Project OpSim – Specifications Document

Version 1.0.3

Preamble

The present document establishes specifications for Open Source Process Simulator (OpSim) project. The effort of development will be segmented in three major areas related to corresponding aspects of the simulator: the user interface, the engine and the data access.

This document is opened for insertions from our development community and will be included in the code repository HTML format, in order that concurrent modifications become feasible.

Index

1 What is new?

3/17/2006

1) Updated specs for database files (Samuel)

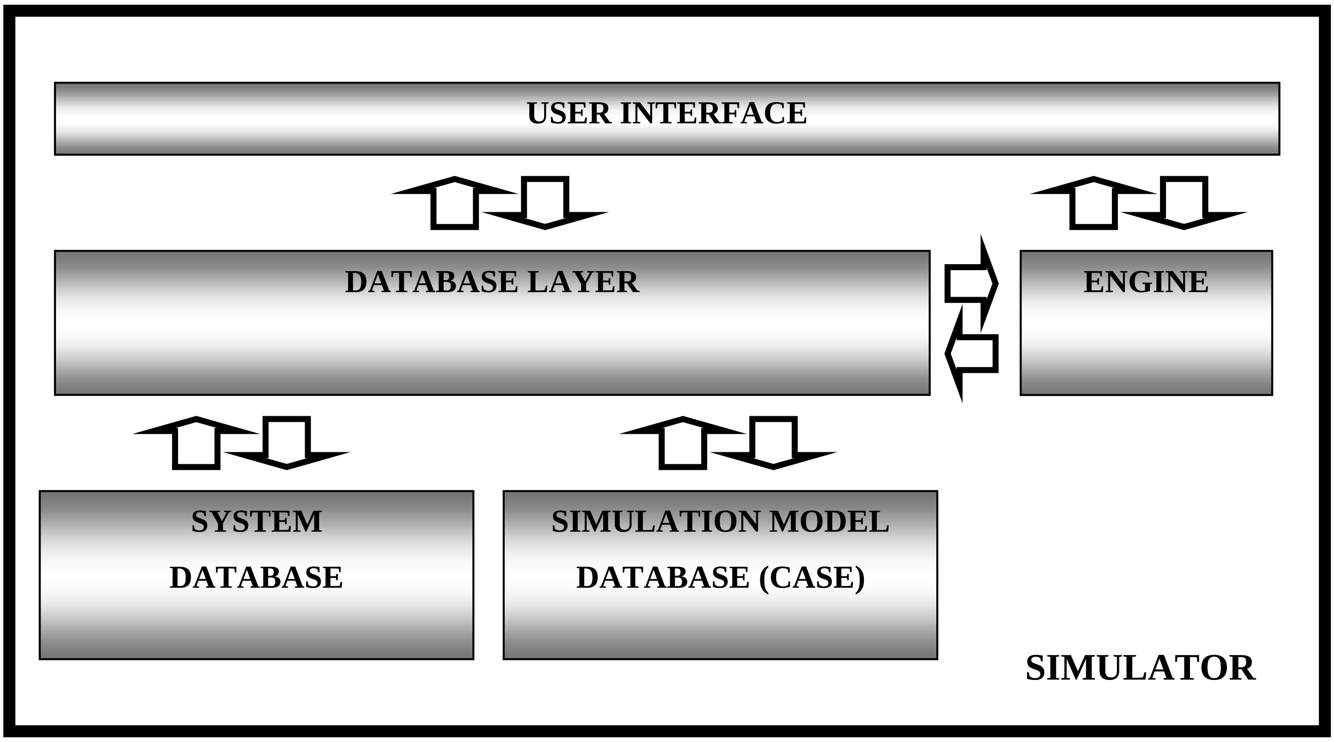
10/03/2006

1) Added schedule proposal (from Hazem).
2) Added specs (from Rahul Anantharaman).
3) Added structure proposal (from Hazem).

2 Overall Architecture

1) The whole simulator structure is divided in three major parts: user interface, engine and data accesses layer. Each one aspect has its own functionality and provides services to each other, through precisely defined boundaries. They exchange information only through these boundaries.

