SimSci™

by Schneider Electric

SimSci®

PRO/II® 10.0 Electrolyte User Guide

July 2016

All terms mentionedin this documentation that are known to be trademarks or service marks have been appropriately capitalized. Schneider Electric Software, LLC. cannot attest to the accuracy of this information. Use of a term in this documentation should not be regarded as affecting the validity of any trademark or service mark.

ActiveFactory, ArchestrA, ARPM, AssayCentral, Autonomous, Avantis, Avantis.PRO, Connoisseur, CrudeManager, CrudeSuite, DATACON, DATAPREP, DT Analyst, DYNSIM, DYNSIM Power, Esscor, EYESIM, Factelligence, FactorySuite, FactorySuite A², Foxboro, Foxboro Evo, FSIM Plus, HEXTRAN, Hot Links, I/A Series, IMPACT, IMServ, InBatch, InControl, Industrial Portal, IndustrialSQL Server, InFusion, INPLANT, InSQL, IntelaTrac, InTouch, IPact, Invensys, Logic Validator, MBM, NETOPT, PIPEPHASE, PRO/II, PROVISION, QI Analyst, ROMeo, SCADAlarm, SIM4ME, SimCentral, SimSci, SimSci-Esscor, Skelta, Skelta BPM.NET, Skelta Sharepoint, Skelta Sharepoint Accelerator, Spiral Software, SuiteVoyager, TACITE, Tactical Batch, The Simulator, TRISIM Plus, VISUAL FLARE, VISUAL FLOW, VISUAL FLOW SUITE, VISUAL NETWORK, Visual Solutions, VOYAGER, WindowMaker, WindowViewer, Wonderware, Wonderware InTouch, and ZI ZHI are trademarks of Schneider Electric SE, its subsidiaries and affiliated companies.

© 2016 by Schneider Electric Software, LLC. All rights reserved.

No part of this documentation shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of Schneider Electric Software, LLC. No liability is assumed with respect to the use of the information contained herein.

Although precaution has been taken in the preparation of this documentation, Schneider Electric Software, LLC assumes no responsibility for errors or omissions. The information in this documentation is subject to change without notice and does not represent a commitment on the part of Schneider Electric Software, LLC. The software described in this documentation is furnished under a license agreement. This software may be used or copied only in accordance with the terms of such license agreement.

ArchestrA, Avantis, DYNSIM, EYESIM, Foxboro, Foxboro Evo, I/A Series, InBatch, InduSoft, IntelaTrac, InTouch, PIPEPHASE, PRO/II, PROVISION, ROMeo, Schneider Electric, SIM4ME, SimCentral, SimSci, Skelta, SmartGlance, Spiral Software, VISUAL FLARE, WindowMaker, WindowViewer, and Wonderware are trademarks of Schneider Electric SE, its subsidiaries, and affiliated companies. An extensive listing of Schneider Electric Software, LLC trademarks can be found at: http://software.schneider-electric.com/legal/trademarks/. All other brands may be trademarks of their respective owners.

Schneider Electric Software, LLC
26561 Rancho Parkway South
Lake Forest, CA 92630 U.S.A.
(949) 727-3200

http://software.schneider-electric.com/

Contents

Chapter E1
Electrolyte Module

Chapter E2
Electrolyte Models

Chapter E3
Creating a User-Added Electrolyte Model

Chapter E4
Electrolyte Effects In Simulations

Chapter E1
Electrolyte Module

Overview

This chapter describes the Electrolyte add-on module available for use in PRO/II. It describes electrolyte-related data entry using PROVISION, the Graphical User Interface (GUI). It also describes keyword entries in the General, Component, and Thermodynamic sections of input data.

Contents of the Electrolyte Module

The Electrolyte Module provides a set of programs, data libraries, and other files, including the following:

- The *Lite Public Electrolyte Library* provides aqueous data for over 2300 electrolyte components. This library is available only with the Electrolyte Module. It also includes the *GeoChem* bank provides aqueous data for over 140 additional mineral and metallic components frequently of interest.
- Version 7.0 of the Electrolyte Calculation engine from OLI System, Inc.
- Chemistry Wizard from OLI Systems, Inc., for creating user-added electrolyte models, private databanks, and executing standalone flash drum calculations and sensitivity analyses.
- 40 different aqueous electrolyte models arranged in 8 groups. Three versions of each model are provided.
  - The first version was created using Chemistry Wizard 3.1 for use in version 7.0 of the OLI calculation engine.
  - The second version was created using Chemistry Wizard version 1.0 for use in version 6.6 of the OLI calculation engine.
  - The third version was created by the (now obsolete) Electrolyte Utility Package (EUP) for use in version 5.6 of the OLI calculation engine.

The OLILIB data bank is part of the standard PROII component library, and not actually part of the Electrolyte module. It contains all necessary data for almost 800 electrolyte species.

Capabilities of the Electrolyte Module

The Electrolyte Module supports only aqueous electrolyte models. These systems typically contain water with up to about 30 percent solutes.

The Electrolyte Module includes only the *Lite Public Electrolyte Library* that provides aqueous data for about 2300 components and for about 140 *GeoChem* components.

Compatibility of PRO/II With Older Electrolyte Versions

PRO/II, the Electrolyte Module, and the OLI Calculation Engine installed for the Electrolyte Module are compatible with all earlier versions of electrolyte models previously supported by PRO/II.

Earlier versions of pre-generated aqueous models are based on older data file formats. These are exemplified by the older versions of the electrolyte models (described above) included in the Electrolyte Module. While PRO/II still is able to support them, continuing evolution of the format may prevent continued support in the future. For this reason, customers should consider upgrading to the newest version of the data files at their earliest convenience.

Although older versions of aqueous electrolyte models are included and function properly, the programs that created them no longer are available. When creating new models, users should use the current Chemistry Wizard version 3.1.

Alternatives to the Electrolyte Module

OLI Systems, Inc. recently introduced their Mixed Solvent Electrolytes package (MSE) that handles stronger concentrations of solutes. They also offer an expanded component library, currently containing data for over 5100 components.

PRO/II is fully capable of utilizing all the features of the MSE package and the expanded libraries from OLI. This still includes support for all existing aqueous models. Separate licensing is required.

Contact your PRO/II representative for licensing information and further technical details.

Included Pre-Built Electrolyte Thermodynamic Models

Eight groups of electrolyte models are available after installing the Electrolyte Module. When none of the pre-built models are satisfactory for your needs, use the Chemistry Wizard to create custom models.

PRO/II simulates aqueous systems in a wide range of industrial applications. The models apply to fixed component lists with a predefined set of thermodynamic methods for K-values, enthalpies, and densities. Table E2-2 on page E2-3 lists all 40 electrolyte models available for the following eight electrolyte groups in this release:

Amine Systems	**Caustic Systems**
Acid Systems	**Benfield Systems**
Mixed Salt Systems	**Scrubber Systems**
Sour Water Systems	**LLE and Hydrate Systems**

The Graphical User Interface uses these groups to organize the 40 pre-generated electrolyte models for convenient selection.

Guidelines For Using Electrolyte Thermodynamics

Electrolyte models include calculation methods for all essential properties. Proper usage requires observing these restrictions:

- Always declare an electrolyte model as a *system* (e.g., `METHOD SYSTEM=DIPA70`).
- It is not possible to define individual methods for K-value, enthalpy, or density when using electrolyte thermodynamic models.
- Electrolyte models cannot be used to calculate the following:
  - ➢Non-aqueous systems.
  - ➢Free water decant (i.e., non-rigorous LLE)
  - ➢Water dew points
  - ➢Hydrocarbon dew points
  - ➢Entropy and heat capacity
- Aqueous Electrolyte models cannot be used to calculate non-aqueous system. All the pre-generated models in the Electrolytes Module are aqueous electrolyte systems.

- Be careful when using both non-electrolyte and electrolyte thermodynamic methods in the same simulation.
  - ➢The PRO/II electrolytic models use a different enthalpy basis from that used for other thermodynamic systems. When both are used, PRO/II automatically takes care of this difference, but it may appear to be confusing.
  - ➢To avoid this, select the electrolyte enthalpy method for all non-electrolyte thermodynamic systems in a mixed application. All systems then use the electrolyte model basis.

Unit Operations that support the Electrolyte Module

When electrolyte components are present in a simulation, PRO/II calls the OLI calculation engine to calculate all their thermodynamic properties. This is essentially a flash calculation algorithm the behaves somewhat differently than the internal PRO/II flash. Because of the differences, and due to the data requirements of some unit operations, full support for electrolytes is inappropriate or partially restricted in some unit operations.

The following is a partial listing of unit operations that support the Electrolyte add-on:

Flash drum	**Mixer**	**Calculator**
Column ELDIST algorithm	**Pipe**	**Heating/Cooling curves**
Controller	**Pump**	**Stream Calculator**
Optimizer	**Splitter**	**Valve**
Heat exchangers **Simple HX,** **Rigorous HX,** **LNG**	**Reactors** **Conversion,** **Equilibrium,** **Batch**	

Refer to Chapter E4, *Electrolyte Effects In Simulations*, for a discussion of the limitations of using electrolytes in PRO/II. The subsection “Basic Unit Operations” on page E4-3 specifically addresses concerns for individual unit operations, including descriptions of electrolyte-specific features and PRO/II features not available for units using electrolyte thermodynamic methods. Summaries of keyword input files for the Flash Drum (as an example of a standard PRO/II unit with a few changes in functionality) and for the `ELDIST` Distillation unit (which is specially designed for PRO/II

Electrolytes (2)). Examples of keyword input files for units using Electrolyte thermodynamic methods also are provided.

How to Use the Electrolyte Module in PRO/II

Preliminaries

The following activities must be completed before attempting to model electrolyte components in a simulation.

1. Install the Electrolyte Add-on Module and the optional Chemistry Wizard program. Installation details are provided when the licenses are acquired. More general information about installing optional modules is available in the *Getting Started Guide*.
2. Install at least one electrolyte thermodynamic model in the form of a data base (.dbs) file. There are three options for accomplishing this:

   Use a pre-generated model

   The Electrolyte module installed with PRO/II includes 40 *pre-generated* data base files. Three versions of each data base file are provided. Refer to “Pregenerated Models” in Chapter E2, *Electrolyte Models*.

   Use an older existing user-defined model

   Early versions of PRO/II allowed user-defined electrolyte models having a pre-defined name M41.dbs through M60.dbs. If you still have some of these files from previous versions, PRO/II still can use them.

   Create and use a new user-defined model

   Use the Chemistry Wizard program to create a new user-defined electrolyte model. Complete information is available in the *OLI Chemistry Wizard User Guide*, located in the \Manuals\OLI\ directory installed with PRO/II. Also refer to “User-Added Models” on page E2-8 to learn about some advantages gained by using this tool.

Overview of Using Electrolytes In A Simulation

Once at least one electrolyte data base file is available, it is quite straightforward to use it in a simulation. The following actions usually are involved.

1. Declare an Electrolyte thermodynamic method in an existing or new simulation.
2. Generate the electrolyte components from the electrolyte model. This is automatic in keywords and Run Batch. A dialog provides this in the GUI.
3. Add generated electrolyte components to at least one stream.
4. Configure at least one stream containing electolytes as a feed to at least one unit operation that supports electrolytes.
5. Assign electrolyte thermodynamic methods to appropriate unit operations.

The following sections describe each of these steps.

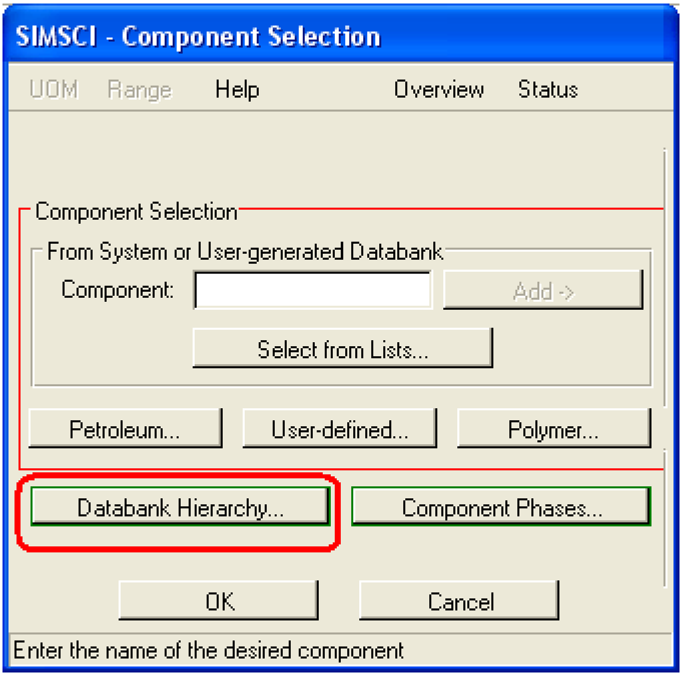
GUI Access to Electrolytes

Set Data Bank Search Hierarchy

When using Electrolyte thermodynamic methods with library components, the OLILIB component databank must be searched first and the SIMSCI databank second. To set this hierarchy,

➢Click the *Hierarchy* button highlighted in the *Component Selection* dialog shown in Figure E1-1.

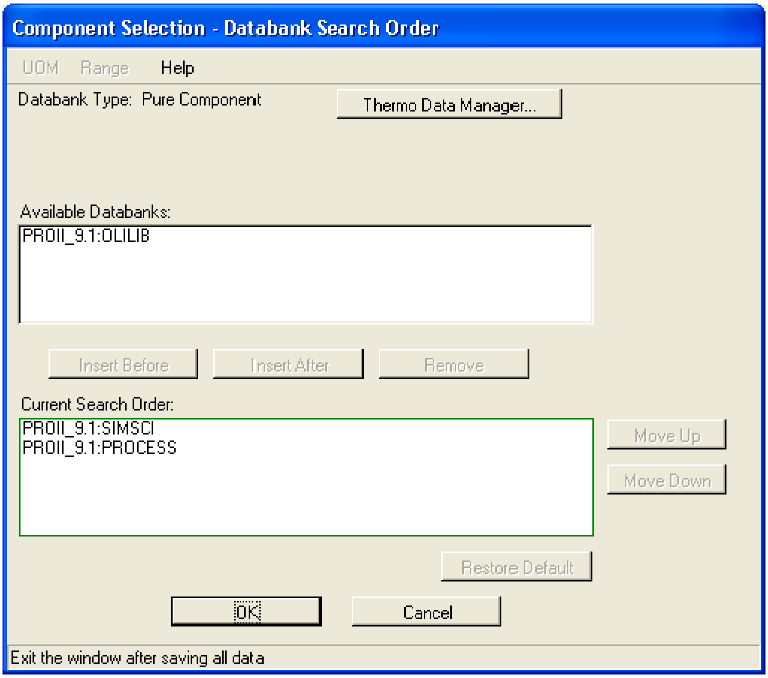
Figure E1-1: Changing the Data Bank Hierarchy



Enter the name of the desired component

➢As shown in Figure E1-2, Highlight the *PROII_10.0:OLILIB* entry in the *Available Databanks:* field; then click *Insert Before* to move the OLILIB data bank to the top of the Current Search Order.

Figure E1-2: Adding Available Data Banks



Component Selection - Databank Search Order

Databank Type: Pure Component

Available Databanks:

PROII_9.1:OLILIB

Current Search Order:

PROII_9.1:SIMSCI
PROII_9.1:PROCESS

Exit the window after saving all data

See the online help for more information on setting the databank search order hierarchy.

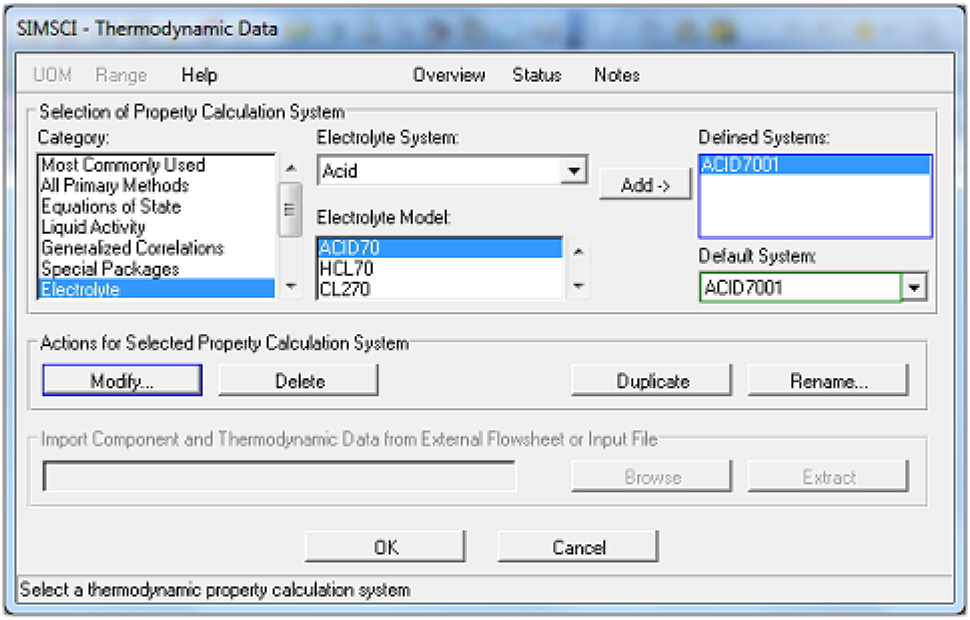
Notice in Figure E1-2 that an additional user-added library, MyElec1, containing data bank NEWELEC1, is highlighted in the *Available* list. custom libraries such as this also may be added to the search order. Refer to the *TDM User Guide* to learn about creating custom component data libraries.

Select An Electrolyte Model

The next essential step is to add an electrolyte model as a PRO/II thermodynamic METHOD set. If it is the only METHOD set in the simulation, or if it is designated the default METHOD set, all unit operations use it by default.

➢Click the *Thermodynamics Data* button on the tool bar to display the *Thermodynamics Data* dialog box (Figure E1-3).

Figure E1-3: Thermodynamic Data Dialog Box



Category: Electrolyte

Electrolyte System: Acid

Electrolyte Model: ACID70

Defined Systems: ACID7001

Default System: ACID7001

Select a thermodynamic property calculation system

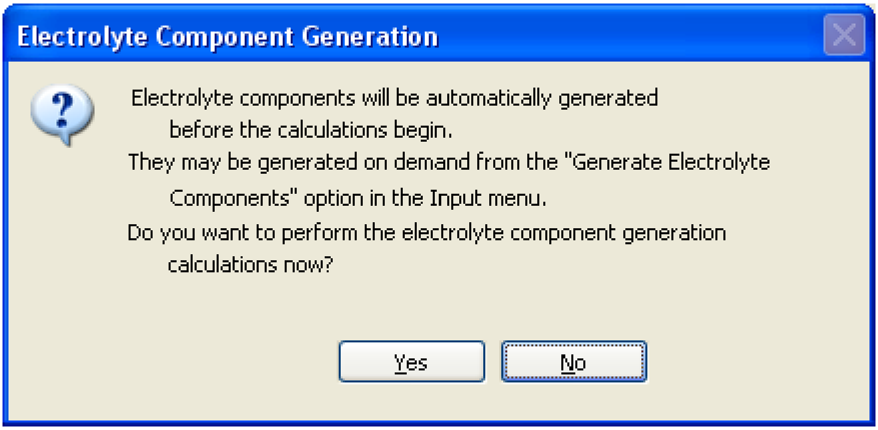
- Select *Electrolyte* in the *Category* list.
- Select a group from the *Electrolyte System:* drop down list box
- Choose an electrolyte model from the *Electrolyte Model:* drop down list box
- Click the *Add* button to add the model to the simulation.
- Click *OK* to save the model as part of the simulation, and to generate the electrolyte components.

Generate Electrolyte Components

Generating electrolyte components means extracting electrolyte data from the thermodynamic METHOD set and configuring it in a form that is usable by PRO/II. In Run Batch and Keyword operation, generation always is automatic.

