-member components. Dimensioning: **jkflag(iktpar,nxtpar)**. Usage: **jkflag(ik,nx)** is the flag for the **ik**-th end-member component of the **nx**-th solid solution:
- = 0 The component appears in the current model
- = 2 It is thermodynamically suppressed
- = 4 It could not be found among the pure minerals

jmflag Status switch array for pure minerals. Dimensioning: **jmflag(nmtpar)**. Usage: **jmflag(nm)** is the flag for the **nm**-th pure mineral:
- = 0 The mineral appears in the current model
- = 2 It is thermodynamically suppressed

jrs The indices of the aqueous reactions, sorted in increasing order of concentration of the associated species. Dimensioning: **jrs(nrstpa)**. Usage: **jrs(n)** is the index of the reaction which corresponds to the non-basis aqueous species whose concentration is **n**-th in this order. This array is used in making sorted summations.

jsflag Status switch array for aqueous species. Dimensioning: **jsflag(nstpar)**. Usage: **jsflag(ns)** is the flag for the **ns**-th aqueous species:
- = 0 The species appears in the current model
- = 2 It is thermodynamically suppressed
- = 3 It does not appear in the current model

jsol Array whose values define chosen models of activity coefficients in solid solutions. Dimensioning: **jsol(nxtpar)**. Usage: **jsol(nx)** is the flag for the **nx**-th solid solution:
- = 0 Ideal solution

(The data files currently support no other options.)

jsort The indices of the aqueous species, sorted in increasing order of concentration. Dimensioning: **jsort(nstpar)**. Usage: **jsort(n)** is the index of the aqueous species whose concentration is **n**-th in this order. This array is used in making sorted summations.

jstack A work array used in sorting aqueous species in increasing order of concentration. Dimensioning: **jstack(nstpar)**.

jxflag Status switch array for solid solutions. Dimensioning: **jxflag(nxtpar)**. Usage: **jxflag(nx)** is the flag for the **nx**-th solid solution:
- = 0 The phase appears in the current model
- = 4 It has no active end-member components

jxmod Flag specifying the type of a species/reaction affected by an alter/suppress option specified on the **input** file. Dimensioning: **jxmod(nxmdpa)**. Usage: **jxmod(n)** is the flag for the **n**-th such option specified (see **kxmod**, **uxmod**, **xlkmod**, and **nxmod**):
= 0 Aqueous species/reaction
= 1 Pure mineral
= 2 Gas
= 3 Solid solution

kct Number of chemical elements present in the aqueous system currently being modeled.

kction A pointer array used in pre-Newton-Raphson optimization. It points to entries in the subset of master iteration variables which must be solved simultaneously which correspond to basis species used to complete the definition of input constraints for other basis species. Dimensioning: **kction(nsqpar)**. Usage: **kction(n)** is the index in this subset for the basis species which completes the constraint definition for the basis species whose index in this subset is **n**. It is unique to EQ3NR.

kdim Dimension of the Jacobian matrix; **kdim** = **ksq** in EQ3NR.

kebal Variable denoting the position of the species selected for electrical balancing in the set of master iteration variables. It is unique to EQ3NR.

khydr Variable denoting the position of H^+ in the set of master iteration variables.

kkndex Array which marks those basis species whose concentrations must be initially estimated simultaneously. Dimensioning:
kkndex(nsqpar). Usage: **kkndex(ns)** is the flag for the **ns**-th such species:
= 0 Do not estimate simultaneously
= 1 Estimate simultaneously
It is unique to EQ3NR.

kmax The maximum number of master variables readable by EQ6. This is the variable corresponding to the parameter **kpar**.

kpar Dimensioning parameter: the maximum number of master variables readable by EQ6. See **kmax**.

ksb Variable denoting the position of the species used as the redox variable, currently $O_{2(g)}$ only; **ksb** = **kct** + 1.

ksq The number of active basis species.

kxmod Flag defining the type of alter/suppress option specified on the **input** file. Dimensioning: **kxmod(nxmdpa)**. Usage: **kxmod(n)** is the flag for the n-th such option specified (see **jxmod**, **uxmod**, **xlkmod**, and **nxmod**):
= -1 The corresponding species/reaction is suppressed
= 0 Its *log K* is replaced by **xlkmod**
= 1 Its *log K* is augmented by **xlkmod**
= 2 Same as = 1, but **xlkmod** is given in units of kcal/mole

$log\ \lambda_{\sigma\psi}$ **lamlg** Array of log activity coefficients of solid solution end-member components. Dimensioning: **lamlg(iktpar,nxtpar)**. Usage: **lamglg(ik,nx)** is the activity coefficient for the ik-th end-member component of the nx-th solid solution.

$m_{T,\varepsilon}$	**mte**	Array of total molalities of dissolved chemical elements. Dimensioning: **mte(nctpar)**. Usage: **mte(nc)** is the molality of the **nc**-th element.
M_i	**mwtss**	Array of molecular weights of aqueous species. Dimensioning: **mwtss(nstpar)**. Usage: **mwtss(ns)** is the molecular weight of the **ns**-th aqueous species.
	nadi	The unit number of the **data1** file.
	nalpha	A pointer array used to find the α_1 and α_2 parameters of Pitzer's equations for a species pair for which ${}^S\lambda_{ij}$ and ${}^S\lambda'_{ij}$ values are to be calculated. Dimensioning: **nalpha(nslpar)**. Usage: **palpha(1,nalpha(n))** = α_1 and **palpha(2,nalpha(n))** = α_2 for the **n**-th such solute species pair.
	nappar	Dimensioning parameter: the maximum number of distinct pairs of α_1, α_2 parameters of Pitzer's equations. See **napt**.
	napt	The maximum number of distinct pairs of α_1, α_2 parameters of Pitzer's equations. This is the variable corresponding to the dimensioning parameter **nappar**.
	narxmx	The maximum number of coefficients per temperature range for an interpolating polynomial coefficient array such as **ars** and **amn**. This is the variable corresponding to the parameter **narxpa**.
	narxpa	Dimensioning parameter: the maximum number of coefficients per temperature range for an interpolating polynomial coefficient array. See **narxmx**.
	nchlor	The index of the species Cl^-.
$\sigma_{T,\Psi}$	**ncomp**	Array giving the number of end-member components in a solid solution. Dimensioning: **ncomp(nxtpar)**. Usage: **ncomp(nx)** is the number of such components for the **nx**-th solid solution.
	ncompb	Array giving the number of solid solution end-member components of solid solutions for which mole fractions are read from the **input** file. Dimensioning: **ncompb(nxtpar)**. Usage: **ncompb(nxb)** is the number of such components for the **nxb**-th solid solution so listed on the **input** file.
ε_T	**nct**	Total number of chemical elements.
	nctmax	The maximum number of chemical elements. This is a variable which corresponds to the parameter **nctpar**.
	nctpar	Dimensioning parameter: the maximum number of chemical elements. See **nctmax**.
	nend	Array that stores the indices of pure minerals that correspond to end-member components of solid solutions. Dimensioning: **nend(iktpar,nxtpar)**. Usage: **nend(ik,nx)** is the pure mineral index for the **ik**-th component of the **nx**-th solid solution.
	nfac	Array of indices of dominant species. Dimensioning: **nfac(nsqpar)**. Usage: **nfac(ns)** is the index of the dominant species in the mass balance for the **ns**-th basis species.