2) A diagram illustrating this is shown below:



3 Interface

3) The simulator will have an intuitive and friendly graphic user interface (GUI). No development effort is going to be spent in implementing a command line. Interaction with the simulation engine will be provided only through the GUI. An immediate consequence of this approach is that, at the first time, external applications will have no means to use the simulation engine (what would be possible with a command line user interface, using the standard I/O pipes). However there is a remedy to this shortcoming, which does not suffer with the poor performance of communication processes via the use of pipes (see section Engine).

4) Fundamental elements of the interface are:

5) Component Pallet for unit operations and other components.

6) Workspace to build the flow sheet. The Workspace has a drag-drop style, so that elements of the simulation model can be point-clicked and dragged from the Component Pallet

7) Report Forms output module.

8) Components of a simulation case are defined in the graphical environment through forms. A list of input forms is shown below:

COMPONENT	DESCRIPTION
Stream	Process stream between two unit ops equipment.
Two Phase Separator	
Three Phase Separator	
Compressor	
Valve	
Pipe	
Heat exchanger	
Distillation Column	
Pump	
Tank	
Heater	
Cooler	
?	
?	
?	

4 Engine

1) Simulation engine will be steady state. Future version may leverage the models to be transient.

2) Simulation engine is closely linked to the user interface, though completely impendent of it. This goal may be achieved through an interfacing layer between both.

3) With this partition, in a later stage of development, the engine may be packaged in a library (dynamic link library or shared object on Linux) and its interfaces exposed to the system environment. This design permits other applications to direct link and use the simulation engine, completely new user interfaces may be design even using another programming languages.

4) The engine comprises basically:

   4.1) Solver.

   4.2) Mathematical models for each unit operation or simulation component.

   4.3) Thermodynamic models.

5) Accessing the database is not a functionality of the engine. It does nothing about the database structure. The data needed to solve the simulation are supplied to the engine by other parts of the simulator. After solving the simulation case, the engine supplies the rest of the simulator (interface and database layer) with the partial or final results.

5 Data Access

1) The simulator will have input/output (I/O) capabilities to save and reload a simulation case from a file.

2) All the data comprising one simulation case is stored in only one file.

3) The file I/O operation will be performed through the embedded Relational Database Management System (DBMS): Firebird.

4) The technical reasoning to use a DBMS is:

   4.1) Low level storage operations are isolated from the simulator.

4.2) We do not need to worry about a file format. This is left to the DBMS. If the file format of the DBMS changes, we only need to dump the data and inject in the new version. A custom file format would require exhaustive implementation of converting routines.

4.3) Reliable data storage. DBMS uses sophisticated schemes to access the hard disk in order to protect data.

4.4) Fragmented access to the data file. A custom data file commonly imposes a three stage cycle: reading the file at once, and storing it in the RAM; modifying the data in RAM and writing the whole file at once. DBMS allows read/write small pieces of information in a much more efficient manner.

4.5) Concurrent access to the data in the simulation file. DBMS permits multiple applications accessing the very same simulation case, what is not allowed with a custom formatted file. A simulation case could be built simultaneously by many individuals remotely apart, via the internet, for example, or a local/wide area network (LAN/WAN).

5) The technical reasoning to use a Firebird is:

5.1) Firebird is an industrial strength DBMS with decades of corporate use.

5.2) Firebird is open source, free of charge, with a very proud active community. The licensing agreement even allows its use with commercial applications, without restrictions (LGPL like).

5.3) Can be easily embedded into the simulator.

5.4) Can be easily scaled up to client/server, allowing simultaneous access of multiple users.

5.5) Available in several operating systems.

5.6) Have a very small footprint and the most advantageous relation [feature set]/[footprint].

5.7) Very fine integration support with Lazarus. Some Delphi components are directly compatible with Lazarus, but regarding the best one (IBObjects), I am not so sure. Anyway, there are native components for a data access in Lazarus.