- In the GUI, a dialog provides the option to generate the electrolyte components automatically when clicking the *OK* button (Figure E1-3) after an electrolyte model is selected. Figure E1-4 illustrates the dialog.

Figure E1-4: Generating Components From Electrolyte



Electrolyte Component Generation

Electrolyte components will be automatically generated before the calculations begin.
They may be generated on demand from the "Generate Electrolyte Components" option in the Input menu.
Do you want to perform the electrolyte component generation calculations now?

Thermodynamics

- ➢To defer generation, select “NO” in response to this query. This allows on-demand generation at a later time.

To generate the electrolyte components on demand:

- ➢Select *Generate Electrolyte Components* from the *Input* menu.

All electrolyte components for the flow sheet (including compounds and solid precipitates) are generated. There are no other options. Also, if this step is omitted, PRO/II always generates electrolyte components automatically before the simulation run.

Once the components are generated, no changes are allowed in the component slate. Deleting an electrolyte model from the active thermodynamic sets invalidates all electrolyte components and displays an information dialog.

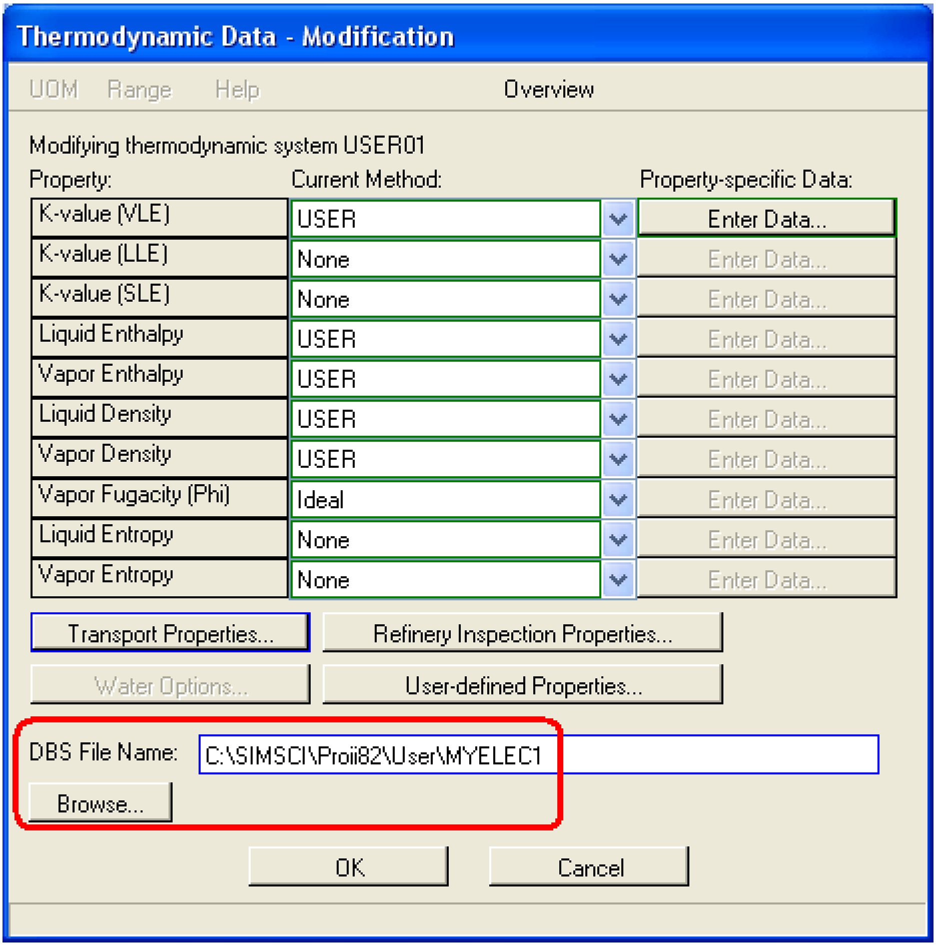
Adding Electrolyte Data Base Files To A Simulation

To add your own models, specifically suited to your application, use Chemistry Wizard 3.1 from OLI Systems, Inc. This generates data base (*.dbs*) files suitable for use by PRO/II. To use the data base file in a simulation:

- ➢Click the *Modify...* button shown at lower-left in Figure E1-3 to open the *Thermodynamic Data -Modification* data entry window.
- ➢As shown in Figure E1-5, the *DBS File Name:* field allows adding a data base file created with the *Chemistry Wizard*. The

associated *Browse...* button provides a convenient means of navigating to the data base file.

Figure E1-5: Adding an Electrolyte file to a Thermodynamic Method



Modifying thermodynamic system USER01

Property:	Current Method:	Property-specific Data:
K-value (VLE)	USER	Enter Data...
K-value (LLE)	None	Enter Data...
K-value (SLE)	None	Enter Data...
Liquid Enthalpy	USER	Enter Data...
Vapor Enthalpy	USER	Enter Data...
Liquid Density	USER	Enter Data...
Vapor Density	USER	Enter Data...
Vapor Fugacity (Phi)	Ideal	Enter Data...
Liquid Entropy	None	Enter Data...
Vapor Entropy	None	Enter Data...

DBS File Name: C:\SIMSCI\Proii82\User\MYELEC1

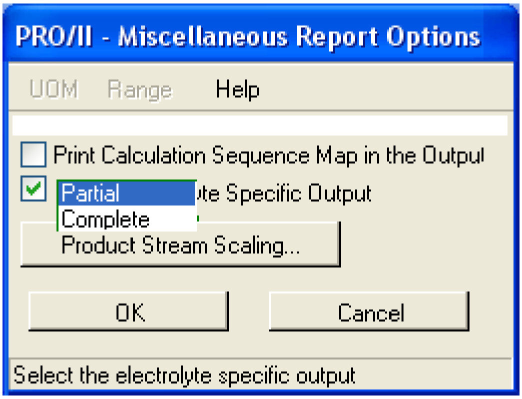
➢Click *OK* to save the selection and close the window.

Electrolyte Print Options Via the GUI

Calculated quantities unique to electrolyte applications can be printed in the output report of a simulation. From the GUI,

➢ Navigate to *Output menu -> Report Format -> Miscellaneous data*

Figure E1-6: Electrolyte PRINT ION Options



☐ Print Calculation Sequence Map in the Output

☑ Partial / Complete ...te Specific Output

Select the electrolyte specific output

➢Place a check mark in the check box to activate or change the ION report option. (The figure is a partial representation of *Miscellaneous Report Options* dialog.)

Keyword Access to Electrolytes

This section discusses the electrolytes-specific keywords in the order in which data sections appear in a keyword input file. All these options are analogous to the options described above for the Graphical User Interface.

Electrolyte Print Options Via Key Words

TITLE ...

PRINT ION= <u>PARTIAL</u> *or* **NONE** *or* **ALL**

In a keyword input file, use the ION keyword on the PRINT statement in the *General Data* category of input data. No other entries on the PRINT statement are unique to electrolytes.

The ION entry has three options: ION=NONE or PART or ALL. The default is PART.

NONE	No electrolyte-specific output is printed.
PART	Adds *Liquid Rates* table(s) to the *Stream Component Rates* section(s) of the output. This table gives the flow rates for ionic and/or other true species in the aqueous solution. For `ELDIST` columns only, this adds tables *Tray Compositions for True Solution Species*, *Tray Liquid pH and Ionic Strength*, and *Tray Component Scaling Indices*.

ALL	Gives all output given by `PART` option. Adds *Electrolyte Thermo Summary* section of the output. This section gives the equilibrium phases and phase compositions obtained from electrolyte flash calculations. Aqueous solution compositions are reported as molalities for the true chemical species. Also given are values for activity and fugacity coefficients as well as equilibrium constants. Adds *Thermo Notes* section to output. This section gives explanatory notes useful for the interpretation of the quantities given in the *Electrolyte Thermo Summary.*

Electrolyte Component Data

COMPONENT DATA ...
  LIBID i, compID {, libno, name / ...} &
    BANK=LibName:BankName {,...}

i	A unique, contiguous component order number. This applies to electrolyte components exactly the same as other library components. Refer to the *Component Data Keyword Manual*, Chapter 2, for a complete description.
compid	A unique text-string identifier associated with each component. Refer to the *Component Data Keyword Manual*, Chapter 2, for a complete description.
libno	Identical usage as with other library components. Refer to the *Component Data Keyword Manual*, Chapter 2, for a complete description.
name	Descriptive name. When supplied, this serves as an alias for the component ID in the output report. See example D below.
LibName	Name of a component library (discussed below).
BankName	Name of a data bank in the library (see below).

Components in OLI's databanks that are not included in the SIMSCI banks are available in the OLILIB bank. For electrolyte applications, the recommended bank search order is OLILIB first and SIMSCI second. This search order is used as the default whenever an electrolyte thermodynamic method is used.

- Use the BANK entry on the LIBID statement of Component Data to declare the component library search order. When using only banks form the default library, only the bank names are required on the BANK entry:

Example A: Minimum declaration of PROII Data banks only

```
LIBID 1, H2O / 2, NACL, BANK=OLILIB,SIMSCI {,...}
```

- The full form of *Bank= LibraryName*:*bankName* must be used to access banks from other libraries.

Example B: Full declaration of PROII data banks only

```
LIBID 1, H2O / 2, NACL,  &
        BANK=PROII_10.0:OLILIB,  PROII_10.0:SIMSCI {,...}
```

Adding user-added data banks to the search order requires using the full form. This form includes the library name and data bank name. The following example adds bank NEWELEC1 from library MYELEC as the first data bank in the search order, and with the PROII OLILIB as the second bank.

Example C: Declaration of user-added library and data bank

```
LIBID 1, H2O / 2, NACL,  &
        BANK=MYELEC:NEWELEC1,  PROII_10.0:OLILIB, ...
```

Component names for the electrolyte models are listed in the “Summary of Electrolyte Models” on page E2-12. The names in these lists are printed as they should be entered in the Component Data section on the `LIBID` statement. However, a Pure Component Library name, as given in the *SIMSCI Component and Thermodynamic Data Reference Manual*, Chapter 1.4, *Pure Component Library–Alpha Sort*, and Chapter 1.5, *Pure Component Library–Formula Sort*, can be used instead of a corresponding electrolyte model component name. For example, you could enter CAHYDROX on the LIBID statement in place of the name `CAOH2` when using the electrolyte model `SCRU`.

> ***Note:*** In contrast to conventional PRO/II, the Electrolytes version does *not* allow two or more different names for the same component to be used at the same time in the Component Data input.

Components can be renamed for printing purposes using the PRO/II *alias* approach.

Example D: Assigning an alias for output reporting

```
LIBID   1,NACL,, HALITE
```

causes the alias, HALITE, to be printed in the output in place of NACL, the OLI library name

Electrolyte Thermodynamic Data

```
THERMODYNAMIC DATA ...
      METHOD   SYSTEM = elecid,   {SETID = setid}
or
      METHOD  SYSTEM=DBSFIL,    {SETID = setid}
and          DBSFIL  FILE= drive:\path\filename.dbs
```

SYSTEM	Required entry to identify the electrolyte model to use for the thermodynamic METHOD.
elecid	A unique pre-defined identifier of a pregenerated electrolyte model. Also may be a predefined name of a version 6.6 user-added electrolyte model (i.e, a model with an assigned invariant name). Cannot be a user-generated name, as used for version 7.0 user-added models.
SETID	An optional identifier assigned to the METHOD set. Often convenient for referencing the set.
setid	.Text assigned as the identifier of the METHOD set. Required when the SETID keyword is present.
SYSTEM=DBSFIL	Required when using a DBSFIL statement. This replaces the actual identifier of an electrolyte model that has a user-assigned name.
DBSFIL	This statement is required when using the SYSTEM=DBSFIL entry on the METHOD statement.
FILE=	Required entry that locates and identifies the file that defines the electrolyte model to use.
drive:	Optional drive letter of the disk drive containing the electrolyte file.
\path	Optional path indicating the directory containing the electrolyte model. If omitted, PRO/II expects the model file to reside in the same directory as the input file or the simulation data base.
filename.dbs	Required. This identifies the file containing the electrolyte model.

Each electrolyte model in PRO/II is characterized by a fixed list of components, a fixed set of chemical equilibria, and a fixed set of thermodynamic methods for K-values, enthalpy, and density. The components and equilibria considered in the 40 pre-generated electrolyte models in PRO/II are summarized in Chapter E2, *Electrolyte Models.*

This section provides a summary of the THERMODYNAMIC DATA keyword input, and a set of application guidelines.

> ***Note:*** See *Thermodynamic Framework in PRO/II* in the *PRO/II Reference Manual, Vol. 1, Chapter 3* for a summary of changes to the thermodynamic framework and the mathematical model used by PRO/II Electrolytes for the calculation of thermodynamic quantities and chemical equilibrium, respectively.

The name of each pre-generated electrolyte model is treated in the input as a keyword by the SYSTEM entry on a thermodynamic METHOD statement.a system of thermodynamic methods. Refer to Table E2-2, “Electrolyte Model Keywords and Data Files,” on page E2-3 for a listing of all key words for all pre-generated electrolyte models.

Example:

```
METHOD  SYSTEM=SW01
```

where `SW01` represents the name of an electrolyte model.

When multiple thermodynamic methods are needed for an application, each of the selected methods should be assigned a method `SET` identifier.

Example:

```
METHOD   SYSTEM=SW01, SET=SET1
```

where `SET1` is the `SET` identifier that may be used to assign the thermodynamic METHOD set to individual unit operations and streams.

Using User-added Electrolyte Methods

To use a user-added model that does not have a pre-assigned name, the METHOD statement in the Thermodynamic section of keyword input must use the alternative syntax that requires a DBSFIL statement. For example, the following accesses a user-generated model named MyElec1.dbs in directory d:\MyStuff\ElecFiles\:

```
THERMODYNAMIC DATA
     METHOD  SYSTEM=DBSFIL,  SETID=MySet
          DBSFIL  FILE= D:\MyStuff\ElecFiles\MyElec1.dbs
```

This syntax also works for any pre-generated model installed in the \System\ directory of the PRO/II installation. For example, the mixed salt model named GENX70 resides in file M139.dbs. Assuming the installed directory is C:\SIMSCI\PROII82\SYSTEM\, the following keyword input accesses the model:

```
THERMODYNAMIC DATA
    METHOD  SYSTEM=DBSFIL,  SETID=GENX70
        DBSFIL FILE= C:\Program Files\SIMSCI\PROII82\SYS-
        TEM\M139.DBS
```

For more information, see “Assigning Names to User-Added Models” on page E2-8.

Vapor-Phase Fugacity Coefficients

Four options are available for calculating vapor-phase fugacities with PRO/II Electrolytes. These options, with their keywords, are:

- **Ideal-Gas (IDEAL):** all fugacity coefficients are assumed to be 1.0.
- **Nothnagel method (NOTH):** generally valid up to 20 atmospheres, but not available for some vapor-phase components.
- **Nakamura method (NAKA):** generally valid at higher pressures up to 200 atmospheres, but not available for some vapor-phase components.
- **Soave-Redlich Kwong (SRK) method:** valid over a wide range of conditions and generally recommended when vapor-phase non-ideality is important. Also valid for any vapor-phase chemical species given in OLI’s component databanks.

These options are specified on the `METHOD` statement with the PHI entry. If no vapor fugacity option is specified, the calculations default to ideal-gas.

Examples:

```
METHOD
SYSTEM=GENX, PHI=SRKMETHOD
SYSTEM=SW01, PHI=NOTH
```

> ***Note:*** Only `SYSTEM`, `SET`, `PHI`, `TRANSPORT`, `VISCOSITY`, `CONDUCTIVITY`, `SURFACE`, and `DIFFUSIVITY` statements (see *PRO/II Keyword Manual*, *Sec. 20*, *Thermodynamic Data*) are available for use with electrolyte thermodynamic methods. However, the PETRO transport property option cannot be used with electrolyte models, nor can the SIMSCI or API liquid-viscosity options. The `KVALUE`, `ENTHALPY`, and `DENSITY` methods for electrolytes are automatically set to the method declared by the `SYSTEM` entry and cannot be modified individually on the `METHOD` statement.

Use of ENTHALPY = ELEC for Mixed Thermodynamic Applications

The thermodynamic framework for PRO/II Electrolytes uses a different basis, or zero-point, for enthalpies from regular PRO/II. When both electrolyte and non-electrolyte thermodynamic methods are used in a flowsheet, PRO/II takes care of this basis change automatically if a stream is changed from one method to another.

However, to avoid the potentially confusing appearance of discontinuities in output values of enthalpy, it is recommended that the `ENTH=ELEC` option be used for all non-electrolyte thermodynamic methods in mixed electrolyte/non-electrolyte thermodynamic applications. `ENTH= ELEC` is PRO/II's IDEAL enthalpy method translated by a constant to give numbers consistent with the enthalpy basis for PRO/II Electrolytes.

Example:

```
METHOD   SYSTEM=SALT,   DEFAULT
METHOD   SYSTEM=IDEAL, ENTHALPY=ELEC, SET=SET2
METHOD   SYSTEM=SRK,    ENTHALPY=ELEC, SET=SET3
```

Note that `ENTH=ELEC` is used with both non-electrolyte thermodynamic methods, IDEAL and SRK. For these two methods, the calculated enthalpy values will be obtained using the IDEAL enthalpy method followed by a conversion to put these values on the electrolyte basis.

An example in which this method is useful is when steam is flashed off from an electrolyte unit (a stage in a multi-effect evaporator, for example) and then condensed to provide duty for a heat exchanger (the adjacent stage in the evaporator, for example). OLI's flash calculation has difficulty with adiabatic (fixed-duty) calculations for pure water in the two-phase region, with the result that the steam side of the heat exchanger might fail to solve with electrolyte thermodynamic methods. Instead, the user could specify the IDEAL thermodynamic method with `ENTH=ELEC` for the steam side of the exchanger to produce a converged solution with consistent enthalpies for all streams.

> ***Note:*** Whenever multiple enthalpy methods are used in a flowsheet, the safest (and recommended) practice for resetting the enthalpy basis is to take the stream that is changing methods through the reset unit that converts to the new enthalpy method. See the discussion in the *PRO/II Keyword Manual*, Section 20.5, *Multiple Thermodynamic Sets*.

Application Guidelines

General Information

PRO/II Electrolytes is composed of an integrated set of programs developed by SIMSCI and OLI Systems, Inc. that allows you to model and simulate aqueous electrolyte solutions. This includes, under the PRO/II flow sheeting environment, algorithms for the simulation of single-stage and multistage steady-state processes involving aqueous electrolytes. These programs calculate both heterogeneous, multi-phase equilibria as well as homogeneous liquid-phase reaction equilibria. The allowed phases at equilibrium include vapor, aqueous liquid, organic liquid, and solid phases. With respect to liquid-solid equilibria, saturation of multiple solid phases can be calculated rigorously.

PRO/II Electrolytes simulates aqueous systems by solving mathematical models that accurately represent these systems. These models are sets of nonlinear algebraic equations containing appropriate thermodynamic parameters. Two thermodynamic methods are available for electrolyte modeling, the OLI method and the NRTL (Chen) method. The Electrolyte NRTL method can be used for modeling very concentrated electrolyte solutions with very low water content (as low as 0.3 wt%) as well as solutions with mixed solvents ranging in composition from completely aqueous to almost completely non-aqueous (again, with water content as low as 0.3 wt%).

PRO/II Electrolytes is built upon a rigorous thermodynamic framework for aqueous electrolyte systems. This framework is an amalgam based upon work by Bromley (3), Meissner (4,5), Pitzer (6,7), Zemaitis (8,9), Nothnagel (10), Nakamura (11), and Tanger (12).