g_T	**ngt**	Total number of gas species.
	ngtmax	The maximum number of gas species. This is a variable which corresponds to the parameter **ngtpar**.
	ngtpar	Dimensioning parameter: the maximum number of gas species. See **ngtmax**.
	nhydr	The index of the species H^+.
	ninpt	The unit number of the **input** file.
	ninpts	The unit number of the file which is a copy of the **input** file that has been stripped of comment lines.
	nmlx	The number of entries in the **nmxx** pointer array, which is used in evaluating terms in the μ_{ijk} coefficients in Pitzer's equations.
ϕ_T	**nmt**	Total number of pure minerals.
	nmtmax	Maximum number of pure minerals.
	nmtpar	Dimensioning parameter: the maximum number of pure minerals. See **nmtmax**.
	nmupar	Dimensioning parameter: the maximum number of species triplets with μ_{ijk} coefficients specified for use in evaluating Pitzer's equations. See **nmut**.
	nmu	The number of species triplets with μ_{ijk} coefficients specified for use in evaluating Pitzer's equations.
	nmut	The maximum number of species triplets with μ_{ijk} coefficients specified for use in evaluating Pitzer's equations. This is the variable corresponding to the dimensioning parameter **nmupar**.
	nmux	Array identifying the species in triplets with μ_{ijk} coefficients specified for use in evaluating Pitzer's equations. Dimensioning: **nmux(3,nmupar)**. Usage: **nmux(i,k)** is the index of the **i**-th species in the **k**-th triplet.
	nmxi	A pointer array which gives the range in the **nmux** array. Dimensioning: **nmxi(2,nstpar)**. Usage: **nmxi(1,ns)** to **nmxi(2,ns)** is the range in the **nmxx** array corresponding to the species whose index is **ns**.
	nmxpar	Dimensioning parameter: the maximum number of entries in the **nmxx** pointer array, which is used in evaluating terms in the μ_{ijk} coefficients in Pitzer's equations. To be absolutely safe, one could set **nmxpar** equal to 3***nmupar**. See **nmxt**.
	nmxx	An ordered pointer array constructed from the **nmux** array which, when used in connection with the **nmxi** pointer array, gives for each solute species the indices of the other two species appearing in a triplet for a μ_{ijk} coefficient in Pitzer's equations and *also the index of the triplet itself. Dimensioning: **nmxx(3,nmxpar)**. Usage:* **nmxx(1,kk)** is the index of the first species in the **kk**-th entry, **nmxx(2,kk)** is the index of the second species in this entry, and **nmxx(3,kk)** is the index of triplet itself. For a given species whose index is **ns**, the corresponding range in the **nmxx** array is defined by **kk** in the range from **nmxi(1,ns)** to **nmxi(2,ns)**. This combination of

pointer arrays is useful in evaluating the sum, $3\sum_j \sum_k \mu_{ijk} m_j m_k$, which appears in the expression for the solute activity coefficient.

nmxt The maximum number of entries in the **nmxx** pointer array, which is used in evaluating terms in the μ_{ijk} coefficients in Pitzer's equations. This is the variable corresponding to the *dimensioning parameter* ***nmxpar***.

noutpt The unit number of the **output** file.

npkup The unit number of the **pickup** file.

r_T **nrst** Total number of reactions among aqueous species.

nrstmx The maximum number of aqueous reactions. This is a variable corresponding to the parameter **nrstpa**.

nrstpa Dimensioning parameter: the maximum number of aqueous reactions. See **nrstmx**.

s_B **nsb** The number of strict basis species. Also the index denoting the aqueous redox species (currently O_2); **nsb** = **nct** + 1.

nsb1 Index denoting the first species in the auxiliary basis set; **nsb1** = **nsb** + 1.

nslpar Dimensioning parameter: the maximum number of species pairs having $^{S}\lambda_{ij}^{(n)}$ parameters specified for evaluation of Pitzer's equations. See **nslmt**.

nslm The number of species pairs with $^{S}\lambda_{ij}^{(n)}$ parameters specified for use in evaluating Pitzer's equations.

nslmt The maximum number of species pairs with $^{S}\lambda_{ij}^{(n)}$ parameters specified for evaluation of Pitzer's equations. This is the variable corresponding to the dimensioning parameter **nslpar**.

nslmx Array identifying the species in pairs with $^{S}\lambda_{ij}^{(n)}$ parameters specified for use in evaluating Pitzer's equations. Dimensioning: **nslmx(2,nslpar)**. Usage: **nslmx(i,k)** is the index of the **i**-th species in the **k**-th pair.

nslx The number of entries in the **nsxx** pointer array, which is used in evaluating terms in the $^{S}\lambda_{ij}^{(n)}$ parameters in Pitzer's equations.

nsp Array containing the indices of the aqueous, mineral, solid solution end-member component, or gas species that are required to define certain kinds of constraints on basis species on the **input** file. Dimensioning: **nsp(nsqpar)**. Usage: **nsp(ns)** is the index of the species for the constraint place on the **ns**-th basis species. This never refers to the **ns**-th basis species itself, but always a second species. Phase equilibrium (solubility) constraints and log activity combination and mean log activity constraints all require the specification of such second species. This array is data file ordered. It is unique to EQ3NR.

	nspec	Array that contains the indices of the basis species which appear on the **input** file. Dimensioning: **nspec(nsqpar)**. Usage: **nspec(n)** is the index of the **n**-th basis species appearing on the **input** file. It is thus input file ordered. It is unique to EQ3NR.
s_Q	**nsq**	Number of aqueous basis species.
	nsq1	The variable equivalent to **nsq** + 1. This is used in reaction coefficient arrays to mark the coefficient of the species formally associated with a given reaction. For example, **cdrs(nsq1,nrs)** refers to the coefficient of the associated aqueous species destroyed in the **nrs**-th reaction.
	nsqb	Number of basis species which appear on the **input** file.
	nsqmax	The maximum number of aqueous basis species. This is the variable corresponding to the parameter **nsqpar**.
	nsqmx1	A variable equal to **nsqmax** + 1. This corresponds to the parameter **nsqpa1**.
	nsqpa1	Dimensioning parameter: the equivalent of **nsqpar** + 1. See **nsqmx1**.
	nsqpar	Dimensioning parameter: the maximum number of aqueous basis species. See **nsqmax**.
s_T	**nst**	Total number of aqueous species; this currently includes all basis species but only those non-basis species appearing in the system for a given problem defined on the **input** file.
	nstmax	The maximum number of aqueous species. This is a variable corresponding to the parameter **nstpar**.
	nstpar	Dimensioning parameter: the maximum number of aqueous species. See **nstmax**.
	nsxi	A pointer array which gives the range in the **nsxx** array. Dimensioning: **nsxi(2,nstpar)**. Usage: **nsxi(1,ns)** to **nsxi(2,ns)** is the range in the **nsxx** array corresponding to the species whose index is **ns**.
	nsxpar	Dimensioning parameter: the maximum number of entries in the **nsxi** pointer array, which is used in evaluating terms in the $^{s}\lambda_{ij}$ coefficients in Pitzer's equations. To be absolutely safe, one could set **nsxpar** equal to 2***nmupar**. See **nsxt**.
	nsxt	The maximum number of entries in the **nsxi** pointer array, which is used in evaluating terms in the $^{s}\lambda_{ij}$ coefficients in Pitzer's equations. This is the variable corresponding to the dimensioning parameter **nsxpar**.
	nsxx	An ordered pointer array constructed from the **nslmx** array which, when used in connection with the **nsxi** pointer array, gives for each solute species the index of the other species appearing in a pair for the $^{s}\lambda_{ij}$ coefficients in Pitzer's equations and also the index of the pair itself. Dimensioning: **nsxx(2,nsxpar)**. Usage: **nsxx(1,kk)** is the index of the other species in the **kk**-th entry and **nsxx(2,kk)** is the pair itself. For a given species whose index is **ns**, the corresponding range in the **nsxx** array is defined by **kk** in the range from **nsxi(1,ns)** to **nsxi(2,ns)**. This combination of point-