5.1 Database Structure

1) The data utilized by the simulator is stored in two databases:

1.1) System Database: SYSDATA.ODB, used to store intrinsic to the whole system of the simulator, e.g. preferences, environment conjurations, etc. The extension “ODB” stands for “Opsim DataBase”

1.2) Model Database: [model-name].OSC, that store the whole data of the simulation model (case) currently opened at the simulator environment. The extension “OSC” stands for “Opsim Simulation Case”

5.1.1. The System Database

The system database is comprised by the following tables:

1) Units of measures used in conversions.

2) Sets of units.

3) What more???????

1. UNITS OF MEASUREMENT: holds data regarding the units of measurement used in the simulator. This table will be used to build custom sets of units.

PHISICAL TABLE NAME: UNIT_MEASURE			
FIELD NAME	**FIELD DESCRIPTION**	**FIELD TYPE**	**FIELD SIZE (BYTES)**
ID	Internal identification used in referential integrity of the data.	INTEGER (AUTO INCREMENT)	
NAME	Name of the unit.	TEXT	70
SYMBOL	Symbol of the unit in normalized text (e.g. W/m2K or W/m^2.K)	TEXT	30
CONV_FACTOR_INTERSECT_SI	Intersect conversion factor to the metric system of measurement.	TEXT	100
CONV_FACTOR_SLOPE_SI	Slope conversion factor to the metric system of	TEXT	100

	measurement.		
SOMETHING MORE??????????			

2. UNIT SETS OF MEASUREMENT: hold unit sets of measurement. A set of units should be used while building the simulation model. Each unit of measurement contained in the set, gets an entry on this table.

PHISICAL TABLE NAME: UNIT_MEASURE_SET			
FIELD NAME	**FIELD DESCRIPTION**	**FIELD TYPE**	**FIELD SIZE (BYTES)**
ID	Internal identification used in referential integrity of the data.	INTEGER (AUTO INCREMENT)	
NAME	Name of the unit set (e.g. SI, EuroSI, English System, etc.)	TEXT	50
UNIT_MEASURE_ID	ID of the unit of measurement.	INTEGER	
SOMETHING MORE??????????			

3. PROPERTIES OF PURE SUBSTANCES: holds data regarding the properties of the pure substances.

PHISICAL TABLE NAME: PROP_PURE_SUBST			
FIELD NAME	**FIELD DESCRIPTION**	**FIELD TYPE**	**FIELD SIZE (BYTES)**
ID	Internal identification used in referential integrity of the data.	INTEGER (AUTO INCREMENT)	
CODE	This is a custom code that may be show to the user. It is not used to create relationship between tables of the database.	INTEGER	

NAME	Name of the substance.	TEXT	70
NAME_IUPAC	IUPAC name o the substance.	TEXT	100
CHEMICAL_FORMULA	The chemical formula of the substance (e.g. CH4)	TEXT	100
MOLECULAR_WEIGHT	The molecular weight.	REAL	
TC	Critical Temperature (K)	REAL	
PC	Critical Pressure (Pa)	REAL	
VC	Critical Volume (m3/kmol)	REAL	
API	API gravity	REAL	
ACC	Accentic Factor	REAL	
NBP	Normal Boiling Point (K)	REAL	
VNB	Viscosity at NBP (kgf.hr/m2)	REAL	
HF	Ideal Gas Heat of Formation at Standard temperature (J/kg)	REAL	
GF	Ideal Gas Free Energy of Formation at Standard Temperature (J/kg)	REAL	
IG_COEFF_1	Coefficient 1 to calculate Ideal Gas Thermodynamic properties (J/kg)	REAL	
IG_COEFF_2	Coefficient 2 to calculate Ideal Gas Thermodynamic properties (J/kg/K)	REAL	
IG_COEFF_3	Coefficient 3 to calculate Ideal Gas Thermodynamic properties (J/kg/K2)	REAL	
IG_COEFF_4	Coefficient 4 to calculate Ideal Gas Thermodynamic properties (J/kg/K3)	REAL	
IG_COEFF_5	Coefficient 5 to calculate	REAL	

	Ideal Gas Thermodynamic properties (J/kg/K4)		
IG_COEFF_6	Coefficient 6 to calculate Ideal Gas Thermodynamic properties (J/kg/K5)	REAL	
IG_COEFF_7	Coefficient 7 to calculate Ideal Gas Thermodynamic properties (J/kg/K6)	REAL	
SOMETHING MORE??????????			