The PRO/II Electrolytes module incorporates electrolyte models created using standalone software from OLI Systems, Inc. The PRO/II Electrolytes module has 40 built-in electrolyte models that can be applied to a wide range of problems of industrial interest, including, but not limited to the following:

- Gas scrubbers/purification using carbonate, amine, or caustic solutions
- Sour water strippers
- Amine towers
- Benfield processes
- Acid waste stream-neutralization drums
- Scale formation prediction for wells and pipes (oilfields, geothermal fields)
- Underground injection well studies
- Solid salts manufacturing (potash, caustic, carbonates, etc.)
- Chlor-alkali plant processes.

The integration of electrolytes into PRO/II is, from the user’s point of view, seamless. In many cases, a keyword input file for an electrolytes problem could be identical (apart from entering the name of the electrolyte model in the Thermodynamic Data section) to a non-electrolyte PRO/II input file. The four most important things that PRO/II Electrolyte users must remember are:

- PRO/II Electrolytes is designed for use in aqueous electrolyte systems within the application guidelines given in Electrolyte Thermodynamic Data. PRO/II Electrolytes cannot currently be used for mixed-solvent or non-aqueous electrolyte systems, with the exception of the user-generated models created in the Electrolyte NRTL framework.

- The model chosen in the `THERMODYNAMIC DATA` keyword input (Chapter E2, *Electrolyte Models*) must contain all of the components appearing in any stream that is to be treated with that thermodynamic method.

- PRO/II Electrolytes activates one new unit operation: the distillation column algorithm ELDIST. See “Electrolyte Distillation Column Algorithm (ELDIST)” on page E4-5.

- A few units are not available, and some other units have reduced functionality when used with an electrolyte thermodynamic method. These differences are discussed fully in Chapter E4, *Electrolyte Effects In Simulations*.
- The Chemistry Wizard from OLI Systems, Inc. is installed with PRO/II Electrolytes. It allows users to generate their own new electrolyte models. See the discussion in Chapter E3, *Creating a User-Added Electrolyte Model*.

If these few differences are kept in mind, users familiar with standard PRO/II keyword input can quickly make effective use of the capabilities of the PRO/II Electrolytes module.

Thermodynamic Data Considerations

A first step in selecting an electrolyte model as a thermodynamic method is to determine whether the application of interest falls within the application guidelines for PRO/II Electrolytes. For example, the expected pressures, temperatures, phases, and phase compositions for the application should fall within the stated validity ranges for PRO/II Electrolytes. Next, the user needs to identify one of the PRO/II Electrolyte models that contains all of the components of interest in its component list (see Table E2-3, "Electrolyte Model Components," on page E2-9). However, for applications using electrolyte models together with non-electrolyte thermodynamic methods, the components to be treated by the non-electrolyte methods do not need to be in the model's component list if they will not enter any unit using the electrolyte model.

Pressure, Temperature, and Composition Validity Ranges

The suggested application range for electrolyte thermodynamic methods is summarized below.

General Ranges of Applicability

Temperature	32-572°F (0-300°C)
Pressure	0-1500 atm
Dissolved gases	0-30 mole%
Ionic solutes	0-30 ionic strength

Amine systems

Pressure 0-30 atm

LLE systems

Organic solutes 10 weight%

where the ionic strength (I) is defined as:

$$I = \frac{1}{2}\sum_{ions} Z^2_{ions}\, m_{ions} \qquad (0\text{-}7)$$

and *Z* denotes the ionic charge and *m* represents the molality of the aqueous ion. Molality is defined as the moles of solute per 1000 grams of H_2O solvent.

Invalid Calculations for Electrolyte Models

- Non-aqueous electrolyte systems. See the *Chemistry Wizard User Guide* for newer capabilities.
- Free water decant.
- Dew water calculations.
- Dew hydrocarbon calculations.
- Entropy and heat capacity calculations.

Special Case STREAM DATA Input

Due to the nature of the aqueous electrolyte flash algorithm, electrolyte feed streams with little or no water require special handling during input; you must specify the phase of the stream in the Stream Data section. Note that water-free liquid feed streams are not permitted. The algorithm also occasionally has difficulty converging for vapors or liquids at conditions very near the vapor-liquid phase boundary. Here again, if the user knows the phase of the stream, it can be entered on the `PROPERTY` statement; this will assure convergence but should be used with caution because the feed flash will not be able to *correct* an incorrect phase designation.

1)Feeds with little or no H_2O

Vapor feed	Enter PHASE=V on PROPERTY statement
Solid feed	Use SOLID card together with PROPERTY statement
Liquid feed	Not allowed. H_2O must be the predominant component in the liquid phase.

2)Feeds near saturation

At pressures and temperatures close to vapor+ liquid saturation conditions, it is recommended to enter the appropriate `PHASE` (V or L) on the `PROPERTY` statement.

True Aqueous Solution Species and Reconstituted Components

The true chemical species in the aqueous solution are those aqueous species that are included in the model's ionic equilibria and VLE expressions. Mass and charge balance constraints for the aqueous phase, together with the model's aqueous ionic equilibria expressions, are used by PRO/II Electrolytes to calculate apparent concentrations of the model's neutral components in the aqueous phase automatically from the corresponding concentrations of the true solution species (ionic and neutral). This calculation is referred to as the reconstitution of model components. Concentrations and rates are reported in the output for reconstituted components and, if `ION=PART` or `ALL` is specified on the `PRINT` statement, they are also reported for true solution species.

Example:

Calculate the results of an isothermal flash of the feed given below at 1 bar and 25°C, where the numbers before denote moles.

Model components:

```
H2O, NaOH, HCl, NaCl
```

Feed components:

```
55.51 H2O + 1 NaOH + 1 HCl
```

Product in terms of True Chemical Species:

$56.51\ H_2O + 1\ NA^{+1} + 1\ Cl^{-1} + 1.0E^{-7}\ OH^{-1} + 1.0E^{-7}\ H^{+1}$

Product in terms of Reconstituted Components:

$56.51\ H_2O + 1\ NaCl$

Examples

Example E1-1: Solid Component Declaration

Set up a pure HCl vapor feed and a pure NaOH solid feed at 25°C and 1 atm, together with a liquid H_2O feed at vapor-liquid saturation conditions of 100°C and 1 atm.

```
     . . .
COMPONENT DATA
     LIBID
      1, H2O/2, NAOH/3, HCL
THERMODYNAMIC DATA
     METHOD
     SYSTEM=GENE
STREAM DATA
     PROPERTY  STRM=HCLV, TEMP=25, PRES=1, &
               COMP=3, 0.5, PHASE=V
     PROPERTY  STRM=SOLD, TEMP=25, PRES=1
     SOLID STRM=SOLD, COMP=2,0.5
     PROPERTY STRM=WATR, TEMP=100, PRES=1, &
               COMP=1, 100, PHASE=L
```

References

1. OLI Systems, Inc., 1995, *Prochem User's Manuals*, Version 12, Morris Plains, NJ.
2. V.B. Shah, and R.W. Bondy, 1991, "A New Approach to Solving Electrolyte Distillation Problems", a paper presented at the AIChE Annual Meeting in Los Angeles.
3. L. A. Bromley, 1973, "Thermodynamic Properties of Strong Electrolytes in Aqueous Solutions", *AIChE J*, 19:313-320.
4. H. P. Meissner, and C. L. Kusik, 1973, "Aqueous Solutions of Two or More Strong Electrolytes - Vapor Pressures and Solubilities", *IEC Proc. Des. Dev.*, 12:205-208.
5. H. P. Meissner, and N. A. Peppas,1973, "Activity Coefficients - Aqueous Solutions of Polybasic Acids and their Salts", *AIChE J.*, 19:806-809.
6. K. S. Pitzer, 1979, "Theory: Ion Interaction Approach", *Activity Coefficients in Electrolyte Solutions*, 1:157-208, R. M. Pytkowicz, ed., CRC Press, Boca Raton, FL.
7. K. S. Pitzer, 1980, "Thermodynamics of Aqueous Electrolytes at Various Temperatures, Pressures and Compositions", *Thermodynamics of Aqueous Systems with Industrial Applications,* 451-466, S. A. Newman, ed., ACS Symposium Series 133.
8. J. F. Zemaitis, Jr., 1980, "Predicting Vapor-Liquid-Solid Equilibria in Multicomponent Aqueous Solutions of Electrolytes", *Thermodynamics of Aqueous Systems with Industrial Applica-*

tions, 227-246, S. A. Newman, ed., ACS Symposium Series 133.

9. J. F. Zemaitis, Jr., D. M. Clark, M. Rafal, and N. C. Scrivner, 1986, *Handbook of Aqueous Electrolyte Thermodynamics,* AIChE Press.
10. K. H. Nothnagel, D. S. Abrams, and J. M. Prausnitz, 1973, “Generalized Correlation of Fugacity Coefficients in Mixtures at Moderate Pressures”, *IEC Process Des. Dev.*, 12:25-35.
11. R. Nakamura, G. J. F. Breedveld, and J. M. Prausnitz, 1976, “Thermodynamic Properties of Gas Mixtures Containing Polar and Nonpolar Components”, *IEC Process Des. Dev.*, 15:557-564.
12. J.C. Tanger, IV, and H.C. Helgeson, 1988, “Calculation of the Thermodynamic and Transport Properties of Aqueous Species at High Pressures and Temperatures: Revised Equations of State for the Standard State Partial Molal Properties of Ions and Electrolytes”, *Am. J. Sci,* 288:19-98.

Chapter E2
Electrolyte Models

Overview

This chapter summarizes the forty pregenerated Electrolyte models included with the PRO/II Electrolyte Module. To facilitate application of these built-in models, the following tables are provided

- "Pre-generated Electrolyte Model Naming Conventions" on page E2-2 explains the data base file naming scheme.
- "Electrolyte Model Keywords and Data Files" on page E2-3 relates the electrolyte model name (used in the GUI) to comparable keywords for each pregenerated model.
- "Electrolyte Model Components" on page E2-9 exhaustively lists all the components available in each model.
- "Summary of Electrolyte Models" on page E2-12 lists all components, all ionic species, and all electrolyte equations in each model.
- "Alphabetical Component Index of Electrolyte Models" on page E2-40 shows every pregenerated model in which each component appears.

Pregenerated Models

Over the years, PRO/II has undergone many changes and has used several versions of the OLI Electrolyte Calculation Engine. Different versions of the OLI engine use different formats for the data base files. For this reason, the Electrolyte Module provides three versions of each of the 40 pregenerated electrolyte models. Naming conventions were chosen to help identify the original version used to create each data base file. The naming conventions are listed in Table E2-1.

Table E2-1: Pre-generated Electrolyte Model Naming Conventions

File Names in numeric order Keyword Suffix	PRO/II Version	OLI Engine Version	Creation Program
M01.dbs to **M40.dbs** no keyword suffix example: data file **M06.dbs** keywords: **DIPA** alias **M06**	**7.0 - 8.1**	6.3-**6.6**	Chemistry Wizard **1.0**
M41.dbs to **M60.dbs** (user-defined models) keywords **M41** to **M60** no keyword suffix and no alias example: data file: **M44.dbs** keyword: **M44** (no alias)	**7.0 - 8.1**	6.3-**6.6**	Chemistry Wizard **1.0**
M61.dbs to **M100.dbs** example: data file **M66** keywords **DIPA56** alias **M66**	prior to **7.0** (Obsolete in PRO/II 8.0)	**5.6** (Obsolete in PRO/II 8.0)	Electrolyte Utility Package (**EUP**)
M101.dbs to **M140.dbs** example: data file **M06.dbs** no keyword suffix keywords: **DIPA70** alias **M106**	**8.2**	**7.0**	Chemistry Wizard **2.0**

With the naming conventions disclosed, Table E2-2 lists all the keywords for all three versions of each of the 40 pregenerated models. The table also lists the keywords for the older user-added models that required pre-assigned file names.

Table E2-2: Electrolyte Model Keywords and Data Files

Electrolyte System	Electrolyte Model	Keywords	Data File
Amines	DEA56	DEA56, M62	M62.DBS
	DEA66	DEA, M02	M02.DBS
	DEA70	DEA70, M102	M102.DBS
	DGA56	DGA56, M64	M64.DBS
	DGA66	DGA, M04	M04.DBS
	DGA70	DGA70, M104	M104.DBS
	DIPA56	DIPA56, M66	M66.DBS
	DIPA66	DIPA, M06	M06.DBS
	DIPA70	DIPA70, M106	M106.DBS
	MEA56	MEA56, M61	M61.DBS
	MEA66	MEA, M01	M01.DBS
	MEA70	MEA70, M101	M101.DBS
	MDEA56	MDEA56, M65	M65.DBS
	MDEA66	MDEA, M05	M05.DBS
	MDEA70	MDEA70, M105	M105.DBS
	TEA56	TEA56, M63	M63.DBS
	TEA66	TEA, M03	M03.DBS
	TEA70	TEA70, M103	M103.DBS
Acids	ACIF56	ACIF56, M82	M82.DBS
	ACID66	ACID, M22	M22.DBS
	ACID70	ACID70, M122	M122.DBS
	CL256	CL256, M71	M71.DBS
	CL266	CL2, M11	M11.DBS
	CL270	CL270, M111	M111.DBS
	CLSF56	CLSF56, M81	M81.DBS
	CLSF66	CLSF, M21	M21.DBS
	CLSF70	CLSF70, M121	M121.DBS
	HCL56	HCL56, M67	M67.DBS
	HCL66	HCL, M07	M07.DBS
	HCL70	HCL70, M107	M107.DBS

Table E2-2: Electrolyte Model Keywords and Data Files

Electrolyte System	Electrolyte Model	Keywords	Data File
Mixed Salts	PHOS56	PHOS56, M69	M69.DBS
	PHOS66	PHOS, M09	M09.DBS
	PHOS70	PHOS70, M109	M109.DBS
	SULF56	SULF56, M68	M68.DBS
	SULF66	SULF, M08	M08.DBS
	SULF70	SULF70, M108	M108.DBS
	CANA56	CANA56, M72	M72.DBS
	CANA66	CANA, M12	M12.DBS
	CANA70	CANA70, M112	M112.DBS
	CANX56	CANX56, M90	M90.DBS
	CANX66	CANX, M30	M30.DBS
	CANX70	CANX70, M130	M130.DBS
	GENE56	GENE56, M86	M86.DBS
	GENE66	GENE, M26	M26.DBS
	GENE70	GENE70, M126	M126.DBS
	GENX56	GENX56, M99	M99.DBS
	GENX66	GENX, M39	M39.DBS
	GENX70	GENX70, M139	M139.DBS
	GEOT56	GEOT56, M79	M79.DBS
	GEOT66	GEOT, M19	M19.DBS
	GEOT70	GEOT70, M119	M119.DBS
	HOTC56	HOTC56, M85	M85.DBS
	HOTC66	HOTC, M25	M25.DBS
	HOTC70	HOTC70, M125	M125.DBS
	HOTX56	HOTX56, M98	M98.DBS
	HOTX66	HOTX, M38	M38.DBS
	HOTX70	HOTX70, M138	M138.DBS
	OILF56	OIL56, M80	M80.DBS
	OILF66	OILF, M20	M20.DBS
	OILF70	OILF70, M120	M120.DBS
	SALT56	SALT56, M70	M70.DBS
	SALT66	SALT, M10	M10.DBS
	SALT70	SALT70, M110	M110.DBS

Table E2-2: Electrolyte Model Keywords and Data Files

Electrolyte System	Electrolyte Model	Keywords	Data File
Sour Water	SW0156	SW0156, M74	M74.DBS
	SW0166	SW01, M14	M14.DBS
	SW0170	SW0170, M114	M114.DBS
	SW1X56	SW1X56, M92	M92.DBS
	SW1X66	SW1X, M32	M32.DBS
	SW1X70	SW1X70, M132	M132.DBS
	SW0256	SW0256, M75	M75.DBS
	SWO266	SW02, M15	M15.DBS
	SW0270	SW0270, M115	M115.DBS
	SW2X56	SW2X56, M93	M93.DBS
	SW2X66	SW2X, M33	M33.DBS
	SW2X70	SW2X70, M133	M133.DBS
	SW0356	SW0356, M76	M76.DBS
	SW0366	SW03, M16	M16.DBS
	SW0370	SW0370, M116	M116.DBS
	SW3X56	SW3X56, M94	M94.DBS
	SW3X66	SW3X, M34	M34.DBS
	SW3X70	SW3X70, M134	M134.DBS
	SW0456	SW0456, M77	M77.DBS
	SW0466	SW04, M17	M17,DBS
	SW0470	SW0470, M117	M117.DBS
	SW4X56	SW4X56, M95	M95.DBS
	SW4X66	SW4X, M35	M35.DBS
	SW4X70	SW4X70, M135	M135.DBS
	SW0556	SW0556, M78	M78.DBS
	SW0566	SW05, M18	M18.DBS
	SW0570	SW0570, M118	M118.DBS
Caustic Systems	CAUS56	CAUS56, M83	M83.DBS
	CAUS66	CAUS, M23	M23.DBS
	CAUS70	CAUS70, M123	M123.DBS
	CAUX56	CAUX56, M96	M96.DBS
	CAUX66	CAUX, M36	M36.DBS
	CAUX70	CAUX70, M136	M136.DBS

Table E2-2: Electrolyte Model Keywords and Data Files

Electrolyte System	Electrolyte Model	Keywords	Data File
	CAU256	CAU256, M87	M87.DBS
	CAU266	CAU2, M27	M27.DBS
	CAU270	CAU270, M127	M127.DBS
	CA2X56	CA2X56, M100	M100.DBS
	CA2X66	CA2X, M40	M40.DBS
	CA2X70	CA2X70, M140	M140.DBS
Benfield Systems	BENF56	BENF56, M84	M84.DBS
	BENF66	BENF, M24	M24.DBS
	BENF70	BENF70, M124	M124.DBS
	BENX56	BENX56, M97	M97.DBS
	BENX66	BENX, M37	M37.DBS
	BENX70	BENX70, M137	M137.DBS
Scrubber Systems	SCRU56	SCRU56, M73	M73.DBS
	SCRU66	SCRU, M13	M13.DBS
	SCRU70	SCRU70, M113	M113.DBS
	SCRX56	SCRX56, M91	M91.DBS
	SCRX66	SCRX, M31	M31.DBS
	SCRX70	SCRX70, M131	M131.DBS
LLE & Hydrate Systems	TWL156	TWL156, M88	M88.DBS
	TWL166	TWL1, M28	M28.DBS
	TWL170	TWL170, M128	M128.DBS
	TWL256	TWL256, M89	M89.DBS
	TWL266	TWL2, M29	M29.DBS
	TWL270	TWL270, M129	M129.DBS
User-Added OLI 6.6 from Chemistry Wizard 1.0	M41	M41	M41.DBS
	M42	M42	M42.DBS
	M43	M43	M43.DBS
	M44	M44	M44.DBS
	M45	M45	M45.DBS
	M46	M46	M46.DBS
	M47	M47	M47.DBS
	M48	M48	M48.DBS
	M49	M49	M49.DBS

Table E2-2: Electrolyte Model Keywords and Data Files

Electrolyte System	Electrolyte Model	Keywords	Data File
	M50	M50	M50.DBS
	M51	M51	M51.DBS
	M52	M52	M52.DBS
	M53	M53	M53.DBS
	M54	M54	M54.DBS
	M55	M55	M55.DBS
	M56	M56	M56.DBS
	M57	M57	M57.DBS
	M58	M58	M58.DBS
	M59	M59	M59.DBS
	M60	M60	M60.DBS
User-Added from OLI 7.0 from Chemistry Wizard 2.0	Names are assigned as users create the .DBS files. Usage is : METHOD SYSTEM=**DBSFILE** DBSFILE FILE= {*Drive:\path*}***filename.dbs***		

Choosing An Appropriate Electrolyte Model

The electrolyte model chosen must contain all components appearing in any stream or unit using that method. To reduce CPU time and eliminate extraneous components from the printout, it is generally best to use the smallest model encompassing the flowsheet's components. Similarly, a model without solids will run faster than a model with the same components that does allow solid precipitation. Non-solids-forming models should be used only for flow sheets where the user is certain precipitation will not occur. Models with solids suppressed are designated by names ending in `X`; for example, `CAUX` is the CAUS model without solid precipitation.