		er arrays is useful in evaluating the sum, $2\sum_j \lambda_{ij} m_j$, which appears in the expression for the solute activity coefficient.
	ntitl	The number of lines of the title on the **input** file.
	ntitld	The number of lines of the title on the **data1** file.
	ntitmx	The maximum number of lines of the title on the **input** or **data1** file. This is a variable corresponding to the parameter **ntitpa**.
	ntitpa	Dimensioning parameter: the maximum number of lines of the title on the **input** or **data1** file. See **ntitmx**.
	ntpr	The index of the temperature range corresponding to the temperature specified on the **input** file.
	ntprmx	The maximum number of temperature ranges for an interpolating polynomial coefficient array such as **ars** and **amn**. This is a variable corresponding to the parameter **ntprpa**.
	ntprpa	Dimensioning parameter: the maximum number of temperature ranges for an interpolating polynomial coefficient array such as **ars** and **amn**. See **ntprmx**.
	nttyo	The unit number of the screen file.
	nxmdmx	The maximum number of alter/suppress options that can be specified on the **input** file. This is a variable corresponding to the parameter **nxmdpa**.
	nxmdpa	Dimensioning parameter: the maximum number of alter/suppress options that can be specified on the **input** file. See **nxmdmx**.
	nxmod	The number of species/reactions affected by alter/suppress options specified on the **input** file. See **jxmod**, **kxmod**, **uxmod**, and **xlkmod**.
Ψ_T	**nxt**	Total number of solid solution phases.
	nxtb	Number of solid solutions whose compositions are read from the **input** file. It is unique to EQ3NR.
	nxtmax	The maximum number of solid solutions. This is a variable corresponding to the parameter **nxtpar**.
	nxtpar	Dimensioning parameter: the maximum number of solid solutions. See **nxtmax**.
Ω	**om**	The water constant, 1000 ÷ the molecular weight of water($\Omega \approx 55.51$).
$log\ \Omega$	**omlg**	Log water constant.
ϕ	oscoff	The osmotic coefficient.
α_1, α_2	**palpha**	The α_1 and α_2 parameters of Pitzer's equations. Dimensioning: **palpha(2,nappar)**. Usage: **palpha(1,k)** = α_1 and **palpha(2,k)** = α_2 for the k-th distinct pair of α_1, α_2

		values; **k = nalpha(n)** for the **n**-th solute species pair for which ${}^{S}\lambda_{ij}$ and ${}^{S}\lambda'_{ij}$ coefficients are to be computed.
pe	**pe**	The electron activity function; $pe = -\log a_{e^-}$.
μ_{ijk}	**pmu**	Array of third order primitive virial coefficients of Pitzer's equations (values at the current specified temperature). Dimensioning: **pmu(nmupar)**. Usage: **pmu(k)** is the coefficient for the **k**-th species triplet. See **bmu, dmu1, dmu2,** and **nmux**.
P	**press**	Pressure, bars.
${}^{S}\lambda_{ij}$	**pslam**	Array of second order "short range" primitive virial coefficients of Pitzer's equations (values at the current specified temperature). Dimensioning: **pslam(3,nslpar)**. Usage: **pslam(n,k)** is the **n**-th such coefficient for the **k**-th species pair. See **bslm, dslm1, dslm2, nslmx,** and **pslm**.
${}^{S}\lambda_{ij}^{(n)}$	**pslm**	Array of second order "short range" primitive virial coefficient parameters of Pitzer's equations (values at the current specified temperature). Dimensioning: **pslm(3,nslpar)**. Usage: **pslm(n,k)** is the **n**-th such parameter for the **k**-th species pair. See **bslm, dslm1, dslm2, nslmx,** and **pslam**.
	qbassw	Logical flag indicating whether or not basis switching has taken place.
	qbswx	Logical flag indicating whether or not there are candidates for automatic basis switching.
R	**rconst**	The gas constant, 1.98726 cal/mol-°K.
	res	A work array used in solving matrix equations. Dimensioning: **res(kpar)**.
$\rho_{g/ml}$	**rho**	Density of aqueous solution, g/ml. It is unique to EQ3NR.
$-\underline{\alpha}$	**rhs**	A negative copy of the array **alpha**. Dimensioning: **rhs(kpar)**.
$J(x)$	**rj**	The $J(x)$ function used in Pitzer's equations.
$J'(x)$	**rjp**	The $J'(x)$ function used in Pitzer's equations.
	screwd	Under-relaxation parameter that bounds **delmax**.
log Q/K	**si**	Saturation index of a solid solution end-member component. Dimensioning: **si(iktpar,nxtpar)**. Usage: **si(ik,nx)** is the saturation index for the **ik**-th component of the **nx**-th solid solution.
	smp100	One hundred times the smallest positive **real*8** number.
log Q/K	**siss**	Array of saturation indices of solid solutions. Dimensioning: **siss(nxtpar)**. Usage: siss(**nx**) is the saturation index of the **nx**-th solid solution.
H_{sr}, H_{zr}	**stor1**	The H_{sr} or H_{zr} factors defined in Chapter 9. Dimensioning: **stor1(nrstpa)**. Usage: **stor1(nrs)** is the factor for the **nrs**-th reaction. Note that these are defined in terms of molalities in EQ3NR, whereas they are defined in terms of moles in EQ6