5.1.2. The Model Database

The model database contains:

1) Global options of the simulation model.

2) One different table for each type of component added to the model, being it a unit operation, a stream, or whatever.

3) Other tables??????

4. PUMPS ON THE SIMULATION MODEL: holds data regarding all the pumps components used in the simulation model. Each pump placed by the user on the PFD diagram, generates a record on this table. Since we may need other tables to keep other others unit operation or components, let establish a prefix now on "PFD_" used to name each table containing unit operation/component of a simulation case.

PHISICAL TABLE NAME: PFD_PUMP			
FIELD NAME	**FIELD DESCRIPTION**	**FIELD TYPE**	**FIELD SIZE (BYTES)**
ID	Internal identification used in referential integrity of the data.	INTEGER (AUTO INCREMENT)	
CODE	This is a custom code that may be shown to	INTEGER	

	the user. It is not used to create relationship between tables of the database.		
NAME	Descriptive name of the pump.	TEXT	70
LABEL	Label used to identification on the PFD.	TEXT	50
HEAD	The head supported by the pump with zero flow rate	REAL	
HEAD_UNIT_ID	ID of the unit of measurement used to specify the HEAD field. Needed to do conversions.	INTEGER	

6 CAPE-OPEN STANDARDS

6.1.1. Standardized Models or Equations (CAPE-OPEN)

This Appendix contains a selection of model equations which may be part of a PPDB. This selection is not meant to be exclusive, every PPDB can add its own equations.

Predictive equations like UNIFAC are exclude on purpose, because they do not need any model parameters.

short name of method	**full name of method**	**equation**	**parameters**
Antoine	Antoine vapor pressure equation	$\log(p) = a - b/(T+c)$	a, b, c
Wrede	Wrede vapor pressure equation	$\log(p) = a - b/T$	a, b
Wrede-ln	Wrede vapor pressure equation	$\ln(p) = a - b/T$	a, b
Cragoe	Cragoe vapor pressure equation	$\log(p) = a + b/T + c*T + d*T^2$	a, b, c, d

short name of method	full name of method	equation	parameters
Riedel	Riedel vapor pressure equation	$\ln(p) = a - b/T + c*T + d*T^2 + e*\ln(T)$	a, b, c, d, e
Wagner	Wagner vapor pressure equation	$\ln(p/p_{crit}) = (a*x + b*x^{(3/2)} + c*x^3 + d*x^6)/(T/T_{crit})$; $x = 1 - T/T_{crit}$	a, b, c, d, criticalPressure, criticalTemperature
Wagner2	2nd Wagner vapor pressure equation	$\ln(p/p_{crit}) = (a*x + b*x^{(3/2)} + c*x^3 + d*x^7 + e*x^9)/(T/T_{crit})$; $x = 1 - T/T_{crit}$	a, b, c, d, e, criticalPressure, criticalTemperature
Wagner3	Wagner vapor pressure equation (Aspen)	$\ln(p/p_{crit}) = (a*x + b*x^{(3/2)} + c*x^3 + d*x^4)/(T/T_{crit})$; $x = 1 - T/T_{crit}$	a, b, c, d, criticalPressure, criticalTemperature
Chebyshev	Chebyshev vapor pressure equation	$T*\log(p) = c_0/2 + \sum_{(s)}[c_s * E_s(x)]$; $x = 2*T - (T_{max}+T_{min})/(T_{max}-T_{min})$; $E_s(x)$ = Chebyshev polynomial of order s	c_0, c_1, c_2, ..., T_min, T_max
polynomial	polynomial	$y = a + b*x + c*x^2 + ...+ j*x^9$ x = any property	a, b, c, d, e, f, g, h, i, j
vapor pressure_1	vapor pressure equation	$\ln(p) = a + b*T + c/T + d/T^2$	a, b, c, d
mod.Antoine(Hysys)	modified Antoine vapor pressure equation (Hysys[9], page A-36)	$\ln(p) = A + B/(T+C) + D*T + E*\ln(T) + F*T^G$	A, B, C, D, E, F, G
mod.Antoine(Aspen)	modified Antoine vapor pressure equation (Aspen[7], page 3-80)	$\ln(p) = A + B/(T+C) + D*\ln(T) + E*T^F$	A, B, C, D, E, F
Jones-Dole	Jones-Dole equation	$\eta/\eta_0 = 1 + a*\sqrt{c} + b*c$	a, b, viscosity_0
Yen-Woods	Yen-Woods equation for densities	$d = d_{crit} * (a + \sum_{(j)}(k_j)*(1T/T_{crit})^{(j/3)})$	criticalDensity, criticalTemperature, a, k_0, k_1, k_2, ...
Antoine viscosity	Antoine equation for the viscosity	$\ln(\eta) = a + b/(T+c)$	a, b, c
Riedel therm.cond.	Riedel equation for thermal conductivities	$\kappa = a * (1 + (20/3)*(1 - T/T_{crit})^{(2/3)})$	a, criticalTemperature
Sprow/Prausnitz	Surface Tension after Sprow and Prausnitz	$\sigma = a * (1 - T/T_{crit})^b$	a, b, criticalTemperature
modified polynomial	modified polynomial	$property = a + b/T + c/T^2 + d*T + e*T^2 + f*T^3 + ...$	a, b, c, d, e, f, ...
Yuan/Mok	Yuan - Mok equation for the heat capacity	$c_p = a + b * \exp(-c/T^n)$	a, b, c, n
Redlich-Kister	Redlich-Kister equation for excess properties in binary systems	Δ property $= x_1 * x_2 * \Sigma_{(i)}(a_i*(x_1 x_2)^{i})$	a_0, a_1, a_2, a_3, ...
thermal conductivity (NEL)	NEL equation for thermal conductivity	$\kappa = a*(1 + b*x^{(1/3)} + c*x^{(2/3)} + d*x)$; $x=1-T/T_{crit}$	a, b, c, d, criticalTemperature