While all the pregenerated models perform properly, it is recommended that users migrate to the newer "7.0" versions of the data files (data base files M101 through M140) at their earliest convenience.

User-Added Models

If none of the 40 pregenerated models is suitable for an application, user-added, electrolyte models should be investigated. OLI Systems, Inc., in partnership with Invensys Operations Management, provide the *OLI Chemistry Wizard version 3.1* that allows generation of user-added models that can be used in the PRO/II Electrolytes program.

- Complete documentation of the new tool set is provided in the *OLI Chemistry Wizard User Guide*.
- The basic electrolyte components in each model are listed in Table E2-3 on page E2-9. A complete list of all species in each model is available in the separate document *PRO/II-OLI 6.6_Components.xls*, a Microsoft Excel spreadsheet.

Both documents are included on the PRO/II installation disk. During installation, they may be copied to the \Manual\OLI subdirectory of the PRO/II installation directory.

Assigning Names to User-Added Models

Note that the Chemistry Wizard does not use any assigned or reserved names for the data base files it creates. This is in contrast to earlier methods that required data base files to use one of the reserved names (M41 through M60).

The naming rules are as follows:

- Names must conform to the DOS 8.3 model.
- The first character of the name should be a letter (A-Z)
- Names are NOT case sensitive; e.g., “A” and “a” alias each other.
- Names may include embedded blanks, but their use is discouraged.

Additionally, alternative keyword syntax allows electrolyte model files to reside in any directory accessible by PRO/II. In the past, files normally were required to reside in the same directory as the .INP or .PRZ file, or in the PRO/II \System\ directory. The newer DBSFIL statement allows entering the path along with the file name. As with the file name, the path may contain embedded spaces, and is not case sensitive.

To use a user-added model that does not have a pre-assigned name, the METHOD statement in the Thermodynamic section of keyword input must use the alternative syntax that requires a DBSFIL statement. For example, the following accesses a user-generated model named MyElec1.dbs in directory d:\MyStuff\ElecFiles\:

```
THERMODYNAMIC DATA
METHOD  SYSTEM=DBSFIL,  SETID=MySet
  DBSFIL  FILE= D:\MyStuff\ElecFiles\MyElec1.dbs
```

This syntax also works for any pregenerated model installed in the \System\ directory. For example, the mixed salt model named `GENX70` resides in file M139.dbs. Assuming the installed directory is C:\Program Files\SIMSCI\PROII82\SYSTEM\, the following keyword input accesses the model:

```
THERMODYNAMIC DATA
METHOD  SYSTEM=DBSFIL,  SETID=GENX70
  DBSFIL FILE= C:\Program Files\SIM-
  SCI\PROII82\SYSTEM\M139.DBS
```

Electrolyte Model Index

Model names (MEA, DEA, etc.) are grouped by systems of general chemical or application interest.

Table E2-3: Electrolyte Model Components

Model	Components	Page
Amine Systems		E2-12
MEA	H_2O / CO_2 / C_2H_6 / CH_4 / H_2S / N_2 / C_4H_{10} / C_3H_8 / MEAH	E2-12
DEA	H_2O / CO_2 / C_2H_6 / CH_4 / H_2S / N_2 / C_4H_{10} / C_3H_8 / DEAH	E2-13
TEA	H_2O / CO_2 / C_2H_6 / CH_4 / H_2S / N_2 / C_4H_{10} / C_3H_8 / TEAH	E2-13
DGA	H_2O / CO_2 / C_2H_6 / CH_4 / H_2S / N_2 / C_4H_{10} / C_3H_8 / DGAH	E2-13
MDEA	H_2O / CO_2 / C_2H_6 / CH_4 / H_2S / N_2 / C_4H_{10} / C_3H_8 / MDEAH	E2-13

Table E2-3: Electrolyte Model Components

Model	Components	Page
DIPA	H_2O / CO_2 / C_2H_6 / CH_4 / H_2S / N_2 / C_4H_{10} / C_3H_8 / DIPAx (x = 1, 2, 3, 4, 5, 6, 7, CO_2 or H_2)	E2-13
Acid Systems		E2-13
ACID	H_2O / CO / CO_2 / H_2 / HCN / HCOOH / N_2 / NH_3 / H_3PO_4	E2-13
HCL	H_2O / HCl	E2-14
CL2	H_2O / HCl / Cl_2 / HClO / N_2	E2-14
CLSF	H_2O / HCl / Cl_2 / HClO / N_2 / O_2 / SO_2 / H_2SO_3 / H_2SO_4	E2-15
SULF	H_2O / SO_2 / H_2SO_3 / H_2SO_4	E2-15
PHOS	H_2O / H_3PO_4	E2-16
Mixed Salt Systems		E2-17
SALT	H_2O / NaCl / KCl	E2-17
CANA, CANX	H_2O / HCl / $MgSO_4$ / Na_2SO_4 / NaCl / H_2SO_4 / $CaSO_4$ / NaOH / $CaCl_2$ / $MgCl_2$ / $Ca(OH)_2$ / $Mg(OH)_2$	E2-17
HOTC, HOTX	H_2O / CO_2 / CO / C_2H_6 / CH_4 / H_2S / H_2 / N_2 / C_3H_8 / $NaHCO_3$ / Na_2CO_3 / NaOH	E2-18
GENE, GENX	H_2O / CO_2 / Cl_2 / HClO / HCl / N_2 / O_2 / SO_2 / NaOH / NaCl / $NaHCO_3$ / Na_2CO_3	E2-19
GEOT	H_2O / CO_2 / NH_3 / H_2S / HCl / $BaCl_2$ / $CaCl_2$ / $CuCl_2$ / $FeIICl_2$ / $FeIIICl_3$ / KCl /LiCl / $MgCl_2$ / $MnCl_2$ / NaCl / Na_2S / Na_2SO_3 / Na_2SO_4 / $PbCl_2$ / $SrCl_2$ / $ZnCl_2$ / Na_4EDTA / Ca_2EDTA / $B(OH)_3$ / $BaCO_3$ / $BaSO_4$ / $CaCO_3$ / $CaSO_4$ / Fe_3O_4 / $MgCO_3$ / $(NH_4)_2SO_4$ / $SrCO_3$ / $SrSO_4$	E2-20
OILF	H_2O / CO_2 / HCl / Na_2SO_4 / $NaHCO_3$ / Na_2CO_3 / $MgCl_2$ / $BaCl_2$ / $CaCl_2$ / NaCl / NaOH / KCl / $BaSO_4$ / $CaCO_3$ / $CaSO_4$ / $BaCO_3$ / $Ca(OH)_2$ / K_2CO_3 / K_2SO_4 / $KHCO_3$ / $MgCO_3$ / $Mg(OH)_2$ / $BaOH_2$ / KOH / $MgSO_4$ / H_2SO_4	E2-23
Sour Water Systems		E2-25

Table E2-3: Electrolyte Model Components

Model	Components	Page
SW01, SW1X	H_2O / CO_2 / H_2S / NH_3 / CH_4 / C_6H_5OH / NaOH / $NaHCO_3$ / Na_2CO_3 / C_2H_6 / C_3H_8 / N_2 / HCl / HCN / H_3PO_4 / NaCl / NaHS / HCOOH	E2-25
SW02, SW2X	H_2O / CO_2 / H_2S / NH_3 / CH_4 / C_6H_5OH / NaOH / $NaHCO_3$ / Na_2CO_3 / HCl / NaCl / NaHS / HCN / ACETACID	E2-27
SW03, SW3X	H_2O / CO_2 / H_2S / NH_3 / CH_4 / C_6H_5OH / NaOH / $NaHCO_3$ / Na_2CO_3 / C_2H_6 / C_3H_8 / N_2 / HCl / HCN / C_4H_{10} / NaCl / NaHS / H_2 / O_2 / CO	E2-28
SW04, SW4X	H_2O / CO_2 / H_2S / NH_3 / CH_4 / C_6H_5OH / NaOH / $NaHCO_3$ / Na_2CO_3 / C_2H_6 / C_3H_8 / N_2 / C_4H_{10}	E2-29
SW05	H_2O / CO_2 / H_2S / NH_3	E2-30
Caustic Systems		E2-31
CAUS, CAUX	H_2O / Cl_2 / HClO / HCl / SO_2 / NaCl / Na_2SO_3 / Na_2SO_4 / NaOH / H_2SO_4	E2-31
CAU2, CA2X	H_2O / NaOH / KOH	E2-32
Benfield Systems		E2-33
BENF, BENX	H_2O / CO_2 / CO / C_2H_6 / C_2H_4 / CH_4 / H_2S / H_2 / N_2 / NH_3 / C_3H_8 / K_2CO_3 / $KHCO_3$ / KHS / H_3PO_4 / $B(OH)_3$ / KOH	E2-33
Scrubber Systems		E2-34
SCRU, SCRX	H_2O / CO_2 / HCl / HCN / HCOOH / N_2 / O_2 / SO_2 / NaOH / $CaCO_3$ / $CaCl_2$ / NaCl / Na_2CO_3 / $NaHCO_3$ / NaCOOH / $Ca(OH)_2$	E2-34
LLE and Hydrate Systems		E2-36
TWL1	H_2O / CO_2 / H_2S / NH_3 / CH_4 / C_6H_5OH / Toluene / NaOH / $NaOH.H_2O$ / Na_2CO_3 / $Na_2CO_3\cdot 10H_2O$ / $Na_2CO_3\cdot H_2O$/ $Na_2CO_3\cdot 7H_2O$ / $NaHCO_3$ / NaCl / NaHS / HCl / $CaCl_2$ / $CaCl_2\cdot H_2O$ / $CaCl_2\cdot 2H_2O$ / $CaCl_2\cdot 4H_2O$ / $CaCl_2\cdot 6H_2O$ / $Ca(HCO_3)_2$ / $CaCO_3$ / $Ca(OH)_2$ / NH_4Cl / NH_4HCO_3 / NH_4HS	E2-36

Table E2-3: Electrolyte Model Components

Model	Components	Page
TWL2	H_2O / NaCl / $NaHSO_4$ / Na_2SO_4 / $Na_2SO_4 \cdot 10H_2O$ / NaOH / $NaOH \cdot H_2O$ / $Ca(HSO_4)_2$ / $CaSO_4$ / $CaSO_4 \cdot 2H_2O$ / $Ca(OH)_2$ / HCl / N_2 / CH_4 / Methanol / Ethanol / M-Xylene / Benzene / Toluene / $CaCl_2$ / $CaCl_2 \cdot H_2O$ / $CaCl_2 \cdot 2H_2O$ / $CaCl_2 \cdot 4H_2O$ / $CaCl_2 \cdot 6H_2O$ / HF / H_2SO_4 / CaF_2 / NaF	E2-38

Summary of Electrolyte Models

All ionic equilibria given in the model summaries below take place in aqueous solution. The `".ppt"` suffix on a chemical species name denotes a solid precipitate, and the `"aq"` suffix stands for a dissolved gas or an electrically neutral ion pair. The general form of a vapor-liquid equilibrium expression is `Cvap = Caq`, where `C` denotes a model component.

Amine Systems

MEA Model

Components:

H_2O / CO_2 / C_2H_6 / CH_4 / H_2S / N_2 / C_3H_8 / C_4H_{10} / MEAH

Ions:

OH^{-1} / HCO_3^{-1} / H^{+1} / HS^{-1} / MEA-1 / MEAH2+1 / CO_3^{-2} / S^{-2}

Ionic Equilibria:

$CO_2aq + H_2O = H^{+1} + HCO_3^{-1}$

$H_2O = H^{+1} + OH^{-1}$

$H_2Saq = H^{+1} + HS^{-1}$

$HCO_3^{-1} = H^{+1} + CO_3^{-2}$

$HS^{-1} = H^{+1} + S^{-2}$

$MEACO_2\text{-}1 + H_2O = MEAHaq + HCO_3^{-1}$

$MEAHaq + H_2O = MEAH2+1 + OH^{-1}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is considered for all components.

DEA, TEA, DGA, MDEA, and DIPA Models

Component lists for the other amine models differ from the MEA model only with respect to their amine components; that is, DEAH, TEAH, DGAH, MDEAH, or DIPAx replaces MEAH. In addition, these other amine models do not consider vapor-liquid equilibrium for their amine component. Finally, there is no $MDEACO_2$-1 ion in the MDEA model.

Acid Systems

ACID Model

Components:

H_2O / CO / CO_2 / H_2 / HCN / HCOOH / N_2 / NH_3 / H_3PO_4

Ions:

OH^{-1} / CO_3^{-2} / $COOH^{-1}$ / $H_2P_2O_7^{-2}$ / $H_2PO_4^{-1}$ / $H_3P_2O_7^{-1}$ / HCO_3^{-1} /

H^{+1} / $HP_2O_7^{-3}$ / HPO_4^{-2} / $NH_2CO_2^{-1}$ / NH_4^{+1} / CN^{-1} / $P_2O_7^{-4}$ / PO_4^{-3}

Ionic Equilibria:

$CO_2aq + H_2O = HCO_3^{-1} + H^{+1}$

$H_2O = H^{+1} + OH^{-1}$

$H_2P_2O_7^{-2} = H^{+1} + HP_2O_7^{-3}$

$H_2PO_4^{-1} = H^{+1} + HPO_4^{-2}$

$H_3P_2O_7^{-1} = H^{+1} + H_2P_2O_7^{-2}$

$HCNaq = H^{+1} + CN^{-1}$

$HCO_3^{-1} = H^{+1} + CO_3^{-2}$

$HCOOHaq = H^{+1} + COOH^{-1}$

$HP_2O_7^{-3} = H^{+1} + P_2O_7^{-4}$

$HPO_4^{-2} = H^{+1} + PO_4^{-3}$

$NH_2CO_2^{-1} + H_2O = NH_3aq + HCO_3^{-1}$

$NH_3aq + H_2O = NH_4^{+1} + OH^{-1}$

$P_2O_7^{-4} + H_2O = 2PO_4^{-3} + 2H^{+1}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is not considered for H_3PO_4.

HCL Model

Components:

H_2O / HCl

Ions:

OH^{-1} / H^{+1} / Cl^{-1}

Ionic Equilibria:

$H_2O = H^{+1} + OH^{-1}$

$HClaq = H^{+1} + Cl^{-1}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is considered for all components.

CL2 Model

Components:

H_2O / Cl_2 / HClO / HCl / N_2

Ions:

OH^{-1} / ClO^{-1} / H^{+1} / Cl^{-1}

Ionic Equilibria:

$Cl_2aq + H_2O = H^{+1} + Cl^{-1} + HClOaq$

$H_2O = H^{+1} + OH^{-1}$

$HClaq = H^{+1} + Cl^{-1}$

$HClOaq = H^{+1} + ClO^{-1}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is considered for all components.

CLSF Model

Components:

H_2O / Cl_2 / $HClO$ / HCl / N_2 / O_2 / SO_2 / H_2SO_3 / H_2SO_4

Ions:

OH^{-1} / ClO^{-1} / H^{+1} / HSO_3^{-1} / HSO_4^{-1} / Cl^{-1} / $S_2O_5^{-2}$ / SO_3^{-2} / SO_4^{-2}

Ionic Equilibria:

$Cl_2aq + H_2O = H^{+1} + Cl^{-1} + HClOaq$

$H_2O = H^{+1} + OH^{-1}$

$HClaq = H^{+1} + Cl^{-1}$

$HClOaq = H^{+1} + ClO^{-1}$

$HSO_3^{-1} = H^{+1} + SO_3^{-2}$

$HSO_4^{-1} = H^{+1} + SO_4^{-2}$

$S_2O_5^{-2} + H_2O = 2SO_3^{-2} + 2H^{+1}$

$SO_2aq + H_2O = HSO_3^{-1} + H^{+1}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is not considered for H_2SO_3 / H_2SO_4.

SULF Model

Components:

H_2O / SO_2 / H_2SO_3 / H_2SO_4

Ions:

OH^{-1} / HSO_3^{-1} / HSO_4^{-1} / H^{+1} / $S_2O_5^{-2}$ /

SO_3^{-2} / SO_4^{-2}

Ionic Equilibria:

$H_2O = H^{+1} + OH^{-1}$

$HSO_3^{-1} = H^{+1} + SO_3^{-2}$

$HSO_4^{-1} = H^{+1} + SO_4^{-2}$

$S_2O_5^{-2} + H_2O = 2SO_3^{-2} + 2H^{+1}$

$SO_2aq + H_2O = HSO_3^{-1} + H^{+1}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is considered for H_2O / SO_2.

PHOS Model

Components:

H_2O / H_3PO_4 / PO_4

Ions:

OH^{-1} / $H_2PO_4^{-1}$ / $H_3P_2O_7^{-1}$ / H^{+1} / $HP_2O_7^{-3}$ / HPO_4^{-2} / $H_2P_2O_7^{-2}$ /

$P_2O_7^{-4}$ / PO_4^{-3}

Ionic Equilibria:

$H_2O = H^{+1} + OH^{-1}$

$H_2P_2O_7^{-2} = H^{+1} + HP_2O_7^{-3}$

$H_2PO_4^{-1} = H^{+1} + HPO_4^{-2}$

$H_3P_2O_7^{-1} = H^{+1} + H_2P_2O_7^{-2}$

$HP_2O_7^{-3} = H^{+1} + P_2O_7^{-4}$

$HPO_4^{-2} = H^{+1} + PO_4^{-3}$

$P_2O_7^{-4} + H_2O = 2PO_4^{-3} + 2H^{+1}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is considered for H_2O only.

Mixed Salt Systems

SALT Model

Components:

H_2O / NaCl / KCl

Ions:

OH^{-1} / H^{+1} / K^{+1} / Na^{+1} / Cl^{-1}

Precipitates and hydrates:

NaCl / KCl

Ionic Equilibria:

$H_2O = H^{+1} + OH^{-1}$

$KClppt = K^{+1} + Cl^{-1}$

$NaClppt = Na^{+1} + Cl^{-1}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is considered for H_2O only.

CANA Model

Components:

H_2O / HCl / $MgSO_4$ / Na_2SO_4 / NaCl / H_2SO_4 / $CaSO_4$ / NaOH /

$CaCl_2$ / $MgCl_2$ / $Ca(OH)_2$ / $Mg(OH)_2$

Ions:

OH^{-1} / $CaOH^{+1}$ / Cl^{-1} / H^{+1} / HSO_4^{-1} / $MgCl^{+1}$ / Mg^{+2} / Na^{+1} /

$NaSO_4^{-1}$ / Ca^{+2} / SO_4^{-2}

Precipitates and hydrates:

$MgSO_4$ / Na_2SO_4 / NaCl / $CaSO_4$ / NaOH / $CaCl_2$ / $MgCl_2$ /

$Ca(OH)_2$ / $Mg(OH)_2$

Ionic Equilibria:

$CaCl_2ppt = Ca^{+2} + 2Cl^{-1}$

$Ca(OH)_2ppt = Ca^{+2} + 2OH^{-1}$

$CaOH^{+1} = Ca^{+2} + OH^{-1}$

$CaSO_4aq = Ca^{+2} + SO_4^{-2}$

$CaSO_4ppt = Ca^{+2} + SO_4^{-2}$

$H_2O = H^{+1} + OH^{-1}$

$HClaq = H^{+1} + Cl^{-1}$

$HSO_4^{-1} = H^{+1} + SO_4^{-2}$

$MgCl_2ppt = Mg^{+2} + 2Cl^{-1}$

$MgCl^{+1} = Mg^{+2} + Cl^{-1}$

$Mg(OH)_2ppt = Mg^{+2} + 2OH^{-1}$

$MgSO_4ppt = Mg^{+2} + SO_4^{-2}$

$Na_2SO_4ppt = 2Na^{+1} + SO_4^{-2}$

$NaClppt = Na^{+1} + Cl^{-1}$

$NaOHppt = Na^{+1} + OH^{-1}$

$NaSO_4^{-1} = Na^{+1} + SO_4^{-2}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is considered for H_2O / HCl.