	tdamax	The nominal upper temperature limit of the data file, °C.
	tdamin	The nominal lower temperature limit of the data file, °C.
$C_{TS,\,mg/kg}$	**tdspkg**	Total dissolved salts, mg/kg. This is unique to EQ3NR.
$C_{TS,\,mg/L}$	**tdspl**	Total dissolved salts, mg/L. This is unique to EQ3NR.
	tempc	Temperature, °C.
T	**tempk**	Temperature, °K.
	tolbt	Convergence bound on **betamx**.
	toldl	Convergence bound on **delmax**.
	tolsat	Limit on the absolute value of the affinity of a mineral of fixed composition for the mineral to be listed as "saturated" on the **output** file. This is unique to EQ3NR.
	tolxat	Limit on the absolute value of the affinity of a solid solution for the phase to be listed as "saturated" on the **output** file. This is unique to EQ3NR.
	uactop	Character string describing the model selected by the user for computing the activity coefficients of aqueous species; this is set by the **iopg1** value read from the **input** file.
	ubasis	Array of names of aqueous species that are to be switched into the basis set. Dimensioning: **ubasis(nsqpar)**. Usage: **ubasis(ns)** is the name of the species to be switched into the basis set in place of the **ns**-th basis species.
	ubbig	The name of the basis species corresponding to the mass balance residual with the most extreme positive value; if none is positive, **ubbig** is assigned a value of 'none'.
	ubgamx	The name of the species corresponding to the max norm (**bgamx**) of activity coefficient residual functions.
	ubneg	The name of the basis species corresponding to the mass balance residual with the most extreme negative value; if none is negative, **ubneg** is assigned a value of 'none'.
	ucode	A variable containing the name of the code.
	udrxd	Holding array used to read in the names of species appearing in reactions listed on the data file. Dimensioning: **udrxd(nsqpa1)**. Usage: **udrxd(n)** is the name of the species corresponding to the **n**-th coefficient in the current reaction as it is written on the data file. See **cdrsd**.
	uelam	Flag read from the supporting data file indicating whether or not higher order electrostatic terms are to be used in Pitzer's equations: = 'off' Higher order electrostatic terms are not used = 'on' They are used
	uelem	Array of names of chemical elements (their chemical symbols). Dimensioning: **uelem(netpar)**. Usage: **uelem(ne)** is the name of the ne-th element.

uelemd Holding array of names of chemical elements. Dimensioning: **uelemd(nctpar)**. Usage: **uelemd(n)** is the name of the **n**-th element listed for the current species. See **cessd**.

uendit The string 'endit.'.

ueqlrn A string containing the release number of the supporting EQLIB library.

ueqlst A string containing the stage number of the supporting EQLIB library.

ugas Array of names of gas species. Dimensioning: **ugas(ngtpar)**. Usage: **ugas(ng)** is the name of the **ng**-th gas species.

ujtype Array of character strings that describe the **jflag** options. Dimensioning: **ujtype(nsqpar)**.

umemb Array of names of solid solution end-member components appearing on the **input** file. Dimensioning: **umemb(iktpar,nxtpar)**. Usage: **umemb(ikb,nxb)** is the name of the **ikb**-th component in the **nxb**-th solid solution whose composition is given on the **input** file. It is thus input file ordered. It is unique to EQ3NR.

umin Array of name of pure minerals. Dimensioning: **umin(nmtpar)**. Usage: **umin(ng)** is the name of the **nm**-th pure mineral.

undms Array of names of data file basis species appearing on the EQ3NR **pickup** file and the EQ6 **input** file. Dimensioning: **undms(nsqpar)**. Usage: **undms(n)** is the name of the **n**-th such species appearing on one of these files.

unone The string 'none'.

unrms Array of names of run basis species appearing on the EQ3NR **pickup** file and the EQ6 **input** file. Dimensioning: **unrms(nsqpar)**. Usage: **unrms(n)** is the name of the **n**-th such species appearing on one of these files. If **unrms(n)** refers to a different species than **undms(n)**, then the former is to be switched into the basis set in place of the latter.

uphas1 Array of names of aqueous basis species or mineral, solid solution, or gas species required to complete the definition of an input constraint under the **jflag** = 17, 18, 19, 20, and 21 options. Dimensioning: **uphas1(nsqpar)**. Usage: **uphas1(n)** is the name of such a species required to complete the constraint for the **n**-th basis species appearing on the **input** file. This array is input file ordered. It is unique to EQ3NR.

uphas2 Array of names of solid solution end-member components required to complete the definition of an input constraint under the **jflag** = 20 option. Dimensioning: **uphas2(nsqpar)**. Usage: **uphas2(n)** is the name of such a species required to complete the constraint for the **n**-th basis species appearing on the **input** file. This array is input file ordered. It is unique to EQ3NR.

uredox The name of an auxiliary basis species which is part of an aqueous redox couple used to calculate the default redox state of aqueous redox couples. It is unique to EQ3NR.

urelno A string containing the release number of the code it is contained in.

	usolb	Array of names of solid solution phases whose composition are specified on the **input** file. Dimensioning: **usolb(nxtpar)**. Usage: **usolb(nxb)** is the name of the **nxb**-th such phase. This array is input file ordered. It is unique to EQ3NR.
	usolx	Array of names of solid solutions. Dimensioning: **usolx(nxtpar)**. Usage: **usolx(nx)** is the name of the **nx**-th solid solution.
	uspec	Array of names of aqueous species. Dimensioning: **uspec(nstpar)**. Usage: **uspec(ns)** is the name of the **ns**-th aqueous species.
	uspecb	Array of names of aqueous basis species listed on the **input** file. Dimensioning: ***uspecb(nsb)***. Usage: ***uspecb(nsb)*** is the name of the **nsb**-th such species.This array is input file ordered. It is unique to EQ3NR.
	ussnp	Array of names of solid solution end members listed on the **data1** file which fail to correspond to any pure mineral on the same file. Dimensioning: **ussnp(iktpar, nxtpar)**. Usage: **ussnp(n,nx)** is the name of the ***n-th*** such component in the **nx**-th solid solution.
	ustage	A string containing the stage number of the code it is contained in.
	utitl	The title (text) from the **input** file. Dimensioning: **utitl(ntitpa)**. Usage: **utitl(n)** is the n-th line of this title.
	utitld	The title (text) from the **data1** file. Dimensioning: ***utitld(ntitpa)***. Usage: ***utitld(n)*** is the n-th line of this title.
	uxmod	The name of a species for which an alter/suppress option is specified on the **input** file. Dimensioning: **uxmod(nxmdpa)**. Usage: **uxmod(n)** is the name of the species for the n-th such option specified (see **jxmod**, **kxmod**, **xlkmod**, and **nxmod**).
	uxtype	Array of character strings that describe the solid solution activity coefficient models defined by the **jsol** flag. Dimensioning: **uxtype(10)**. Usage: **uxtype (n)** is the string which describes the model identified by **jsol(n)**.
	uzvec1	Array containing the names of the species corresponding to master iteration variables. Dimensioning: **uzvec1(kpar)**. Usage: **uzvec1 (kcol)** is the species name corresponding to the **kcol**-th master iteration variable.
	vmin0	Array containing the molar volumes of the pure minerals. Dimensioning: **vmin0(nmtpar)**. Usage: **vmin0 (nm)** is the molar volume of the **nm**-th pure mineral.
V_g	**volg**	The volume of a perfect gas at 298.15°K and 1 bar pressure; $V_g = 22413.6\ cm^3$. It is unique to EQ3NR.
W_ψ, $W_{i\psi}$, $W_{ij\psi}$	**w**	Array of solid solution activity coefficient model parameters. Dimensioning: **w(iktpar,nxtpar)**. Usage: **w (i,nx)** is the **i**-th parameter for the **nx**-th solid solution. These are computed from the coefficients in the **apx** array.
	weight	A work array containing weighting factors for aqueous species used in evaluating balance equations. Dimensioning: **weight(nstpar)**. Usage: **weight (ns)** is the factor for the **ns**-th aqueous species.