short name of method	**full name of method**	**equation**	**parameters**
virial equation	virial equation	$Z = 1 + vc_2*p + vc_3*p_2 + vc_4*p_4 + ...$	vc_2, vc_3, vc_4
BWR	BWR-equation of state	$p = R*T*d + (b_0*R*T -a_0\ c_0/T_2)*d_2 + (b_0*R*T - a_0)*d_3 +a*\alpha*d_6 + (c*d_3/T_2)*(1 + \gamma*d_2)*exp(-\gamma*d_2)$	a_0, b_0, c_0, a, b, c, alpha, gamma
BWR-Lee-Starling	Benedict-Webb-Rubin-Lee-Starling equation of state (Aspen[7], page 3-8)	$Z_m = Z_{m0} + \gamma_i*Z_{m1}$; Z_{m0}, Z_{m1} =function($T,T_{crit},v_m,v_{crit,m}$)	criticalTemperature _i, criticalVolume_i, gamma_i, epsilon_i_j, eta_i_j
Hayden-O'Connel	Hayden-O'Connel equation of state (Aspen[7], page 3-9)	$Z_m = 1 + B*p/R*T$; $B = \Sigma_{(i)}\Sigma_{(j)}B_{ij}(T)$	B_i_j
Lee-Kesler	Lee-Kesler equation of state (Aspen[7], page 3-18)	$Z = Z_0 + (Z_r-Z_0)*\omega/\omega_r$ $Z_0 = fct_0(T/T_{crit}, p/p_{rit})$ $Z_r = fct_r(T/T_{crit}, p/p_{crit})$	criticalTemperature, criticalPressure, omega
Lee-Kesler-Plöcker	Lee-Kesler-Plöcker equation of state (Aspen[7], page 3-19)	$Z_m = Z_{m0} + (\omega/\omega_r)*(Z_{m0} - Z_{mr})$; $Z_{m0} = fct_0(T,T_{crit}, v_m, v_{crit,m})$; $Z_{mr} = fct_f(T,T_{crit},v_m,v_{crit,m})$ mixing rules for $v_{crit,m}$, T_{crit}	criticalTemperature, criticalPressure, vcriticalVolume, omega, Z_c_i, K_i_j
Peng-Robinson-Boston-Mathias	Peng-Robinson-Boston-Mathias equation of state (Aspen[7], page 325)	$p = R*T/(v_m-b)\ a/[v_m*(v_m+b)+b*(v_m-b)]$	criticalTemperature_i, criticalPressure_i, omega_i, k_1_2
Redlich-Kwong	Redlich-Kwong equation of state (Aspen[7], page 3-27)	$p = R*T/(v_m-b)\ (a/\sqrt{(T)})/[v_m*(v_m+b)]$	criticalTemperature_i, criticalPressure_i
Redlich-Kwong-Aspen	Aspen modification of the Redlich-Kwong equation of state(Aspen[7], page 3-28)	$p = R*T/(v_m-b) - a/[v_m*(v_m+b)]$ with mixing rules	criticalTemperature_i, criticalPressure_i, omega_i, eta_i,k_0_a__i_j, k_1_a_i_j, k_0_D_i_j, k_1_D_i_j,
Redlich-Kwong-Soave-Boston-Mathias	Redlich-Kwong equation of state with Boston-Mathias alpha function (Aspen[7], page 3-29)	$p = R*T/(v_m-b) - a/[v_m*(v_m+b)]$ with mixing rules	criticalTemperature_i, criticalPressure_i, omega_i, k_i_j
Schwartzentruber-Renon	Schwartzentruber-Renon equation of state (Aspen[7], page 3-31)	$p = R*T/(V_m+c-b)\ a/[(v_m+c)*(V_m+c+b)]$ with mixing rules	criticalTemperature_i, criticalPressure_i, omega_i, q_0_i, q_1_i, q_2_i, c_0_i, c_1_i, c_2_i, k_0_a_i-j, k_1_a_i_j, k_2_a_i_j, l_0_i_j, l_1_i_j, l_2_i_j, k_0_D_i_j, k_1_D_i_j, k_2_D_i_j
Peng-Robinson	standard Peng-Robinson equation of state (Aspen[7], page 3-34)	$p = R*T/(v_m-b)\ a/[v_m*(v_m+b)+b*(v_m-b)]$	criticalTemperature_i, criticalPressure_i, omega_i (i=1..2), k_1_2
Redlich-Kwong-Soave	standard Redlich-Kwong-Soave equation of state (Aspen[7], page 335)	$p = R*T/(v_m-b) - a/[v_m*(v_m+b)]$ with mixing rules	criticalTemperature_i, criticalPressure_i, omega_i, k_i_j