CANX Model

Identical to CANA model except that solid precipitation equilibria are not considered.

HOTC Model

Components:

H_2O / CO_2 / CO / C_2H_6 / CH_4 / H_2S / H_2 / N_2 / C_3H_8 / $NaHCO_3$ /

Na_2CO_3 / NaOH

Ions:

OH^{-1} / HCO_3^{-1} / H^{+1} / HS^{-1} / Na^{+1} / CO_3^{-2} / S^{-2}

Precipitates and Hydrates:

$NaHCO_3$ / Na_2CO_3 / $NaOH$

Ionic Equilibria:

$CO_2aq + H_2O = HCO_3^{-1} + H^{+1}$

$H_2O = H^{+1} + OH^{-1}$

$H_2Saq = H^{+1} + HS^{-1}$

$HCO_3^{-1} = H^{+1} + CO_3^{-2}$

$HS^{-1} = H^{+1} + S^{-2}$

$Na_2CO_3ppt = 2Na^{+1} + CO_3^{-2}$

$NaHCO_3ppt = Na^{+1} + HCO_3^{-1}$

$NaOHppt = Na^{+1} + OH^{-1}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is not considered for $NaHCO_3$ / Na_2CO_3 / NaOH.

HOTX Model

Identical to HOTC model except that solid precipitation equilibria are not considered.

GENE Model

Components:

H_2O / CO_2 / Cl_2 / HClO / HCl / N_2 / O_2 / SO_2 / NaOH / NaCl /

$NaHCO_3$ / Na_2CO_3

Ions:

OH^{-1} / ClO^{-1} / CO_3^{-2} / HCO_3^{-1} / H^{+1} / HSO_3^{-1} / Na^{+1} / Cl^{-1} /

$S_2O_5^{-2}$ / SO_3^{-2}

Precipitates and Hydrates:

$NaOH$ / $NaCl$ / $NaHCO_3$ / Na_2CO_3

Ionic Equilibria:

$Cl_2aq + H_2O = H^{+1} + Cl^{-1} + HClOaq$

$CO_2aq + H_2O = HCO_3^{-1} + H^{+1}$

$H_2O = H^{+1} + OH^{-1}$

$HClaq = H^{+1} + Cl^{-1}$

$HClOaq = H^{+1} + ClO^{-1}$

$HCO_3^{-1} = H^{+1} + CO_3^{-2}$

$HSO_3^{-1} = H^{+1} + SO_3^{-2}$

$Na_2CO_3ppt = 2Na^{+1} + CO_3^{-2}$

$NaClppt = Na^{+1} + Cl^{-1}$

$NaHCO_3ppt = Na^{+1} + HCO_3^{-1}$

$NaOHppt = Na^{+1} + OH^{-1}$

$S_2O_5^{-2} + H_2O = 2SO_3^{-2} + 2H^{+1}$

$SO_2aq + H_2O = HSO_3^{-1} + H^{+1}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is not considered for NaOH / NaCl / $NaHCO_3$ / Na_2CO_3.

GENX Model

Identical to GENE model except that solid precipitation equilibria are not considered.

GEOT Model

Components:

H_2O / CO_2 / NH_3 / H_2S / HCl / $BaCl_2$ / $CaCl_2$ / $CuCl_2$ / $FeIICl_2$ /

$FeIIICl_3$ / KCl / $LiCl$ / $MgCl_2$ / $MnCl_2$ / $NaCl$ / Na_2S / Na_2SO_3 /

Na_2SO_4 / $PbCl_2$ / $SrCl_2$ / $ZnCl_2$ / Na_4EDTA / Ca_2EDTA / $B(OH)_3$ /

$BaCO_3$ / $BaSO_4$ / $CaCO_3$ / $CaSO_4$ / Fe_3O_4 / $MgCO_3$ / $(NH_4)_2SO_4$ /

$SrCO_3$ / $SrSO_4$

Ions:

OH^{-1} / NH_4^{+1} / Ba^{+2} / $BaOH^{+1}$ / Ca^{+2} / $CaHCO_3^{+1}$ / $CaOH^{+1}$ /

Cu^{+2}/$FeII^{+2}$ / $FeIIHCO_3^{+1}$ / $FeIIOH^{+1}$ / $FeIII^{+3}$ / K^{+1} / LI+1 / Mg^{+2} /

$MgCl^{+1}$ / $MgHCO_3^{+1}$ / Mn^{+2} / $MnHCO_3^{+1}$ / $MnOH^{+1}$ / Na^{+1} / Pb^{+2} /

Sr^{+2} / Zn^{+2} / $ZnHCO_3^{+1}$ / H^{+1} / HCO_3^{-1} / CO_3^{-2} / $NH_2CO_2^{-1}$ / HS^{-1} /

S^{-2} / Cl^{-1} / $NaSO_4^{-1}$ / HSO_3^{-1} / SO_3^{-2} / HSO_4^{-1} / SO_4^{-2} / $EDTA^{-4}$ /

$B(OH)_3OH^{-1}$

Precipitates and Hydrates:

$BaCO_3$ / $BaSO_4$ / $CaCO_3$ / Fe_3O_4 / $MgCO_3$ / $(NH_4)_2SO_4$ /

$SrCO_3$ / $SrSO_4$

Ionic Equilibria:

$H_2O = H^{+1} + OH^{-1}$

$CO_2aq + H_2O = H^{+1} + HCO_3^{-1}$

$NH_3aq + H_2O = NH_4^{+1} + OH^{-1}$

$H_2Saq = H^{+1} + HS^{-1}$

$CaSO_4aq = Ca^{+2} + SO_4^{-2}$

$Na_4EDTAaq = 4Na^{+1} + EDTA^{-4}$

$Ca_2EDTAaq = 2Ca^{+2} + EDTA^{-4}$

$B(OH)_3aq + H_2O = B(OH)_3OH^{-1} + H^{+1}$

$HCO_3^{-1} = H^{+1} + CO_3^{-2}$

$NH_2CO_2^{-1} + H_2O = NH_3aq + HCO_3^{-1}$

$HS^{-1} = H^{+1} + S^{-2}$

$BaOH^{+1} = Ba^{+2} + OH^{-1}$

$CaHCO_3^{+1} = Ca^{+2} + HCO_3^{-1}$

$CaOH^{+1} = Ca^{+2} + OH^{-1}$

$FeIIHCO_3^{+1} = FeII^{+2} + HCO_3^{-1}$

$FeIIOH^{+1} = FeII^{+2} + OH^{-1}$

$HSO_3^{-1} = H^{+1} + SO_3^{-2}$

$HSO_4^{-1} = H^{+1} + SO_4^{-2}$

$MgCl^{+1} = Mg^{+2} + Cl^{-1}$

$MgHCO_3^{+1} = Mg^{+2} + HCO_3^{-1}$

$MnHCO_3^{+1} = Mn^{+2} + HCO_3^{-1}$

$MnOH^{+1} = Mn^{+2} + OH^{-1}$

$NaSO_4^{-1} = Na^{+1} + SO_4^{-2}$

$ZnHCO_3^{+1} = Zn^{+2} + HCO_3^{-1}$

$BaCO_3ppt = Ba^{+2} + CO_3^{-2}$

$BaSO_4ppt = Ba^{+2} + SO_4^{-2}$

$CaCO_3ppt = Ca^{+2} + CO_3^{-2}$

$Fe_3O_4ppt + 8H^{+1} = FeII^{+2} + 2FeIII^{+3} + 4H_2O$

$MgCO_3ppt = Mg^{+2} + CO_3^{-2}$

$(NH_4)_2SO_4ppt = 2NH_4^{+1} + SO_4^{-2}$

$SrCO_3ppt = Sr^{+2} + CO_3^{-2}$

$SrSO_4ppt = Sr^{+2} + SO_4^{-2}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is considered for H_2O / CO_2 / NH_3 / H_2S.

OILF Model

Components:

H_2O / CO_2 / HCl / Na_2SO_4 / $NaHCO_3$ / Na_2CO_3 / $MgCl_2$ / $BaCl_2$ /

$CaCl_2$ / NaCl / NaOH / KCl / $BaSO_4$ / $CaCO_3$ / $CaSO_4$ / $BaCO_3$ /

$Ca(OH)_2$ / K_2CO_3 / K_2SO_4 / $KHCO_3$ / $MgCO_3$ / $Mg(OH)_2$ / $BaOH_2$ /

KOH / $MgSO_4$ / H_2SO_4

Ions:

OH^{-1} / $BaOH^{+1}$ / $CaHCO_3^{+1}$ / Ca^{+2} / $CaOH^{+1}$ / Cl^{-1} / CO_3^{-2} /

HCO_3^{-1} / H^{+1} / HSO_4^{-1} / K^{+1} / $MgCl^{+1}$ / $MgHCO_3^{+1}$ / Mg^{+2} /

Na^{+1} / $NaSO_4^{-1}$ / Ba^{+2} / SO_4^{-2}

Precipitates and Hydrates:

Na_2SO_4 / $NaHCO_3$ / Na_2CO_3 / $MgCl_2$ / $BaCl_2$ / $CaCl_2$ / NaOH /

KCl / $BaSO_4$ / $CaCO_3$ / $CaSO_4$ / $BaCO_3$ / $Ca(OH)_2$ / K_2CO_3 /

K_2SO_4 / $KHCO_3$ / $MgCO_3$ / $Mg(OH)_2$ / $BaOH_2$ / KOH / $MgSO_4$

Ionic Equilibria:

$BaCl_2ppt = Ba^{+2} + 2Cl^{-1}$

$BaCO_3ppt = Ba^{+2} + CO_3^{-2}$

$BaOH_2ppt = Ba^{+2} + 2OH^{-1}$

$BaOH^{+1} = Ba^{+2} + OH^{-1}$

$BaSO_4ppt = Ba^{+2} + SO_4^{-2}$

$CaCl_2ppt = Ca^{+2} + 2Cl^{-1}$

$CaCO_3ppt = Ca^{+2} + CO_3^{-2}$

$CaHCO_3^{+1} = Ca^{+2} + HCO_3^{-1}$

$Ca(OH)_2ppt = Ca^{+2} + 2OH^{-1}$

$CaOH^{+1} = Ca^{+2} + OH^{-1}$

$CaSO_4aq = Ca^{+2} + SO_4^{-2}$

$CaSO_4ppt = Ca^{+2} + SO_4^{-2}$

$CO_2aq + H_2O = HCO_3^{-1} + H^{+1}$

$H_2O = H^{+1} + OH^{-1}$

$HClaq = H^{+1} + Cl^{-1}$

$HCO_3^{-1} = H^{+1} + CO_3^{-2}$

$HSO_4^{-1} = H^{+1} + SO_4^{-2}$

$K_2CO_3ppt = 2K^{+1} + CO_3^{-2}$

$K_2SO_4ppt = 2K^{+1} + SO_4^{-2}$

$KClppt = K^{+1} + Cl^{-1}$

$KHCO_3ppt = K^{+1} + HCO_3^{-1}$

$KOHppt = K^{+1} + OH^{-1}$

$MgCl_2ppt = Mg^{+2} + 2Cl^{-1}$

$MgCl^{+1} = Mg^{+2} + Cl^{-1}$

$MgCO_3ppt = Mg^{+2} + CO_3^{-2}$

$MgHCO_3^{+1} = Mg^{+2} + HCO_3^{-1}$

$Mg(OH)_2ppt = Mg^{+2} + 2OH^{-1}$

$MgSO_4ppt = Mg^{+2} + SO_4^{-2}$

$Na_2CO_3ppt = 2Na^{+1} + CO_3^{-2}$

$Na_2SO_4ppt = 2Na^{+1} + SO_4^{-2}$

$NaHCO_3ppt = Na^{+1} + HCO_3^{-1}$

$NaOHppt = Na^{+1} + OH^{-1}$

$NaSO_4^{-1} = Na^{+1} + SO_4^{-2}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is considered for H_2O / CO_2 / HCl.

Sour Water Systems

SW01 Model

Components:

H_2O / CO_2 / H_2S / NH_3 / CH_4 / C_6H_5OH / NaOH / $NaHCO_3$ / Na_2CO_3 /

C_2H_6 / C_3H_8 / N_2 / HCl / HCN / H_3PO_4 / NaCl / NaHS / HCOOH

Ions:

OH^{-1} / Cl^{-1} / CN^{-1} / CO_3^{-2} / $COOH^{-1}$ / $H_2P_2O_7^{-2}$ / $H_2PO_4^{-1}$ / $H_3P_2O_7^{-1}$ /

HCO_3^{-1} / H^{+1} / $HP_2O_7^{-3}$ / HPO_4^{-2} / HS^{-1} / Na^{+1} / $NH_2CO_2^{-1}$ / NH_4^{+1} /

$C_6H_5O^{-1}$ / $P_2O_7^{-4}$ / PO_4^{-3} / S^{-2}

Precipitates and Hydrates:

$NaHCO_3$ / Na_2CO_3 / NaOH / NaCl

Ionic Equilibria:

$C_6H_5OHaq = H^{+1} + C_6H_5O^{-1}$

$CO_2aq + H_2O = HCO_3^{-1} + H^{+1}$

$H_2O = H^{+1} + OH^{-1}$

$H_2P_2O_7^{-2} = H^{+1} + HP_2O_7^{-3}$

$H_2PO_4^{-1} = H^{+1} + HPO_4^{-2}$

$H_2Saq = H^{+1} + HS^{-1}$

$H_3P_2O_7^{-1} = H^{+1} + H_2P_2O_7^{-2}$

$HClaq = H^{+1} + Cl^{-1}$

$HCNaq = H^{+1} + CN^{-1}$

$HCO_3^{-1} = H^{+1} + CO_3^{-2}$

$HCOOHaq = H^{+1} + COOH^{-1}$

$HP_2O_7^{-3} = H^{+1} + P_2O_7^{-4}$

$HPO_4^{-2} = H^{+1} + PO_4^{-3}$

$HS^{-1} = H^{+1} + S^{-2}$

$Na_2CO_3ppt = 2Na^{+1} + CO_3^{-2}$

$NaClppt = Na^{+1} + Cl^{-1}$

$NaHCO_3ppt = Na^{+1} + HCO_3^{-1}$

$NaOHppt = Na^{+1} + OH^{-1}$

$NH_2CO_2^{-1} + H_2O = NH_3aq + HCO_3^{-1}$

$NH_3aq + H_2O = NH_4^{+1} + OH^{-1}$

$P_2O_7^{-4} + H_2O = 2PO_4^{-3} + 2H^{+1}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is not considered for H_3PO_4 / $NaHCO_3$ / Na_2CO_3 / NaOH / NaCl / NaHS.

SW1X Model

Identical to SW01 model except that solid precipitation equilibria are not considered.

SW02 Model

Components:

H_2O / CO_2 / H_2S / NH_3 / CH_4 / C_6H_5OH / NaOH / $NaHCO_3$ /

Na_2CO_3 / HCl / NaCl / NaHS / HCN / ACETACID

Ions:

OH^{-1} / $C_6H_5O^{-1}$ / Cl^{-1} / CN^{-1} / CO_3^{-2} / HCO_3^{-1} / H^{+1} / HS^{-1} /

Na_2CO_3 / $NH_2CO_2^{-1}$ / NH_4^{+1} / $Acetate^{-1}$ / S^{-2}

Precipitates and Hydrates:

$NaHCO_3$ / NaOH / Na_2CO_3 / NaCl

Ionic Equilibria:

$C_6H_5OHaq = H^{+1} + C_6H_5O^{-1}$

$CO_2aq + H_2O = HCO_3^{-1} + H^{+1}$

$H_2O = H^{+1} + OH^{-1}$

$H_2Saq = H^{+1} + HS^{-1}$

$ACETACIDaq = H^{+1} + Acetate^{-1}$

$HClaq = H^{+1} + Cl^{-1}$

$HCNaq = H^{+1} + CN^{-1}$

$HCO_3^{-1} = H^{+1} + CO_3^{-2}$

$HS^{-1} = H^{+1} + S^{-2}$

$Na_2CO_3ppt = 2Na^{+1} + CO_3^{-2}$

$NaClppt = Na^{+1} + Cl^{-1}$

$NaHCO_3ppt = Na^{+1} + HCO_3^{-1}$

$NaOHppt = Na^{+1} + OH^{-1}$

$NH_2CO_2^{-1} + H_2O = NH_3aq + HCO_3^{-1}$

$NH_3aq + H_2O = NH_4^{+1} + OH^{-1}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is not considered for Acetic Acid / $NaHCO_3$ / NaOH / Na_2CO_3 / NaHS / NaCl.

SW2X Model

Identical to SW02 model except that solid precipitation equilibria are not considered.

SW03 Model

Components:

H_2O / CO_2 / H_2S / NH_3 / CH_4 / C_6H_5OH / NaOH / $NaHCO_3$ /

Na_2CO_3 /C_2H_6 /C_3H_8 / N_2 / HCl / HCN / C_4H_{10} / NaCl / NaHS /

H_2 / O_2 / CO

Ions:

OH^{-1} / Cl^{-1} / CN^{-1} / CO_3^{-2} / HCO_3^{-1} / H^{+1} / HS^{-1} / Na^{+1} / NH_4^{+1} /

$NH_2CO_2^{-1}$ / $C_6H_5O^{-1}$ / S^{-2}

Precipitates and Hydrates:

Na_2CO_3 / NaOH / $NaHCO_3$ / NaCl

Ionic Equilibria:

$C_6H_5OHaq = H^{+1} + C_6H_5O^{-1}$

$CO_2aq + H_2O = HCO_3^{-1} + H^{+1}$

$H_2O = H^{+1} + OH^{-1}$

$H_2Saq = H^{+1} + HS^{-1}$

$HClaq = H^{+1} + Cl^{-1}$

$HCNaq = H^{+1} + CN^{-1}$

$HCO_3^{-1} = H^{+1} + CO_3^{-2}$

$HS^{-1} = H^{+1} + S^{-2}$

$Na_2CO_3ppt = 2Na^{+1} + CO_3^{-2}$

$NaClppt = Na^{+1} + Cl^{-1}$

$NaHCO_3ppt = Na^{+1} + HCO_3^{-1}$

$NaOHppt = Na^{+1} + OH^{-1}$

$NH_2CO_2^{-1} + H_2O = NH_3aq + HCO_3^{-1}$

$NH_3aq + H_2O = NH_4^{+1} + OH^{-1}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is not considered for Na_2CO_3 / NaOH / $NaHCO_3$ / NaHS / NaCl.

SW3X Model

Identical to SW03 model except that solid precipitation equilibria are not considered.

SW04 Model

Components:

H_2O / CO_2 / H_2S / NH_3 / CH_4 / C_6H_5OH / NaOH / $NaHCO_3$ /

Na_2CO_3 / C_2H_6 / C_3H_8 / N_2 / C_4H_{10}

Ions:

OH^{-1} / CO_3^{-2} / HCO_3^{-1} / H^{+1} / HS^{-1} / Na^{+1} / $NH_2CO_2^{-1}$ /
NH_4^{+1} /

$C_6H_5O^{-1}$ / S^{-2}

Precipitates and Hydrates:

NaOH / $NaHCO_3$ / Na_2CO_3

Ionic Equilibria:

- $C_6H_5OHaq = H^{+1} + C_6H_5O^{-1}$
- $CO_2aq + H_2O = HCO_3^{-1} + H^{+1}$
- $H_2O = H^{+1} + OH^{-1}$
- $H_2Saq = H^{+1} + HS^{-1}$
- $HCO_3^{-1} = H^{+1} + CO_3^{-2}$
- $HS^{-1} = H^{+1} + S^{-2}$
- $Na_2CO_3ppt = 2Na^{+1} + CO_3^{-2}$
- $NaHCO_3ppt = Na^{+1} + HCO_3^{-1}$
- $NaOHppt = Na^{+1} + OH^{-1}$
- $NH_2CO_2^{-1} + H_2O = NH_3aq + HCO_3^{-1}$
- $NH_3aq + H_2O = NH_4^{+1} + OH^{-1}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is not considered for NaOH / $NaHCO_3$ / Na_2CO_3.