N_w	**wfs**	Weight fraction of solvent water in aqueous solution.
$x_{\sigma\psi}$	**xbar**	Array of mole fractions of solid solution end-member components. Dimensioning: **xbar(iktpar,nxtpar)**. Usage: **xbar(ik,nx)** is the mole fraction of the **ik**-th component of the **nx**-th solid solution.
$x_{\sigma\psi}$	**xbarb**	Array of mole fractions of solid solution end-member components read from the **input** file. Dimensioning: **xbarb(iktpar,nxtpar)**. Usage: **xbarb(ikb,nxb)** is the mole fraction of the **ikb**-th component given for the **nxb**-th solid solution listed. This array is input file ordered. It is unique to EQ3NR.
$x_{\sigma\psi}$	**xbarh**	Array of mole fractions of solid solution end-member components corresponding to the compositions that maximize the saturation indices of the corresponding phases. Dimensioning: **xbarh(iktpar,nxtpar)**. Usage: **xbarh(ik,nx)** is the mole fraction of the **ik**-th component of the **nx**-th solid solution which maximizes the saturation index of this phase.
$log\ x_{\sigma\psi}$	**xbarlg**	Array of log mole fractions of solid solution end-member components. Dimensioning: **xbarlg(iktpar,nxtpar)**. Usage: **xbarlg(ik,nx)** is the log mole fraction of the **ik**-th component of the **nx**-th solid solution.
x_w	**xbarw**	The mole fraction of solvent water.
$log\ x_w$	**xbarlw**	The log mole fraction of solvent water.
I	**xi**	Ionic strength, molal.
$log\ K_{Eh}$	**xlkeh**	Log equilibrium constant of the half reaction relating the hypothetical electron and $O_{2(g)}$.
$log\ K_g$	**xlkg**	Array of log equilibrium constants of the dissolution reactions of gas species. Dimensioning: **xlkg(ngtpar)**. Usage: **xlkg(ng)** is the log equilibrium constant for the **ng**-th gas.
$log\ K_\phi$	**xlkm**	Array of log equilibrium constants of the dissolution reactions of pure minerals. Dimensioning: **xlkm(nmtpar)**. Usage: **xlkm(nm)** is the log equilibrium constant for the **nm**-th pure mineral.
	xlkmod	The *log K* alter option parameter for an alter/suppress option specified on the **input** file. Dimensioning: **xlkmod(nxmdpa)**. Usage: **xlkmod(n)** is the parameter for the **n**-th such option specified (see **jxmod**, **kxmod**, **uxmod**, and **nxmod**).
$log\ K_r$	**xlks**	Array of log equilibrium constants of the dissociation/destruction reactions of aqueous species. Dimensioning: **xlks(nrstpa)**. Usage: **xlks(nrs)** is the log equilibrium constant for the **nrs**-th such species.
$log\ Q/K$	**xlqk**	Array of saturation indices of pure minerals. Dimensioning: **xlqk(nmtpar)**. Usage: **xlqk(nm)** is the saturation index for the **nm**-th pure mineral.
$log\ Q/K$	**xqkx**	Array of saturation indices of solid solution end-member components. Dimensioning: **xqkx(iktpar,nxtpar)**. Usage: **xqkx(ik,nx)** is the saturation index for the **ik**-th component of the **nx**-th solid solution.

z_i	**z**	Array of electrical charges of the aqueous species. Dimensioning: **z(nstpar)**. Usage: **z(ns)** is the electrical charge of the **ns**-th aqueous species.
$\frac{z_i^2}{2}$	**zsq2**	Array of values equal to one half the charge squared of the aqueous species. Dimensioning: **zsq2(nstpar)**. Usage: **zsq2(ns)** is one half the charge squared for the **ns**-th aqueous species.
z	**zvclg1**	Array of logarithmic master iteration variables. Dimensioning: **zvclg1(kpar)**. Usage: **zvclg1(kcol)** *is the value of the* **kcol***-th logarithmic master iteration variable.*

Appendix B. Glossary of EQ3NR Modules

EQ3NR is a moderately large code. The source code consists of the main program and a number of subroutines. In addition, EQ3NR uses a number of modules from the EQLIB library. These are described in Appendix B of the EQ3/6 Package Overview and Installation Guide (Wolery, 1992). The modules are described as ".f" files, as this is how they are normally worked with under a UNIX operating system. For a description of the code architecture, see Chapter 10.

arrset.f This module is called by **eq3nr.f**. It sets up the matrix structure for Newton-Raphson iteration and estimates starting values for the iteration variables. The initial estimates are refined according to a pre-Newton-Raphson optimization algorithm. They may also be refined by automatic basis switching, if this feature is enabled.

arrsim.f This module is called by **arrset.f** to compute those basis species concentrations which by the nature of the chosen constraints must be estimated simultaneously.

balcon.f This module is called by **matrix.f**. It writes the part of the Jacobian matrix that consists of rows corresponding to balance equations (mass and charge balance).

betas.f This module is called by **arrset.f** and the EQLIB modules **newton.f** and **nrstep.f** (these latter two modules call it as **betae**). It computes the residual functions (α, β, and β_{max}).

dawfix.f This module is called by **arrsim.f**. It determines whether or not the problem input constraints directly fix the activity of water, as for example would happen if one constrained dissolved calcium by equilibrium with gypsum and dissolved sulfate by equilibrium with anhydrite.

echox.f This module is called by **eq3nr.f**. It writes a description of the input problem on the **output** file. Unlike the "instant echo" of the input file, this description includes data file statistics, any default values that were chosen by EQ3NR, and any changes in the **input** constraint options made by the code to resolve inconsistencies among these constraints or with truncation limits imposed on certain run parameters.

eq3nr.f This is the main program of EQ3NR. It supervises the speciation-solubility calculation. It connects the necessary files (**input**, **data1**, **output**, **pickup**, and **rlist**). It initializes key arrays and sets default values for important run parameters. It also does an extensive checking of the problem that has been input, looking for inconsistencies and missing inputs that are required by the options chosen.

flgstx.f This module is called by **eq3nr.f**. It sets up the species status switch arrays **jsflag**, **jmflag**, **jkflag**, **jxflag**, and **jgflag**.

gases.f This module is called by **scripx.f**. It computes the equilibrium fugacities of gas species.

gcsp.f The module is called by **indatx.f**. It interprets the constraints (**jflag**, **csp**) for basis species listed on the **input** file by matching the species names with those read from the supporting data file.