SW4X Model

Identical to SW04 model except that solid precipitation equilibria are not considered.

SW05 Model

Components:

H_2O / CO_2 / H_2S / NH_3

Ions:

OH^{-1} / HCO_3^{-1} / H^{+1} / HS^{-1} / $NH_2CO_2^{-1}$ / NH_4^{+1} / CO_3^{-2} / S^{-2}

Ionic Equilibria:

$CO_2aq + H_2O = HCO_3^{-1} + H^{+1}$

$H_2O = H^{+1} + OH^{-1}$

$H_2Saq = H^{+1} + HS^{-1}$

$HCO_3^{-1} = H^{+1} + CO_3^{-2}$

$HS^{-1} = H^{+1} + S^{-2}$

$NH_2CO_2^{-1} + H_2O = NH_3aq + HCO_3^{-1}$

$NH_3aq + H_2O = NH_4^{+1} + OH^{-1}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is considered for all components.

Caustic Systems

CAUS Model

Components:

H_2O / Cl_2 / HClO / HCl / SO_2 / NaCl / Na_2SO_3 / Na_2SO_4 /

NaOH / H_2SO_4

Ions:

OH^{-1} / ClO^{-1} / H^{+1} / HSO_3^{-1} / HSO_4^{-1} / Na^{+1} / $NaSO_4^{-1}$ / Cl^{-1} /

$S_2O_5^{-2}$ / SO_3^{-2} / SO_4^{-2}

Precipitates and Hydrates:

NaCl / Na_2SO_3 / Na_2SO_4 / NaOH

Ionic Equilibria:

$Cl_2aq + H_2O = H^{+1} + Cl^{-1} + HClOaq$

$H_2O = H^{+1} + OH^{-1}$

$HClaq = H^{+1} + Cl^{-1}$

$HClOaq = H^{+1} + ClO^{-1}$

$HSO_3^{-1} = H^{+1} + SO_3^{-2}$

$HSO_4^{-1} = H^{+1} + SO_4^{-2}$

$Na_2SO_3ppt = 2Na^{+1} + SO_3^{-2}$

$Na_2SO_4ppt = 2Na^{+1} + SO_4^{-2}$

$NaClppt = Na^{+1} + Cl^{-1}$

$NaOHppt = Na^{+1} + OH^{-1}$

$NaSO_4^{-1} = Na^{+1} + SO_4^{-2}$

$S_2O_5^{-2} + H_2O = 2SO_3^{-2} + 2H^{+1}$

$SO_2aq + H_2O = HSO_3^{-1} + H^{+1}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is considered for H_2O / Cl_2 / HClO / HCl / SO_2.

CAUX Model

Identical to CAUS model except that solid precipitation equilibria are not considered.

CAU2 Model

Components:

H_2O / NaOH / KOH

Ions:

OH^{-1} / K^{+1} / Na^{+1} / H^{+1}

Precipitates and Hydrates:

NaOH / KOH

Ionic Equilibria:

$H_2O = H^{+1} + OH^{-1}$

$KOHppt = K^{+1} + OH^{-1}$

$NaOHppt = Na^{+1} + OH^{-1}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is considered for H_2O only.

CA2X Model

Identical to CAU2 model except that solid precipitation equilibria are not considered.

Benfield Systems

BENF Model

Components:

H_2O / CO_2 / CO / C_2H_6 / C_2H_4 / CH_4 / H_2S / H_2 / N_2 / NH_3 / C_3H_8 /

K_2CO_3 / $KHCO_3$ / KHS / H_3PO_4 / $B(OH)_3$ / KOH

Ions:

OH^{-1} / CO_3^{-2} / $H_2P_2O_7^{-2}$ / $H_2PO_4^{-1}$ / $H_3P_2O_7^{-1}$ / HCO_3^{-1} / H^{+1} /

$HP_2O_7^{-3}$ / HPO_4^{-2} / HS^{-1} / K^{+1} / $NH_2CO_2^{-1}$ / NH_4^{+1} / BO_2^{-1} /

$P_2O_7^{-4}$ / PO_4^{-3} / S^{-2}

Precipitates and Hydrates:

K_2CO_3 / $KHCO_3$ / KOH

Ionic Equilibria:

$CO_2aq + H_2O = HCO_3^{-1} + H^{+1}$

$H_2O = H^{+1} + OH^{-1}$

$H_2P_2O_7^{-2} = H^{+1} + HP_2O_7^{-3}$

$H_2PO_4^{-1} = H^{+1} + HPO_4^{-2}$

$H_2Saq = H^{+1} + HS^{-1}$

$H_3P_2O_7^{-1} = H^{+1} + H_2P_2O_7^{-2}$

$HCO_3^{-1} = H^{+1} + CO_3^{-2}$

$P2O7-3 = H^{+1} + P_2O_7^{-4}$

$HPO_4^{-2} = H^{+1} + PO_4^{-3}$

$HS^{-1} = H^{+1} + S^{-2}$

$K_2CO_3ppt = 2K^{+1} + CO_3^{-2}$

$KHCO_3ppt = K^{+1} + HCO_3^{-1}$

$KOHppt = K^{+1} + OH^{-1}$

$NH_2CO_2^{-1} + H_2O = NH_3aq + HCO_3^{-1}$

$NH_3aq + H_2O = NH_4^{+1} + OH^{-1}$

$P_2O_7^{-4} + H_2O = 2PO_4^{-3} + 2H^{+1}$

$B(OH)_3aq = BO_2^{-1} + H_2O + H^{+1}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is not considered for K_2CO_3 / $KHCO_3$ / KHS / H_3PO_4 / $B(OH)_3$ / KOH.

BENX Model

Identical to BENF model except that solid precipitation equilibria are not considered.

Scrubber Systems

SCRU Model

Components:

H_2O / CO_2 / HCl / HCN / HCOOH / N_2 / O_2 / SO_2 / NaOH / $CaCO_3$ /

$CaCl_2$ / NaCl / Na_2CO_3 / $NaHCO_3$ / NaCOOH / $Ca(OH)_2$

Ions:

OH^{-1} / $CaHCO_3^{+1}$ / Ca^{+2} / $CaOH^{+1}$ / Cl^{-1} / CN^{-1} / CO_3^{-2} /

$COOH^{-1}$ / HCO_3^{-1} / H^{+1} / HSO_3^{-1} / Na^{+1} / $CaCOOH^{+1}$ / $S_2O_5^{-2}$ / SO_3^{-2}

Precipitates and Hydrates:

NaOH / $CaCO_3$ / $CaCl_2$ / NaCl / Na_2CO_3 / $NaHCO_3$ /

NaCOOH / $Ca(OH)_2$

Ionic Equilibria:

$CaCl_2ppt = Ca^{+2} + 2Cl^{-1}$

$CaCO_3ppt = Ca^{+2} + CO_3^{-2}$

$CaCOOH^{+1} = Ca^{+2} + COOH^{-1}$

$CaHCO_3^{+1} = Ca^{+2} + HCO_3^{-1}$

$Ca(OH)_2ppt = Ca^{+2} + 2OH^{-1}$

$CaOH^{+1} = Ca^{+2} + OH^{-1}$

$CO_2aq + H_2O = HCO_3^{-1} + H^{+1}$

$H_2O = H^{+1} + OH^{-1}$

$HClaq = H^{+1} + Cl^{-1}$

$HCNaq = H^{+1} + CN^{-1}$

$HCO_3^{-1} = H^{+1} + CO_3^{-2}$

$HCOOHaq = H^{+1} + COOH^{-1}$

$HSO_3^{-1} = H^{+1} + SO_3^{-2}$

$Na_2CO_3ppt = 2Na^{+1} + CO_3^{-2}$

$NaClppt = Na^{+1} + Cl^{-1}$

$NaCOOHppt = Na^{+1} + COOH^{-1}$

$NaHCO_3ppt = Na^{+1} + HCO_3^{-1}$

$NaOHppt = Na^{+1} + OH^{-1}$

$S_2O_5^{-2} + H_2O = 2SO_3^{-2} + 2H^{+1}$

$SO_2aq + H_2O = HSO_3^{-1} + H^{+1}$

Vapor-Liquid Equilibria:

Vapor-liquid equilibrium is considered for H_2O / CO_2 / HCl / HCN / HCOOH / N_2 / O_2 / SO_2.

LLE and Hydrate Systems

TWL1 Model

Components:

H_2O / CO_2 / H_2S / NH_3 / CH_4 / C_6H_5OH /Toluene / NaOH /

$NaOH{\cdot}H_2O$ / Na_2CO_3 / $Na_2CO_3{\cdot}10H_2O$ / $Na_2CO_3{\cdot}H_2O$ /

$Na_2CO_3{\cdot}7H_2O$ / $NaHCO_3$ / NaCl /NaHS / HCl / $CaCl_2$ / $CaCl_2{\cdot}H_2O$ /

$CaCl_2{\cdot}2H_2O$ / $CaCl_2{\cdot}4H_2O$ / $CaCl_2{\cdot}6H_2O$ /$Ca(HCO_3)_2$ / $CaCO_3$ /

$Ca(OH)_2$ / NH_4Cl / NH_4HCO_3 / NH_4HS

Ions:

OH^{-1} / HCO_3^{-1} / $NaCO_3^{-1}$ / $NH_2CO_2^{-1}$ / Cl^{-1} /HS^{-1} / $C_6H_5O^{-1}$ /

CO_3^{-2} / S^{-2} / $CaCl^{+1}$ / $CaHCO_3^{+1}$ / $CaOH^{+1}$ / H^{+1} /

Na^{+1} / NH_4^{+1} / Ca^{+2}

Precipitates and Hydrates:

$CaCl_2$ / $CaCO_3$ / $Ca(OH)_2$ / Na_2CO_3 /NaCl / $NaHCO_3$ / NaOH / NH_4Cl /

NH_4HCO_3 / NH_4HS / $CaCl_2{\cdot}H_2O$ / $CaCl_2{\cdot}2H_2O$ / $CaCl_2{\cdot}4H_2O$ /

$CaCl_2{\cdot}6H_2O$ / $Na_2CO_3{\cdot}10H_2O$ / $Na_2CO_3{\cdot}H_2O$ / $Na_2CO_3{\cdot}7H_2O$ /

$NaOH{\cdot}H_2O$

Ionic Equilibria:

$C_6H_5OHaq = H^{+1} + C_6H_5O^{-1}$

$CaCl_2{\cdot}H_2O = Ca^{+2} + 2Cl^{-1} + H_2O$

$CaCl_2{\cdot}2H_2O = Ca^{+2} + 2Cl^{-1} + 2H_2O$

$CaCl_2{\cdot}4H_2O = Ca^{+2} + 2Cl^{-1} + 4H_2O$

$CaCl_2{\cdot}6H_2O = Ca^{+2} + 2Cl^{-1} + 6H_2O$

$CaCl_2aq = CaCl^{+1} + Cl^{-1}$

$CaCl_2ppt = Ca^{+2} + 2Cl^{-1}$

$CaCl^{+1} = Ca^{+2} + Cl^{-1}$

$CaCO_3aq = Ca^{+2} + CO_3^{-2}$

$CaCO_3ppt = Ca^{+2} + CO_3^{-2}$

$CaHCO_3^{+1} = Ca^{+2} + HCO_3^{-1}$

$Ca(OH)_2ppt = Ca^{+2} + 2OH-$

$CaOH^{+1} = Ca^{+2} + OH^{-1}$

$CO_2aq + H_2O = H^{+1} + HCO_3^{-1}$

$H_2O = H^{+1} + OH^{-1}$

$H_2Saq = H^{+1} + HS^{-1}$

$HClaq = H^{+1} + Cl^{-1}$

$HCO_3^{-1} = H^{+1} + CO_3^{-2}$

$HS^{-1} = H^{+1} + S^{-2}$

$Na_2CO_3 \cdot 10H_2O = 2Na^{+1} + CO_3^{-2} + 10H_2O$

$Na_2CO_3 \cdot H_2O = 2Na^{+1} + CO_3^{-2} + H_2O$

$Na_2CO_3 \cdot 7H_2O = 2Na^{+1} + CO_3^{-2} + 7H_2O$

$Na_2CO_3ppt = 2Na^{+1} + CO_3^{-2}$

$NaClppt = Na^{+1} + Cl^{-1}$

$NaCO_3^{-1} = Na^{+1} + CO_3^{-2}$

$NaHCO_3aq = Na^{+1} + HCO_3^{-1}$

$NaHCO_3ppt = Na^{+1} + HCO_3^{-1}$

$NaOH \cdot H_2O = Na^{+1} + OH^{-1} + 1H_2O$

$NaOHppt = Na^{+1} + OH^{-1}$

$NH_2CO_2^{-1} + H_2O = NH_3aq + HCO_3^{-1}$

$NH_3aq + H_2O = NH_4^{+1} + OH^{-1}$

$NH_4Clppt = NH_4^{+1} + Cl^{-1}$

$NH_4HCO_3ppt = NH_4^{+1} + HCO_3^{-1}$

$NH_4HSppt = NH_4^{+1} + HS^{-1}$

Vapor-Liquid and Liquid-Liquid Equilibria:

Vapor-liquid equilibrium is considered for H_2O / C_6H_5OH / CH_4 / CO_2 / H_2S / HCl / NH_3 / Toluene.

TWL2 Model

Components:

H_2O / NaCl / $NaHSO_4$ / Na_2SO_4 / $Na_2SO_4 \cdot 10H_2O$ /NaOH /

$NaOH \cdot H_2O$ / $Ca(HSO_4)_2$ / $CaSO_4$ / $CaSO_4 \cdot 2H_2O$ / $Ca(OH)_2$ /

HCl / N_2 / CH_4 / Methanol / Ethanol / M-Xylene / Benzene /

Toluene / $CaCl_2$ / $CaCl_2 \cdot H_2O$ / $CaCl_2 \cdot 2H_2O$ / $CaCl_2 \cdot 4H_2O$ /

$CaCl_2 \cdot 6H_2O$ / HF / H_2SO_4 / CaF_2 / NaF

Ions:

OH^{-1} / Cl^{-1} / HF_2^{-1} / HSO_4^{-1} / $NaSO_4^{-1}$ / F^{-1} / SO_4^{-2} / H^{+1} /

$CaCl^{+1}$ / CaF^{+1} / $CaOH^{+1}$ / Na^{+1} / Ca^{+2}

Precipitates and Hydrates:

$CaCl_2$ / CaF_2 / $Ca(OH)_2$ / $CaSO_4$ / Na_2SO_4 / NaCl / NaF / $NaHSO_4$/

NaOH / $CaCl_2 \cdot H_2O$ / $CaCl_2 \cdot 2H_2O$ / $CaCl_2 \cdot 4H_2O$ / $CaCl_2 \cdot 6H_2O$ /

$CaSO_4 \cdot 2H_2O$ / $Na_2SO_4 \cdot 10H_2O$ / $NaOH \cdot H_2O$

Ionic Equilibria:

$CaCl_2 \cdot H_2O = Ca^{+2} + 2Cl^{-1} + H_2O$

$CaCl_2 \cdot 2H_2O = Ca^{+2} + 2Cl^{-1} + 2H_2O$

$CaCl_2 \cdot 4H_2O = Ca^{+2} + 2Cl^{-1} + 4H_2O$

$CaCl_2 \cdot 6H_2O = Ca^{+2} + 2Cl^{-1} + 6H_2O$

$CaCl_2aq = CaCl^{+1} + Cl^{-1}$

$CaCl_2ppt = Ca^{+2} + 2Cl^{-1}$

$CaCl^{+1} = Ca^{+2} + Cl^{-1}$

$CaF_2ppt = Ca^{+2} + 2F^{-1}$

$CaF^{+1} = Ca^{+2} + F^{-1}$

$Ca(OH)_2ppt = Ca^{+2} + 2OH^{-1}$

$CaOH^{+1} = Ca^{+2} + OH^{-1}$

$CaSO_4 \cdot 2H_2O = Ca^{+2} + SO_4^{-2} + 2H_2O$

$CaSO_4aq = Ca^{+2} + SO_4^{-2}$

$CaSO_4ppt = Ca^{+2} + SO_4^{-2}$

$H_2O = H+ + OH^{-1}$

$HClaq = H+ + Cl^{-1}$

$HF_2^{-1} = F^{-1} + HFaq$

$HFaq = H^{+1} + F^{-1}$

$HSO_4^{-1} = H^{+1} + SO_4^{-2}$

$Na_2SO_4 \cdot 10H_2O = 2Na^{+1} + SO_4^{-2} + 10H_2O$

$Na_2SO_4ppt = 2Na^{+1} + SO_4^{-2}$

$NaClppt = Na^{+1} + Cl^{-1}$

$NaFaq = Na^{+1} + F^{-1}$

$NaFppt = Na^{+1} + F^{-1}$

$NaHSO_4ppt = Na^{+1} + HSO_4^{-1}$

$NaOH \cdot H_2Oppt = Na^{+1} + OH^{-1} + H_2O$

$NaOHppt = Na^{+1} + OH^{-1}$

$NaSO_4^{-1} = Na^{+1} + SO_4^{-2}$

Vapor-Liquid and Liquid-Liquid Equilibria:

Vapor-liquid equilibrium is considered for H_2O / Benzene / CH_4 / Ethanol / HCl / HF / Methanol / M-Xylene / N_2 / Toluene.

Alphabetical Component Index of Electrolyte Models

The Electrolyte models that contain a given component (or ionic species) are listed directly underneath the component names in this index. The phases allowed for the components are given in parentheses following the component names, where:

> AQ = aqueous phase (ion, ion pair, or dissolved neutral molecule)
>
> CI = completely ionized aqueous electrolyte
>
> V= vapor phase
>
> S = solid phase
>
> HC = hydrocarbon or non-aqueous liquid phase.

For example, it can be deduced from the index that aqueous calcium ion, Ca^{+2}(AQ), hydroxide ion, OH^{-1}(AQ), calcium mono-hydroxide ion pair, $CaOH^{+1}$(AQ), and solid calcium hydroxide, $Ca(OH)_2$(S) are included in the CANA model. The CANX model has the same components as CANA; however, solid phases of the components are not allowed. Consequently, $Ca(OH)_2$ in the CANX model is treated as an aqueous electrolyte that is completely ionized (CI).

A

- Acetacid (AQ)
  - SW023
  - SW2X3
- $Acetate^{-1}$ (AQ)
  - SW021
  - SW2X1

B

- Ba^{+2} (AQ)
  - GEOT1
  - OILF1
- $BaCl_2$ (CI)
  - GEOT1
- $BaCl_2$ (S)
  - OILF1
- $BaCO_3$ (S)
  - GEOT1
  - OILF1
- $BaOH^{+1}$ (AQ)
  - GEOT1
  - OILF1
- $Ba(OH)_2$ (S)
  - OILF1
- $BaSO_4$ (S)
  - GEOT1
  - OILF1
- Benzene (V,AQ,HC)
  - TWL2
- BO_2^{-1} (AQ)
  - BENF1
  - BENX1
- $B(OH)_3$ (AQ)
  - BENF1
  - BENX1
  - GEOT1
- $B(OH)_3OH^{-1}$ (AQ)
  - GEOT1

C

- C_2H_4 (V,AQ)
  - BENF2
  - BENX2C2H6(V,AQ)
  - BENF2
  - BENX2
  - DEA2
  - DGA2
  - DIPA2
  - HOTC2
  - HOTX2
  - MDEA2
  - MEA2
  - SW012
  - SW032
  - SW042
  - SW1X2
  - SW3X2
  - SW4X2
  - TEA2
- C_3H_8 (V,AQ)
  - BENF6
  - BENX6
  - DEA7
  - DGA7
  - DIPA7
  - HOTC7
  - HOTX7
  - MDEA7
  - MEA7
  - SW017
  - SW037
  - SW047
  - SW1X7
  - SW3X7
  - SW4X7
  - TEA7
- C_4H_{10} (V,AQ)
  - DEA6
  - DGA6
  - DIPA6
  - MDEA6
  - MEA6
  - SW036
  - SW046
  - SW3X6
  - SW4X6
  - TEA6
- $C_6H_5O^{-1}$ (AQ)
  - SW011
  - SW021
  - SW031
  - SW041
  - SW1X1
  - SW2X1
  - SW3X1
  - SW4X1
  - TWL1
- C_6H_5OH (V,AQ)
  - SW011
  - SW021
  - SW031
  - SW041
  - SW1X1
  - SW2X1
  - SW3X1
  - SW4X1
- C_6H_5OH (V,AQ,HC)
  - TWL1
- Ca^{+2} (AQ)
  - CANA1
  - CANX1
  - GEOT1
  - OILF1
  - SCRU1
  - SCRX1
  - TWL2
- Ca2EDTA (AQ)
  - GEOT1
- $CaCl^{+1}$ (AQ)
  - TWL1
  - TWL2
- $CaCl_2$ (CI)
  - CANX1
  - GEOT1
  - SCRX1
- $CaCl_2$ (S)
  - CANA1
  - OILF1
  - SCRU1
  - TWL1
  - TWL2
- $CaCl_2 \cdot H_2O$ (S)
  - TWL1
  - TWL2
- $CaCl_2 \cdot 2H_2O$ (S)
  - TWL1
  - TWL2
- $CaCl_2 \cdot 4H_2O$ (S)
  - TWL1
  - TWL2
- $CaCl_2 \cdot 6H_2O$ (S)
  - TWL1
  - TWL2
- $CaCO_3$ (CI)
  - SCRX1
- $CaCO_3$ (S)
  - GEOT1
  - OILF1
  - SCRU1
  - TWL1
- $CaCOOH^{+1}$ (AQ)
  - SCRU1
  - SCRX1
- CaF^{+1} (AQ)
  - TWL2
- CaF_2 (S)
  - TWL2
- $CaHCO_3^{+1}$ (AQ)
  - GEOT1
  - OILF1
  - SCRU1
  - SCRX1
  - TWL1
- $Ca(HCO_3)_2$ (CI)
  - TWL1
- $CaHSO_4$ (CI)
  - TWL2
- $CaOH^{+1}$ (AQ)
  - CANA1
  - CANX1
  - GEOT1
  - OILF1
  - SCRU1
  - SCRX1
  - TWL1
  - TWL2
- $Ca(OH)_2$ (CI)
  - CANX1
  - SCRX1
- $Ca(OH)_2$ (S)
  - CANA1
  - OILF1
  - SCRU1
  - TWL1
  - TWL2
- $CaSO_4$ (AQ)
  - CANX1
  - GEOT1
- $CaSO_4$ (S,AQ)
  - CANA1
  - OILF1
  - TWL2
- $CaSO_4 \cdot 2H_2O$ (S)
  - TWL2
- CH_4 (V,AQ)
  - BENF1
  - BENX1
  - DEA1
  - DGA1
  - DIPA1
  - HOTC1
  - HOTX1
  - MDEA1
  - MEA1
  - SW011
  - SW021
  - SW031
  - SW041
  - SW1X1
  - SW2X1
  - SW3X1
  - SW4X1
  - TEA1
- CH_4 (V,AQ,HC)
  - TWL1
  - TWL2
- Cl^{-1} (AQ)
  - CANA1
  - CANX1
  - CAUS1
  - CAUX1
  - Cl21
  - CLSF1
  - GENE1
  - GENX1
  - GEOT1
  - HCL1
  - OILF1
  - SALT1
  - SCRU1
  - SCRX1
  - SW011
  - SW021
  - SW031
  - SW1X1

- SW2X1
- SW3X1
- TWL1
- TWL2

- Cl_2 (V,AQ)
  - CAUS1
  - CAUX1
  - CL21
  - CLSF1
  - GENE1
  - GENX1
- ClO^{-1} (AQ)
  - CAUS1
  - CAUX1
  - CL21
  - CLSF1
  - GENE1
  - GENX1
- CN^{-1} (AQ)
  - ACID1
  - SCRU1
  - SCRX1
  - SW011
  - SW021
  - SW031
  - SW1X1
  - SW2X1
  - SW3X1
- CO (V,AQ)
  - ACID2
  - BENF2
  - BENX2
  - HOTC2
  - HOTX2
  - SW032
  - SW3X2
- CO_2 (V,AQ)
  - ACID2
  - BENF2
  - BENX2
  - DEA2
  - DGA2
  - DIPA2
  - GENE2
  - GENX2
  - GEOT2
  - HOTC2
  - HOTX2
  - MDEA2
  - MEA2
  - OILF2
  - SCRU2
  - SCRX2
  - SW012
  - SW022
  - SW032
  - SW042
  - SW052
  - SW1X2
  - SW2X2
  - SW3X2
  - SW4X2
  - TEA2
- CO_2 (V,AQ,HC)
  - TWL1
- CO_3^{-2} (AQ)
  - ACID2
  - BENF2
  - BENX2
  - DEA2
  - DGA2
  - DIPA2
  - GENE2
  - GENX2
  - GEOT2
  - HOTC2
  - HOTX2
  - MDEA2
  - MEA2
  - OILF2
  - SCRU2
  - SCRX2
  - SW012
  - SW022
  - SW032
  - SW042
  - SW052
  - SW1X2
  - SW2X2
  - SW3X2
  - SW4X2
  - TEA2
  - TWL1
- $COOH^{-1}$ (AQ)
  - ACID2
  - SCRU2
  - SCRX2
  - SW012
  - SW1X2
- Cu^{+2} (AQ)
  - GEOT
- $CuCl_2$ (CI)
  - GEOT2

D

- $DEACO_2^{-1}$ (AQ)
  - DEA2
- DEAH (AQ)
  - DEA2
- $DEAH_2^{+1}$ (AQ)
  - DEA2
- $DGACO_2^{-1}$ (AQ)
  - DGA2
- DGAH (AQ)
  - DGA2
- $DGAH_2^{+1}$ (AQ)
  - DGA2
- $DIPACO_2^{-1}$ (AQ)
  - DIPA2
- DIPAx (AQ)
  - DIPA2
- $DIPAH_2^{+1}$ (AQ)
  - DIPA

E

- $EDTA^{-4}$ (AQ)
  - GEOT2
- Ethanol (V,AQ,HC)
  - TWL2

F

- F^{-1} (AQ)
  - TWL2
- Fe_3O_4 (S)
  - GEOT2
- $FeII^{+2}$ (AQ)
  - GEOT2
- $FeIICl^2$ (CI)
  - GEOT2
- $FeIIHCO_3^{+1}$ (AQ)
  - GEOT2
- $FeIII^{+3}$ (AQ)
  - GEOT2
- $FeIIICl_3$ (CI)
  - GEOT2
- $FeIIOH^{+1}$ (AQ)
  - GEOT2

H

- H^{+1} (AQ)
  - (all models)
- H_2 (V,AQ)
  - ACID3
  - BENF3
  - BENX3
  - HOTC3
  - HOTX3
  - SW033
  - SW3X3
- H_2O (V,AQ)
  - (all models)
- $H_2P_2O_7^{-2}$ (AQ)
  - ACID3
  - BENF3
  - BENX3
  - PHOS3
  - SW013
  - SW1X3
- $H_2PO_4^{-1}$ (AQ)
  - ACID3
  - BENF3
  - BENX3
  - PHOS3
  - SW013
  - SW1X3
- H_2S (V,AQ)
  - BENF3
  - BENX3
  - DEA3
  - DGA3
  - DIPA3
  - GEOT3
  - HOTC3
  - HOTX3
  - MDEA3
  - MEA3
  - SW013
  - SW023
  - SW033
  - SW043
  - SW053
  - SW1X3
  - SW2X3
  - SW3X3
  - SW4X3
  - TEA3
- H_2S (V,AQ,HC)
  - TWL1
- H_2SO_3 (CI)
  - CLSF3
  - SULF3
- H_2SO_4 (CI)
  - CANA3
  - CANX3
  - CAUS3
  - CAUX3
  - CLSF3
  - OILF3
  - SULF3
  - TWL2
- $H_3P_2O_7^{-1}$ (AQ)
  - ACID3
  - BENF3
  - BENX3
  - PHOS3
  - SW013
  - SW1X3
- H_3PO_4 (CI)
  - ACID3
  - BENF3
  - BENX3
  - PHOS3
  - SW013
  - SW1X3
- HCl (CI)
  - GEOT3
- HCl (V,AQ)
  - CANA3
  - CANX3
  - CAUS3
  - CAUX3
  - CL23
  - CLSF3
  - GENE3
  - GENX3

  - HCL3
  - OILF3
  - SCRU3
  - SCRX3
  - SW013
  - SW023
  - SW033
  - SW1X3
  - SW2X3
  - SW3X3
  - TWL1
  - TWL2
- HClO (V,AQ)
  - CAUS3
  - CAUX3
  - CL23
  - CLSF3
  - GENE3
  - GENX3
- HCN (V,AQ)
  - ACID4
  - SCRU4
  - SCRX4
  - SW014
  - SW024
  - SW034
  - SW1X4
  - SW2X4
  - SW3X4
- HCO_3^{-1} (AQ)
  - ACID4
  - BENF4
  - BENX4
  - DEA4
  - DGA4
  - DIPA4
  - GENE4
  - GENX4
  - GEOT4
  - HOTC4
  - HOTX4
  - MDEA4
  - MEA4
  - OILF4
  - SCRU4
  - SCRX4
  - SW014
  - SW024
  - SW034
  - SW044
  - SW054
  - SW1X4
  - SW2X4
  - SW3X
  - SW4X4
  - TEA4
  - TWL1
- HCOOH (V,AQ)
  - ACID4
  - SCRU4
  - SCRX4
  - SW014
  - SW1X4
- HF_2^{-1} (AQ)
  - TWL2
- HF (V,AQ,HC)
  - TWL2
- $HP_2O_7^{-3}$ (AQ)
  - ACID4
  - BENF4
  - BENX4
  - PHOS4
  - SW014
  - SW1X4
- HPO_4^{-2} (AQ)
  - ACID4
  - BENF4
  - BENX4
  - PHOS4
  - SW014
  - SW1X4
- HS^{-1} (AQ)
  - BENF4
  - BENX4
  - DEA4
  - DGA4
  - DIPA4
  - GEOT4
  - HOTC4
  - HOTX4
  - MDEA4
  - MEA4
  - SW014
  - SW024
  - SW034
  - SW044
  - SW054
  - SW1X4
  - SW2X4
  - SW3X4
  - SW4X4
  - TEA4
- HSO_3^{-1} (AQ)
  - CAUS4
  - CAUX4
  - CLSF4
  - GENE4
  - GENX4
  - GEOT4
  - SCRU4
  - SCRX4
  - SULF4
- HSO_4^{-1} (AQ)
  - CANA4
  - CANX4
  - CAUS4
  - CAUX4
  - CLSF4
  - GEOT4
  - OILF4
  - SULF4
  - TWL2

K

- K^{+1} (AQ)
  - BENF4
  - BENX4
  - CA2X4
  - CAU24
  - GEOT4
  - OILF4
  - SALT4
- K_2CO_3 (CI)
  - BENX4
- K_2CO_3 (S)
  - BENF4
  - OILF4
- K_2SO_4 (S)
  - OILF4
- KCl (CI)
  - GEOT4
- KCl (S)
  - OILF4
  - SALT4
- $KHCO_3$ (CI)
  - BENX4
- $KHCO_3$ (S)
  - BENF4
  - OILF4
- KHS (CI)
  - BENF4
  - BENX4
- KOH (CI)
  - BENX4
  - CA2X4
- KOH (S)
  - BENF4
  - CAU24
  - OILF4

L

- Li^{+1} (AQ)
  - GEOT4
- LiCl (CI)
  - GEOT4

M

- MDEAH (AQ)
  - MDEA4
- $MDEAH_2^{+1}$ (AQ)
  - MDEA4
- $MEACO_2^{-1}$ (AQ)
  - MEA4

- MEAH (V,AQ)
  - MEA4
- $MEAH_2^{+1}$ (AQ)
  - MEA4
- Methanol (V,AQ,HC)
  - TWL2
- Mg^{+2} (AQ)
  - CANA4
  - CANX4
  - GEOT4
  - OILF4
- $MgCl^{+1}$ (AQ)
  - CANA4
  - CANX4
  - GEOT4
  - OILF4
- $MgCl_2$ (CI)
  - CANX4
  - GEOT4
- $MgCl_2$ (S)
  - CANA4
  - OILF4
- $MgCO_3$ (S)
  - GEOT5
  - OILF5
- $MgHCO_3^{+1}$ (AQ)
  - GEOT5
  - OILF5
- $Mg(OH)_2$ (CI)
  - CANX5
- $Mg(OH)_2$ (S)
  - CANA5
  - OILF5
- $MgSO_4$ (CI)
  - CANX5
- $MgSO_4$ (S)
  - CANA5
  - OILF5
- Mn^{+2} (AQ)
  - GEOT5
- $MnCl_2$ (CI)
  - GEOT5
- $MnHCO_3^{+1}$ (AQ)
  - GEOT5
- $MnOH^{+1}$ (AQ)
  - GEOT5
- M-Xylene (V,AQ,HC)
  - TWL2

N

- N_2 (V,AQ)
  - ACID5
  - BENF5
  - BENX5
  - CL25
  - CLSF5
  - DEA5
  - DGA5
  - DIPA5
  - GENE5
  - GENX5
  - HOTC5
  - HOTX5

  - MDEA5
  - MEA5
  - SCRU5
  - SCRX5
  - SW015
  - SW035
  - SW045
  - SW1X5
  - SW3X5
  - SW4X5
  - TEA5
- N_2 (V,AQ,HC)
  - TWL2
- Na^{+1} (AQ)
  - CA2X5
  - CANA5
  - CANX5
  - CAU25
  - CAUS5
  - CAUX5
  - GENE5
  - GENX5
  - GEOT5
  - HOTC5
  - HOTX5
  - OILF5
  - SALT5
  - SCRU5
  - SCRX5
  - SW015
  - SW025
  - SW035
  - SW045
  - SW1X5
  - SW2X5
  - SW3X5
  - SW4X5
  - TWL2
- Na_2CO_3 (CI)
  - GENX5
  - HOTX5
  - SCRX5
  - SW1X5
  - SW2X5
  - SW3X5
  - SW4X5
- Na_2CO_3 (S)
  - GENE5
  - HOTC5
  - OILF5
  - SCRU5
  - SW015
  - SW025
  - SW035
  - SW045
  - TWL1
- $Na_2CO_3\cdot H_2O$ (S)
  - TWL1
- $Na_2CO_3\cdot 7H_2O$ (S)
  - TWL1
- $Na_2CO_3.10H_2O$ (S)
  - TWL1
- Na_2S (CI)
  - GEOT5
- Na_2SO_3 (CI)
  - CAUX5
  - GEOT5
- Na_2SO_3 (S)
  - CAUS5
- Na_2SO_4 (CI)
  - CANX5
  - CAUX5
  - GEOT5
- Na_2SO_4 (S)
  - CANA5
  - CAUS5
  - OILF5
  - TWL2
- $Na_2SO_4.10H_2O$
  - TWL2
- Na_4EDTA (AQ)
  - GEOT5
- NaCl (CI)
  - CANX5
  - CAUX5
  - GENX5
  - GEOT5
  - OILF5
  - SCRX5
  - SW1X5
  - SW2X5
  - SW3X5
- NaCl (S)
  - CANA5
  - CAUS5
  - GENE5
  - SALT5
  - SCRU5
  - SW015
  - SW025
  - SW035
  - TWL1
  - TWL2
- NaCOOH (CI)
  - SCRX5
- NaCOOH (S)
  - SCRU5
- NaF (S)
  - TWL2
- $NaHCO_3$ (CI)
  - GENX5
  - HOTX5
  - SCRX5
  - SW1X5
  - SW2X5
  - SW3X5
  - SW4X5
- $NaHCO_3$ (S)
  - GENE5
  - HOTC5
  - OILF5
  - SCRU5
  - SW015
  - SW025
  - SW035
  - SW045
  - TWL1
- NaHS (CI)
  - SW015
  - SW025
  - SW035
  - SW1X5
  - SW2X5
  - SW3X5
  - TWL1
- $NaHSO_4$ (S)
  - TWL2
- NaOH (CI)
  - CA2X5
  - CANX5
  - CAUX5
  - GENX5
  - HOTX5
  - SCRX5
  - SW1X6
  - SW2X6
  - SW3X6
  - SW4X6
- NaOH (S)
  - CANA6
  - CAU26
  - CAUS6
  - GENE6
  - HOTC6
  - OILF6
  - SCRU6
  - SW016
  - SW026
  - SW036
  - SW046
  - TWL1
  - TWL2
- $NaOH\cdot H_2O$ (S)
  - TWL1
  - TWL2
- $NaSO_4^{-1}$ (AQ)
  - CANA6
  - CANX6
  - CAUS6
  - CAUX6
  - GEOT6
  - OILF6
  - TWL2
- $NH_2CO_2^{-1}$ (AQ)
  - ACID6
  - BENF6
  - BENX6
  - GEOT6
  - SW016
  - SW026
  - SW036
  - SW046
  - SW056
  - SW1X6
  - SW2X6
  - SW3X6
  - SW4X6
  - TWL1
- NH_3 (V,AQ)
  - ACID6
  - BENF6
  - BENX6
  - GEOT6
  - SW016
  - SW026
  - SW036
  - SW046
  - SW056
  - SW1X6
  - SW2X6
  - SW3X6
  - SW4X6
- NH_3 (V,AQ,HC)
  - TWL1
- NH_4^{+1} (AQ)
  - ACID6
  - BENF6
  - BENX6
  - GEOT6
  - SW016
  - SW026
  - SW036
  - SW046
  - SW056
  - SW1X6
  - SW2X6
  - SW3X6
  - SW4X6
  - TWL1
- NH_4Cl (S)
  - TWL1
- NH_4HCO_3 (S)
  - TWL1
- NH_4HS (S)
  - TWL1
- $(NH_4)_2SO_4$ (S)
  - GEOT6

O

- O_2 (V,AQ)
  - CLSF6
  - GENE6
  - GENX6
  - SCRU6
  - SCRX6
  - SW036
  - SW3X6

- OH^{-1} (AQ)
  - (all models)6

P

- $P_2O_7^{-4}$ (AQ)
  - ACID6
  - BENF6
  - BENX6
  - PHOS6
  - SW016
  - SW1X6
- Pb^{+2} (AQ)
  - GEOT6
- $PbCl^2$ (Cl)
  - GEOT6
- PO_4^{-3} (AQ)
  - ACID6
  - BENF6
  - BENX6
  - PHOS6
  - SW016
  - SW1X6

S

- S^{-2} (AQ)
  - BENF7
  - BENX7
  - DEA7
  - DGA7
  - DIPA7
  - GEOT7
  - HOTC7
  - HOTX7
  - MDEA7
  - MEA7
  - SW017
  - SW027
  - SW037
  - SW047
  - SW057
  - SW1X7
  - SW2X7
  - SW3X7
  - SW4X7
  - TEA7
  - TWL1
- $S_2O_5^{-2}$ (AQ)
  - CAUS7
  - CAUX7
  - CLSF7
  - GENE7
  - GENX7
  - SCRU7
  - SCRX7
  - SULF7
- SO_2 (V,AQ)
  - CAUS7
  - CAUX7
  - CLSF7
  - GENE7
  - GENX7
  - SCRU7
  - SCRX7
  - SULF7
- SO_3^{-2} (AQ)
  - CAUS7
  - CAUX7
  - CLSF7
  - GENE7
  - GENX7
  - GEOT7
  - SCRU7
  - SCRX7
  - SULF7
- SO_4^{-2} (AQ)
  - CANA7
  - CANX7
  - CAUS7
  - CAUX7
  - CLSF7
  - GEOT7
  - OILF7
  - SULF7
  - TWL2
- Sr^{+2} (AQ)
  - GEOT7
- $SrCl_2$ (Cl)
  - GEOT7
- $SrCO_3$ (S)
  - GEOT7
- $SrSO_4$ (S)
  - GEOT7

T

- $TEACO_2^{-1}$ (AQ)
  - TEA7
- TEAH (AQ)
  - TEA7
- $TEAH_2^{+1}$ (AQ)
  - TEA7
- Toluene (V,AQ,HC)
  - TWL1
  - TWL2

Z

- Zn^{+2} (AQ)
  - GEOT7
- $ZnCl_2$ (Cl)
  - GEOT7
- $ZnHCO_3^{+1}$ (AQ)
  - GEOT7

Chapter E3
Creating a User-Added Electrolyte Model

Available Tools for Creating Electrolyte Models

Users can develop their own user-added electrolyte models for use in Electrolytes versions of PRO/II by using Chemistry Wizard version 3.1 from OLI Systems, Inc.This upgrade includes OLI's new state-of-the-art Mixed Solvent Electrolyte (MSE) chemistry model that enables chemical process simulations that heretofore were not possible to accomplish.