getrdx.f This module is called by **rdtyp4.f**. It gets the redox constraints specified on the **input** file in "D" format.

getspc.f This module is called by both **getrdx.f** and **rdtyp4.f**. It gets input constraints for the basis species listed on the **input** file in "D" format.

getss.f This module is called by both **getspc.f** and **rdtyp5.f**. It gets solid solution data that are part of constraints listed on the **input** file in "D" format.

indatx.f This module is called by **eq3nr.f**. It reads the data file **data1**.

init3v.f This module is called by **rdninp.f**.

matrix.f This module is called by the EQLIB routine **nrstep.f** (which calls it by the name of **matrxe**). It computes the Jacobian matrix (*J*).

ncmpx.f This module is called by **arrset.f** and the EQLIB modules **ngcadv.f** and **nrstep.f** (which call it by the name of **ncmpe**). It computes all parameters that derive from the primary iteration variables and are necessary to write the Jacobian matrix (e.g., all aqueous species concentrations and activities).

ndiagx.f This module is called by **eq3nr.f**. If Newton-Raphson iteration fails, it attempts to generate diagnostics.

rd3tds.f This module is called by **rdninp.f**. It gets the input for total dissolved salts from the **input** file in "D" format.

rdinp.f This module is called by **eq3nr.f**. It oversees reading the **input** file. If the **input** file is in "W" format, it calls **readx.f** to carry out this function. If the **input** file is in "D" format, it calls **rdninp.f** to do this.

rdninp.f This module is called by **rdinp.f**. It carries out reading the **input** file in "D" format.

rdtyp1.f This module is called by **rdninp.f**. It gets the input for temperature and density from the **input** file in "D" format.

rdtyp2.f This module is called by **rdninp.f**. It gets the input for electrical balancing from the **input** file in "D" format.

rdtyp4.f This module is called by **rdninp.f**. It gets the input for the basis species constraints from the **input** file in "D" format. It does this by calling **getrdx.f** and **getspc.f**.

rdtyp5.f This module is called by **rdninp.f**. It gets the input for solid solutions from the **input** file in "D" format. It calls **getss.f**.

rdtyp9.f This module is called by **rdninp.f**. It gets the input for **tolbt**, **toldl**, **tolsat**, and **itermx** from the **input** file in "D" format.

readx.f This module reads the **input** file in "W" format. It writes an "instant echo" of this **input** data on the **output** file. It contains full internal documentation.

scribo.f This module is called by **eq3nr.f**. It writes the **pickup** file in "W" format.

scribx.f This module is called by **eq3nr.f**. It writes the **pickup** file in "D" format.

scripx.f This module is called by **eq3nr.f**. This module writes the results of the speciation-solubility calculation on the **output** file.

setup.f This module is called by **eq3nr.f**. This module converts input concentration data other than molalities to molalities.

Appendix C. EQ3NR Error Messages

All EQ3/6 error messages fit into one of three categories: *error*, *warning*, and *note*. An *error* implies a fatal error. Execution of the current input problem will cease without completion, immediately in some cases, later in others. Which is the case depends on whether it makes more sense to stop immediately or to continue checking for other errors before ceasing execution. A *warning* indicates a condition which may or may not represent a real error. A *note* indicates a condition which could possibly indicate an error, but normally does not. All three types of messages are written to both the screen file and the **output** file. If an *error* message is issued, analysis of the problem may be facilitated by checking any preceding *error*, *warning*, or *note* messages.

Each EQ3/6 error message has the following format:

* *msgtype* - (*source*/*module*) *Message*.

where *msgtype* = *error*, *warning*, or *note*, *source* is the root name of the source file (e.g., *eqlib*, *eqpt*, *eq3nr*, or *eq6*) containing the *module*, *module* is the name of the module (main program or subroutine) which writes the message, and *Message* is the message itself. The messages are designed to be as self-explanatory as possible. The messages are reproduced here using *AAAA* to stand for a character variable, *IIII* for an integer, and *RRRR* for a floating point number.

Most of the error messages that users are likely to encounter deal with problems regarding the input file, the data file, or both of these. In most instances, the meaning of these messages should be immediately clear to the user. In other instances, it may be necessary to search out other information. In such cases, there are three principal actions that users should take. The first is to check the **output** file for additional diagnostic messages (*warnings* and *notes*) which may bear on the matter. If this does not suffice to identify corrective action, compare the instant echo of the **input** file on the **output** file with the original **input** file. You may find that certain data were not entered in the correct fields, that certain inputs fail to correspond with the necessary lines to follow, or that a line is missing or you have an extra line. In addition, it may help to re-run the problem with the debugging option switch **iodb1** set to 1 or 2. This will trigger the printing of additional information which should help to identify the problem. A small number of messages deal with installation errors. These should also be quite clear.

Some messages deal with programming errors. The user should see these rarely if ever. These are likely to appear somewhat more cryptic to users. Problems of this type must be dealt with by diagnosing the problem (probably with the help of a symbolic debugger) and modifying the code. Most users should probably not attempt corrective action of this sort. The code custodian should be notified of suspected programming errors and may be able to provide fixes.

Some of the messages displayed in this appendix are followed by *Comments* that may help to explain them. Users of EQ3NR may also encounter error messages from EQLIB modules. These messages are listed in similar format in the EQ3/6 Package Overview and Installation Guide (Wolery, 1992). The *errors* are listed first, then the *warnings* and finally the *notes*.

Message: * error - (eq3nr/arrset) Have encountered an automatic basis switching error in loop IIII.

Comment: This is a programming error.

Message: * error - (eq3nr/arrsim) The phase assemblage by the specified solubility constraints fixes the activity of water. The code is presently unable to solve problems of this type.

Message: * error - (eq3nr/arrsim) The specified counterion "AAAA" in the jflag = 17 or 18 option for "AAAA" is not in the basis set.

Message: * error - (eq3nr/arrsim) The speciation model appears to be singular. There is probably a violation of the mineralogic phase rule in the set of solubility constraints specified on the input file.

Message: * error - (eq3nr/betas) Have encountered an illegal jflag value = IIII for the species "AAAA".

Message: * error - (eq3nr/eq3nr) Can not use the jflag= IIII option for "AAAA" because this species is suppressed.

Message: * error - (eq3nr/eq3nr) Can not use the jflag= IIII option for "AAAA" because this species has no charge.

Message: * error - (eq3nr/eq3nr) Can not use the jflag= IIII option for "AAAA" because the specified counterion "AAAA" is not in the active basis set.

Message: * error - (eq3nr/eq3nr) Can not use the jflag= IIII option for "AAAA" because the specified counterion "AAAA" is not in the present system.

Message: * error - (eq3nr/eq3nr) Can not use the jflag= IIII option for "AAAA" because the specified counterion "AAAA" is suppressed.

Message: * error - (eq3nr/eq3nr) Can not use the jflag= IIII option for "AAAA" because the specified counterion "AAAA" has zero charge.

Message: * error - (eq3nr/eq3nr) Can not use the jflag= IIII option for "AAAA" because the specified counterion "AAAA" has the same charge sign.

Message: * error - (eq3nr/eq3nr) Choosing iopt1 = -3 requires setting jflag(o2(g)) to 19, 20, or 21. The input file has jflag(o2(g)) = IIII.

Message: * error - (eq3nr/eq3nr) The species "AAAA" was selected to determine the redox state, but the associated reaction is not a redox reaction.

Message: * error - (eq3nr/eq3nr) The species "AAAA" was selected to determine the redox state, but it is not in the present system.

Message: * error - (eq3nr/eq3nr) The species "AAAA" was selected to specify the redox state, but it has a jflag value of IIII. The jflag for such species can not have a value of 27 or 30.

Comment: You have to enter some actual data for this species. The jflag = 27 and 30 options do not correspond to input of actual data.

Message: * error - (eq3nr/eq3nr) The species "AAAA" was selected to specify the redox state, but the necessary corresponding strict basis species representing the complementary oxidation state is not in the present system.

Comment: To use this option for the redox state, you must enter data for both the auxiliary basis species whose associated reaction defines the couple, and the strict basis species which represents the other half of the redox couple and appears in that reaction.

Message: * error - (eq3nr/eq3nr) Have illegal negative csp value of RRRR for the basis species "AAAA".

Message: * error - (eq3nr/eq3nr) Did you mean to specify alkalinity for "AAAA"? Because of theoretical problems with the concept and not uncommon analytical difficulties, EQ3/6 no longer allows the use of alkalinity.

Comment: If you have not already done so, read the section on alkalinity in Chapter 2.

Message: * error - (eq3nr/eq3nr) The species "AAAA" can not be constrained by the reaction-

Comment: This is followed by the reaction in question.

Message: * error - (eq3nr/eq3nr) The species "AAAA" can not be constrained by equilibrium with a solid solution because iopt4 = IIII, not 2.

Comment: In order to use such a constraint, it is necessary to input the composition of the solid solution.

Message: * error - (eq3nr/eq3nr) The species "AAAA" has a jflag value of IIII. This is not permitted. This species is in the strict basis set and therefore has no associated reaction in which it is destroyed.

Message: * error - (eq3nr/eq3nr) The following heterogeneous reaction constraint has been used more than once-

Comment: This is followed by the reaction in question.

Message: * error - (eq3nr/eq3nr) The input file has jflag(o2(g) = IIII. This requires iopt1 to be -3, but iopt1 is now set to IIII.

Message: * error - (eq3nr/eq3nr) Hybrid newton-raphson iteration failed after IIII steps. Calling diagnostics routine.

Message: * error- (eq3nr/gcsp) The basis species "AAAA" was specified more than once on the input file.

Message: * error - (eq3nr/getrdx) When specifying redox parameters, column four must contain "eh", "pe", "fo2lg" or "redox couple". Check INPUT file line: IIII

Message: * error - (eq3nr/getrdx) Break line or end-of-file found where redox couple input was expected. Check INPUT file line: IIII

Message: * error - (eq3nr/getspc) Species was not specified in column 1. Check INPUT file line: IIII

Message: * error - (eq3nr/getspc) Maximum number of allowed entries: IIII. Have exceeded this value on INPUT file line IIII

Message: * error - (eq3nr/getspc) The "pH" species can only be "h+". Check INPUT file line: IIII

Message: * error - (eq3nr/getspc) The "pHCl" species can only be "h+". Check INPUT file line: IIII

Message: * error - (eq3nr/getspc) Column 2 can not be blank for the "gas", "mineral", "log activity combo", and "log mean activity" constraints. Check INPUT file line: IIII

Message: * error - (eq3nr/getspc) Column 2 must left blank for the "log activity", "ph", and "phcl" constraints. Check INPUT file line: IIII

Message: * error - (eq3nr/getss) Maximum number of allowed entries: IIII. Have exceeded this value on INPUT file line: IIII

Message: * error - (eq3nr/getss) Must specify all end-members of a solid solution (mole fractions must sum to 1.0). Check above INPUT file line: IIII

Message: * error - (eq3nr/getss) Solid solution name can not be blank. Check INPUT file line: IIII

Message: * error - (eq3nr/getss) End-member names can not be blank. Check INPUT file line: IIII

Message: * error - (eq3nr/getss) Maximum number of end-members per solid solution: IIII. Check above INPUT file line: IIII

Message: * error - (eq3nr/indatx) Have wrong file header = "AAAA" on the data file. The first five characters must be "data1".

Comment: Make sure you have not defined **data1** *to be a* **data0** *file.*

Message: * error - (eq3nr/indatx) The number of chemical elements on the data file is IIII. This exceeds the dimensioned limit (nctpar) of IIII.

Message: * error - (eq3nr/indatx) The number of basis species on the data file is IIII. This exceeds the dimensioned limit (nsqpar) of IIII.

Message: * error - (eq3nr/indatx) The composition of species "AAAA" on the data file is described in terms of an unrecognized chemical element called "AAAA". If this is an actual element, it is not in the list of chemical elements on this data file.

Message: * error - (eq3nr/indatx) The maximum number of aqueous species (nstpar) is IIII. This has been exceeded while trying to read the data file.

Message: * error - (eq3nr/indatx) The reaction for the destruction of the species "AAAA" is written on the data file in terms of an unrecognized basis species called "AAAA". If this is an actual species, it is not among the basis species on this data file.

Message: * error - (eq3nr/indatx) The maximum number of pure minerals (nmtpar) is IIII. This has been exceeded while trying to read the data file.

Message: * error - (eq3nr/indatx) The maximum number of gas species (ngtpar) is IIII. This has been exceeded while trying to read the data file.

Message: * error - (eq3nr/indatx) The basis species "AAAA" was referenced on the input file but was not read from the data file.

Comment: The name of the basis species may contain a typographical error. Otherwise, the data file does not contain this species. You may be using the wrong data file.

Message: * error - (eq3nr/indatx) The maximum number of solid solutions (nxtpar) is IIII. This has been exceeded while trying to read the data file.

Message: * error - (eq3nr/indatx) The species "AAAA" is specified in user-directed basis switches to switch with both "AAAA" and "AAAA".

Message: * error - (eq3nr/indatx) The species "AAAA" is specified in a user-directed basis switch to switch with "AAAA", but is not in the list of aqueous species read from the data file.

Message: * error - (eq3nr/indatx) The species "AAAA" is required for an abs(zj)*log ai +/- abs(zi)*log aj constraint but was not read from the data file.

Message: * error - (eq3nr/indatx) The species "AAAA" is required for a log a(+/-,ij) constraint but was not read from the data file.

Comment: " a(+/-,ij)" denotes the mean activity of ions i and j.

Message: * error - (eq3nr/indatx) The species "AAAA" is required for a heterogeneous equilibrium constraint but was not read from the data file.

Message: * *error* - (eq3nr/matrix) Have encountered an illegal jflag value = IIII for the species "AAAA".

Comment: This is a programmer error.