Starting with PRO/II version 8.0, OLI Systems, Inc. assumed responsibility for documenting the tools they supply. Because that documentation is the most current, the instructions that previously appeared in this chapter have been removed.

The PRO/II installation package includes the following documentation to instruct users in generating their own electrolyte models. During installation, these files may be copied to the \Manual\OLI subdirectory of the PRO/II installation directory.

- *OLI Chemistry Wizard User Guide* provides complete documentation of the new tool set. This Guide identifies the current version of the Chemistry Wizard used by PRO/II.
- *PRO/II-OLI 6.6_Components.xls*, a Microsoft Excel spreadsheet, provides an expanded list of electrolyte components as included in the 40 pre-defined systems installed with PRO/II. The 5.6, 6.6, and 7.0 versions of the predefined electrolyte data base files all contain the same slate of electrolyte species and components.

Both documents are included on the PRO/II installation disk.

Compatibility with Previous Versions of PRO/II

Formerly, PRO/II incorporated version 5.6, and then version 6.3 of the OLI electrolytes calculation engine. These versions used the Electrolyte Utility Package (EUP) provided by OLI Systems, Inc., to generate electrolyte models. This program now is obsolete.

Beginning with PRO/II version 7.1, PRO/II upgraded to OLI engine version 6.6. Also, the EUP was replaced by the newer OLI Chemistry Wizard version 1.0. Concurrently, OLI Systems assumed responsibility for support and documentation of the newer tools. This program also is obsolete.

Beginning with version 9.1, PRO/II upgraded to version 8.2.4 of the OLI engine and the accompanying Chemistry Wizard version 3.1. These versions of the tools currently are used by PRO/II.

Note: All of the older tools mentioned in this section now are obsolete and no longer are available. When creating new electrolyte models, be sure to use the current version of the Chemistry Wizard.

Note: While the older tools no longer are available, older electrolyte models created by using them still function properly in PRO/II. Specifically, PRO/II still is able to translate and use electrolyte models created for earlier OLI engine versions 5.6 and 6.6. However, we urge all electrolyte users to upgrade to the latest version at their earliest convenience.

Chapter E4
Electrolyte Effects In Simulations

Overview

Throughout the development of PRO/II Electrolytes, a major goal has been to integrate the Electrolyte thermodynamic methods seamlessly into PRO/II's graphical user interface (GUI). In other words, as far as the unit operations are concerned, PRO/II Electrolytes should operate as "just another thermodynamic method." There are several units still unavailable for use with Electrolytes, while others have partial functionality or other special considerations. This document discusses work-arounds, where possible, for these limitations.

General Considerations

The most fundamental consideration that can make PRO/II Electrolytes different from other thermodynamic methods is that it is completely tied to the flash algorithm, K-value, enthalpy, and density methods as developed by OLI Systems Inc. (Morris Plains, N.J.). OLI's flash algorithm does not work exactly the same as the "normal" flash in PRO/II. In some ways that is beneficial, but it can create problems for a few PRO/II unit operations. The following are important differences between the two, along with statements regarding the consequences of those differences:

- There are conditions (most commonly where there is no aqueous liquid phase) in which OLI's thermodynamic method is simply not applicable. In such conditions, the flash will usually fail to converge. This is in contrast to most non-electrolyte PRO/II thermodynamic methods, in which the flash will usually return some sort of an answer even if you have chosen an inappropriate method for the conditions.
- There are occasionally valid conditions for which OLI's flash fails to converge. The most common of these failures occurs in adiabatic flashes with nearly pure fluids that cross a phase boundary; this case will be discussed below in Simple HX. The only other known significant problem is that OLI can have problems converging a flash in which a large amount of a sparingly soluble solid will precipitate out. This problem is alleviated somewhat for internal flowsheet streams by implementing

an initial guess for solids based on the solids in the feed. This can be done for external feed streams by putting an estimate of the solid that will precipitate on the `SOLID` statement for the stream.

- There are some thermodynamic quantities that OLI does not calculate. In particular, OLI does not return values for the entropy or heat capacity of electrolyte streams. This means that electrolyte thermodynamic methods cannot be used for units that require entropy, such as the Gibbs reactor. Some units also use heat capacity internally; in some cases special adjustments have been made to the code for electrolyte thermodynamic methods.
- The list of specifications recognized by OLI's flash is different from that recognized by PRO/II's flash. Many of the specifications can be translated, but some (such as isentropic flashes) cannot.
- OLI, under the current release of PRO/II Electrolytes, will now do liquid/liquid equilibrium (LLE); that is, it can handle a second (organic) liquid phase. However, this capability is limited to non-column unit operations such as the flash drum and the splitter.

SPEC, DEFINE, VARY

In general, all of the flowsheet parameters as listed in Section 43 of the PRO/II Keyword Manual are available for cross-referencing, and anything that can be varied with other thermodynamic methods can be varied with electrolytes. There are some stream properties (for example, D86 and WFRAC) that do not make sense for aqueous electrolyte streams.

In addition, the electrolyte-specific stream property pH is available for cross-referencing.

Basic Unit Operations

The following table indicates which unit operations support the use of the Electrolyte module.

Table E4-1: Unit Operations That Support Electrolytes

Support Electrolytes	NO Electrolyte Support
Flash	**Compressor**
Pump, Valve, Mixer, Splitter	**Expander**
Column ELDIST (Electrolyte Algorithm)	**Column** (All algorithms except ELDIST)
Heat Exchangers **Simple, Rigorous, LNG, ACE**	Heat Exchangers **Furnace** (Fired Heater)
Reactors **Batch,** **Conversion,** **Equilibrium**	Reactors **Gibbs,** **CSTR,** **Plug Flow**
Solids Handling **Counter-Current Decanter** **Crystallizer** **Dissolver** **Filtering Centrifuge** **Rotary Filter**	Solids Handling **Dryer** **Freezer** **Melter**
Utilities **BVLE** **HCURVE**	Utilities **HEXTABLES** **Hydrates** **Phase Envelope**
Calculator (see DEFINE)	
Controller, Optimizer	
Pipe	

Flash Drum

As with all flash-type units, the most important caution is not to use electrolyte thermodynamic methods at conditions where it is not valid. Apart from that, most of the functionality of the unit is available for electrolytes. The flash drum is able to handle a second (organic) liquid phase, as well as a vapor and/or precipitated solid phase. As mentioned before, the `ISENTROPIC` flash is not available. The `DEWHC` and `DEWWATER` options are not available (or meaningful) because electrolyte methods do not decant water.

The `TPSPEC` construct is currently limited with electrolytes. The one added feature is the ability to SPEC the pH of an outlet liquid stream. The only other internal flash spec allowed is a vapor or liquid flowrate as a fraction of the molar flowrate of a stream that is the only feed to the unit:

```
SPEC STRM=L1, RATE, RATIO, STRM=F1, RATE, VALUE=0.8
```

The `REFFEED` construct may also be used to reference a combined unit feed. One can simulate any other `TPSPEC` by putting the flash drum inside a controller loop.

For example, to specify a liquid weight fraction:

```
CONTROLLER UID=C1

SPEC STRM=L1, COMP=2, FRACTION(W), VALUE=0.3

VARY FLASH=FL1, TEMP
```

Pump, Valve, Mixer, Splitter

These units all work normally with electrolytes. There is a potential problem in the Valve, in which the adiabatic flash may fail to converge for a nearly pure fluid if the user is crossing a phase boundary. Frequently, this may be alleviated simply by taking the valve in two (or more) steps; a stream that will not converge when being throttled from 50 to 15 psi may converge if it is taken to 20 psi in one valve and then down to 15 in a second valve (in the second case it gets a better initial guess). Alternatively, the valve can be simulated by putting an ISOT flash (or a vapor-fraction-specified flash) inside a controller with a controller specification of zero duty on the flash.

Compressor or Expander

Both of these units require entropy and are consequently not available with PRO/II Electrolytes. There are, however, still cases where they may be used in electrolyte flow sheets. For example, if the units operate only on a vapor phase (steam, for example), there is no reason not to use a non-electrolyte thermodynamic method. Also, if the stream in question is nearly pure water, the units may be replaced by a flash drum to give the DP in the flowsheet, and then they may be modeled separately using a pure water stream to design the unit and feed back the correct temperature for the flash drum.

Pipe

This unit is functional with PRO/II Electrolytes. The Pipe might have problems doing two-phase flow with nearly pure water streams because it does adiabatic/fixed-duty flashes on which OLI may fail. Of course, the Pipe requires viscosities and sometimes surface tensions. See the comments on Transport Properties below for availability in regards to electrolytes.

Heat Exchangers

Simple HX

The full functionality of this unit, including all the Spec types, is available to PRO/II Electrolytes.

There is, however, one commonly encountered condition that can cause a problem: when the hot side of an exchanger is condensing steam (in the case of a multi-effect evaporator, the steam may come as the vapor from the cold side of another HX). The HX does a fixed-duty flash on this stream. Unfortunately, OLI has great difficulty converging for pure steam crossing the two-phase region. The remedy for this situation is to use non-electrolyte thermodynamic methods for the side of the HX that uses condensing steam; because it is a pure water stream, there is no need to use electrolyte thermodynamic methods anyway.

Rigorous HX

This unit is fully functional with PRO/II Electrolytes.

LNGHX

This unit is fully functional with PRO/II Electrolytes. The comments regarding the use of condensing steam (see the HX discussion above) apply here as well.

Electrolyte Distillation Column Algorithm (ELDIST)

This column algorithm was designed to solve non-ideal aqueous electrolytic distillation columns involving ionic species. It uses a Newton-Raphson method to solve the mass balance, vapor-liquid equilibrium, and specification equations simultaneously. The

K-values and enthalpies are supplied by the electrolyte thermodynamic model.

The Electrolytic Column Algorithm is selected from the *Column Algorithm* drop-down list in the *Column* dialog box.

> ***Note:*** Electrolytic thermodynamic models support only VLE, and so total phase draws are not permitted.

Advantages and disadvantages of the Electrolytic Column Algorithm follow:

- Advantages —
  - Rigorously models ionic equilibrium systems.
  - Solves highly non-ideal distillation columns.
- Disadvantages —
  - Side columns are not supported.
  - Pumparounds and tray hydraulics are not available
  - Certain column specifications and variables are not permitted.

For the most part, the functionality of ELDIST is the same as that of Chemdist. That means, for example, that pumparounds are not allowed, and the range of Specs available is limited. In contrast to Chemdist, ELDIST does not allow the specification of component flows or mole fractions in liquid product streams. This is because the presence of ionic reactions sometimes makes it ambiguous as to the “correct” way to report the composition of a liquid (this is the problem known as “reconstitution”). It should be noted that, as with other units, non-allowed Specs may be met by putting the column inside a controller or optimizer. Another limitation of ELDIST is neither VLLE distillation nor liquid-liquid extraction is available. Column hydraulics also are unavailable at present, but may be enabled in the future.

Because electrolyte systems often are very non-ideal, ELDIST sometimes has difficulty converging. Supplying initial estimates for column tray temperatures and/or vapor-liquid flows (perhaps obtained by solving a simplification of the problem) will often aid convergence. Complete initial estimates (including compositions) can be saved from one run with the PRINT PROFILE option. These estimates not only help similar columns converge, but they also save calculation time because the ELDIST initial estimate generator is somewhat time consuming.

Reactors

Equilibrium and Conversion Reactors

It should be remembered that, in PRO/II Electrolytes, you do not necessarily need a Reactor to get a reaction. For example, water, NaOH and HC1 will yield NaCl and H_2O in a flash drum or any other unit that does a flash. For reactions in aqueous solution that involve only a “rearrangement” of ions, the Reactor is not necessary.

If it is desired to use a reactor with Electrolyte thermodynamic methods, the user is currently limited to isothermal mode with the NOHBAL option turned on. These limitations will probably be lifted in later releases of PRO/II Electrolytes. Of course, fixed-duty mode can be simulated by putting an isothermal reactor in a controller loop.

Gibbs Reactor

This unit requires entropy, and therefore cannot be used with electrolyte thermodynamic methods. Again, since PRO/II Electrolytes performs a simultaneous phase and chemical equilibration, the functionality of a Gibbs reactor would be redundant if the only reactions were ionic rearrangements.

Plug Flow and Continuously Stirred Tank Reactors

These units are not currently available with Electrolyte thermodynamic methods. The primary reason for this is that neither unit is able to handle components with phase type LS. Almost all reasonable electrolyte models would contain such components.

Solids Handling Utilities

Dryer

This unit is not available with Electrolyte thermodynamic methods. The main reason for this is that OLI has difficulty converging flashes with lots of solid and little water. These would be conditions likely to arise in the dryer. It might be possible to work around this last problem by stringing multiple dryers in series to model one unit.

RFilter, FCentrifuge, CCDecanter

These units have their full functionality with Electrolyte thermodynamic methods. There is, however, one thing to watch out for. These units are not designed to do phase equilibrium calculations, so if the conditions are such that more solids will precipitate (or dissolve) in the unit, solids may appear (or disappear) in the filtrate stream following solution of the unit.

Dissolver and Crystallizer

These two units may be used successfully with PRO/II Electrolytes if care is taken in setting up the problem. If the only purpose is to produce solid precipitate (or dissolve solids) to the equilibrium concentration, a flash drum should be used instead. The ability of OLI's flash to do SLE is a big advantage over PRO/II's non-electrolyte thermodynamic methods in this case. The crystallizer and dissolver should be used with electrolytes in only two cases:

- When the process does not reach equilibrium.
- When it is important to know the particle size distribution (PSD) of the product.

Both of these units are unable to go all the way to equilibrium (of course, this is not an issue unless the kinetics are zero-order). In such cases, the warning message will often say something similar to "supersaturation limit reached." The dissolver unit will also fail at conditions such that all of the solid will dissolve. A flash drum should be used in such cases.

As for the kinetics in these units, the parameters must be adjusted to match experimental data. While the dissolver has a correlation to predict mass transfer, that correlation is completely inappropriate for electrolyte solutes. When testing different parameters to match data, it is best (because of the problems getting to complete equilibrium) to start with "slow" kinetics and work your way up.

For electrolytes, both of these units operate only in isothermal mode. Fixed-duty conditions may be simulated by putting the unit in a controller loop.

Melter/Freezer

This unit exists only to circumvent non-electrolyte PRO/II's inability to do SLE, so it has no useful purpose for electrolytes and is therefore disallowed.

Calculator

See comments above regarding what may be accessed with the DEFINE statement.

Stream Calculator

This unit is fully functional with PRO/II Electrolytes.

Flowsheet Control

You are likely to get warnings in electrolyte problems about components with undefined separation factors. This is because a PRO/II Electrolytes model will add any components it has to what is on the LIBID statement, so there will be more components in the simulation than the user might have thought. As long as none of these added components are produced in the simulation, these warnings may be safely ignored.

Miscellaneous

It is also possible to encounter a flash failure in this unit when doing feed blending, such as is commonly done to figure the rate of a makeup stream. This is because the unit first does an adiabatic flash on the entire blended feed, even if only one or two components are going to be picked out with an FPROD statement. In some cases, even though you may just be figuring the makeup rate of a pure water stream, the blended feed may put you in a composition region that OLI cannot solve (for example, 1 mole of H2O with 100 moles $CaCO_3$). The workaround for this case, if all you want is water in the makeup stream, is to make a dummy stream of pure solid $CaCO_3$ and put it into the blend with a factor of whatever is necessary (-100, for example) to get rid of that component.

HCURVE

This unit works normally with Electrolytes. However, since electrolyte thermodynamic methods do not give you heat capacity or entropy, those properties will be missing in the reports. The pseudocritical and petro reports are available, but are unlikely to give meaningful quantities for electrolyte streams.

Phase Envelope

The Phase Envelope is available only with SRK and PR thermodynamic methods and is therefore disallowed with Electrolytes.

Hydrates

The Hydrates unit does not have a method set associated with it; therefore, it may be used with a stream exiting an electrolyte unit.

BVLE

The BVLE unit may be used with Electrolyte thermodynamic methods, provided water is one of the two components chosen for the unit's calculations. You must be careful not to specify the BVLE unit such that it goes into a composition region where the flash will not solve because there is too little water. It is probably preferable to do BVLE-type analysis using the electrolyte flash capabilities in the *Chemistry Wizard* software available from OLI, Inc.

Hextables

The Hextables utility is not yet available with Electrolyte thermodynamic methods. The controllers, the optimizer, and recycle data all work normally with electrolytes. See above, however, for comments on what parameters can be used in SPEC and DEFINE statements. With controllers and the optimizer, one must be careful not to allow a VARY statement to create a condition where OLI cannot solve a flash.

Case Study, Restart, Interactive

These options work exactly the same for Electrolyte problems as for other problems.

Depressuring Unit

This unit is currently disallowed with Electrolytes.

Considerations Regarding Transport Properties

Currently, all transport property methods are allowed through keywords with PRO/II Electrolytes (though most electrolyte models will have at least one component for which TRAPP is not valid). Vapor-phase transport methods should be chosen without regard to the electrolyte nature of the problem; PETRO would typically be the best option. For aqueous electrolyte streams, the recommended method for the liquid phase is PURE, since the other methods in PRO/II are primarily intended for hydrocarbon streams. The resulting transport properties will, for most liquid streams, be very close to the values for pure water.

Output Considerations

To see what is going on in Electrolyte flowsheets, the user should always use at least the PART setting (which is now the default) on the PRINT ION flag. The user stream report writer works normally with Electrolytes. There are, of course, several quantities, such as heat capacities and entropies, which are not available. Several others, such as RVP and GHV, will not be meaningful for electrolyte streams. In addition, quantities such as NBP and TR, which depend on averages of pure-component numbers, may not be meaningful because some of the components in an electrolyte problem are likely to be salts, for which (for example) there is no NBP or TC in the library. For some of these numbers, dummy numbers (usually water values) are "filled in" in the library to satisfy PRO/II input checking; averages using these numbers will not be very meaningful.

Speed Considerations

Compared to most other standard PRO/II thermodynamic methods, Electrolytes is slow. This is unavoidable because it has to solve chemical equilibria simultaneously with phase equilibria. For a problem with the same number of components, Electrolytes will typically be slower than a method like SRKM or NRTL by one or two orders of magnitude.

One way to help the speed factor is to use the smallest model possible that includes all the components in the problem. It would probably be a good investment (especially if the model is going to be used many times) to generate in a small model (for example, four components) if the alternative were to use a much larger pre-generated model. Problems also run faster in "no-solids" models, and so these versions should be used if it is known that there is no chance of solid precipitation.

Because of the time that may be required for solution, it becomes more worthwhile to provide good initial guesses. Results from earlier runs may be saved and used in subsequent problems; the `PRINT PROFILE` option on the Column is particularly useful for this purpose. While saving profiles may not be worth the trouble for normal columns, it can be quite worthwhile for an electrolyte column.

Index