Message: * error - (eq3nr/rdninp) Looking for heading: "AAAA" Found string: "AAAA" See INPUT file line: IIII

Message: * error - (eq3nr/rdninp) End-of-file unexpectedly encountered. See INPUT file line: IIII

Message: * error - (eq3nr/rdninp) Errors encountered while reading the INPUT file. Process stopped

Message: * error - (eq3nr/rdninp) The electrical balancing species "AAAA" was not found among the basis species listed on the input file.

Message: * error - (eq3nr/rdninp) The species "h+" must be specified in the species list on the input file.

Message: * error - (eq3nr/rdtyp2) Only one electrical balancing option can be selected. Check INPUT file line: IIII

Message: * error - (eq3nr/rdtyp9) Unrecognized tolerance variable "AAAA". Check INPUT file line: IIII

Message: * error - (eq3nr/rdtyp9) Invalid tolerance descriptor string "AAAA". Check INPUT file line: IIII

Message: * error - (eq3nr/readx) The input file has no title.

Message: * error - (eq3nr/readx) The maximum number of nxmod options (nxmdpa) is IIII. This has been exceeded while trying to read the input file.

Message: * error - (eq3nr/readx) The maximum number of basis species (nsqpar) is IIII. This has been exceeded while trying to read the data file.

Message: * error - (eq3nr/scribo) Bad pickup file has been written. The basis species "AAAA" appears on it but is not in the model.

Comment: The **pickup** *file contains a strict basis species that is not in the model. You used an auxiliary basis species which is linked to this strict basis species. The EQ3NR calculation is still valid, but you can not use the* **pickup** *file to input the solution model to EQ6. If you want to do this, re-run the problem switching the auxiliary basis species in question into the strict basis set.*

Message: * error (eq3nr/setup) An undefined jflag value of IIII was specified for the basis species "AAAA".

Message: * warning - (eq3nr/eq3nr) The input file contains conflicting redox options - ioptl = IIII overrides jflag(o2(g)) = IIII.

Message: * warning - (eq3nr/eq3nr) The input file contains conflicting redox options - iopt1 = IIII overrides uredox = "AAAA".

Message: * warning - (eq3nr/eq3nr) The species "AAAA" was specified to be adjusted for electrical balance, but it is not in the set of basis species.

Message: * warning - (eq3nr/eq3nr) The species "AAAA" was specified to be adjusted for electrical balance, but it has no electrical charge. Success will depend on its concentration affecting that of one or more charged species with which it is in equilibrium.

Message: * warning - (eq3nr/eq3nr) The species "AAAA" was specified to be adjusted for electrical balance, but it is not in the present model.

Message: * warning - (eq3nr/eq3nr) Tried to pick a species to be adjusted for electrical balance, but did not succeed.

Message: * warning - (eq3nr/eq3nr) The species "AAAA" was specified to define the default redox state, but is not in the active auxiliary basis set.

Message: * warning - (eq3nr/eq3nr) The strict basis species "AAAA" has an illegal jflag value of IIII.

Message: * warning - (eq3nr/readx) The input line for o2(g) will be ignored because iopt1 is not set to -3.

Message: * note - (eq3nr/arrset) Can not switch the species "AAAA" out of the basis because it is tied up in a jflag = 17 or 18 option for another species.

Message: * note - (eq3nr/arrset) Optimization ended outside requested limits.

Comment: Don't panic. Hybrid Newton-Raphson iteration may still succeed. If so, the results of the run are still perfectly good. If it doesn't succeed, see the comment following the following message.

Message: * note - (eq3nr/arrset) Optimization ended far outside requested limits.

Comment: Don't panic. Hybrid Newton-Raphson iteration may still succeed, though the probability of this is significantly diminished. If it does succeed, the results of the run are still perfectly good. If it doesn't succeed, re-run with **iodb** *set to 1 or 2 in order to see what is going on during the optimization stage. You may be able to get the problem to converge by directing the code to make certain basis switches.*

Message: * note - (eq3nr/arrsim) The matrix solver routine (eqlib/msolvr) failed.

Message: * note - (eq3nr/arrsim) The species "AAAA" has a required concentration near RRRR molal.

*Comment: Don't panic. This is not always fatal. If the code fails to solve the problem, however, this probably means that one or more equilibrium constraints (***jflag** *= 19, 20, 21, or 27) are such that no sensible answer exists to the problem currently posed. Think about what you are asking the code to do.*

Message: * note - (eq3nr/echox) The species "AAAA" is not in the model.

Message: * note- (eq3nr/eq3nr) The input value for the iopr9 print option flag is inconsistent with the specified value of iopg1 (choice of model for activity coefficients of aqueous species). The iopr9 flag has been reset to 0.

Message: * note - (eq3nr/eq3nr) Have picked the species "AAAA" to be adjusted for electrical balance. Any other specified constraint will be overridden.

Message: * note - (eq3nr/eq3nr) Could not compute acceptable starting estimates for Newton-Raphson iteration. Will therefore not attempt it.

Comment: Re-run with **iodb** *set to 1 or 2 in order to see what is going on during the optimization stage. You may be able to get the problem to converge by directing the code to make certain basis switches.*

Message: * note - (eq3nr/eq3nr) Can not write a valid pickup file for the current problem because the auxiliary basis species in each of the following strict-auxiliary basis species pair(s) is present in the model while the corresponding strict basis species is not. Add a trivial amount of each such strict basis species to the model, or switch it from the strict set to the auxiliary set and suppress it if necessary.

Comment: The relevant data follow this message. The **pickup** *file contains a strict basis species that is not in the model. You used an auxiliary basis species which is linked to this strict basis species. The EQ3NR calculation is still valid, but you can not use the* **pickup** *file to input the solution model to EQ6. If you want to do this, re-run the problem switching the auxiliary basis species in question into the strict basis set.*

Message: * note - (eq3nr/getspc) A non-zero concentration should not be specified as an input parameter for a "mineral", "solid solution", "dependent", or "eliminated" species. Check INPUT file line: IIII

Message: * note - (eq3nr/ndiagx) The ion being adjusted to achieve electrical balance is crashing to zero. Electrical balancing must be done on an ion of opposite charge.

Message: * note - (eq3nr/ndiagx) The oxygen fugacity is crashing, probably because a bad electrical balance constraint on h+ is causing the concentration of that species to crash to zero.

Message: * note - (eq3nr/ndiagx) The oxygen fugacity is crashing, probably because of a bad constraint on one of the aqueous species appearing in the redox reaction that is being used to constrain the redox state.

Message: * note - (eq3nr/ndiagx) No diagnostics were generated from the failed iteration. Look at the del and beta data in the iteration summary for clues to why it did not work.

Appendix D. Notes on Known Bugs and Such

This appendix presents notes on known bugs and other known unusual phenomena concerning EQ3NR.

1. The code tends to converge slowly or not at all in dealing with extremely concentrated electrolyte solutions. Roughly speaking, these are solutions whose ionic strengths are greater than about 12 molal. However, there is no simple way to accurately categorize the code's performance envelope.

For a complete list of known bugs and such for EQ3/6, see Appendix D of the EQ3/6 Package Overview and Installation Guide (Wolery, 1992).