short name of method	full name of method	equation	parameters
Bromley-Pitzer	Bromley-Pitzer activity coefficient model (Aspen[7], page 3-54)		beta_ion, delta_ion, beta_0, beta_1, beta_2, beta_3
Chien-Null	Chien-Null model for calculation activity coefficient of highly non-ideal systems (Aspen[7], page 3-55)		a_i_j, b_i_j, v_i_j
Electrolyte-NRTL	NRTL activity coefficient model for electrolytes(Aspen[7], page 3-58)		A_B, B_B, C_B, r_i, A_BB, A_BsB, B_BBs, B_BsB, alpha_BB, F_BBs, F_BsB, G_BBs, G_BsB, C_ca_B, C_B_ca, D_ca_B, D_B_ca, E_ca_B, E_B-ca, alpha_ca_B, C_cas_cass, C_cass-cas, c_csa,cssa, C_cssa_csa, D_cas_cass,D_cass_cas, D_csa_cssa, D_cssa_csa, E_cas_cass, E_cass_cas,E_csa_cssa, E_cssa_csa, alpha_cas_cass, alpha_csa_cssa
NRTL	NRTL activity coefficient model (DDB[8], page XVI)		A_i_j, A_j_i, alpha (i,j=1...2)
extended NRTL (Aspen)	NRTL activity coefficient model (Aspen[7], page 3-62)		a_i_j, b_i_j, c_i_j, d_i_j, e_i_j, f_i_j (i,j=1...2)
general NRTL	general NRTL activity coefficient model (Hysys[9], page A-22)		form-of_equation, A_j_j, B_i_j, C_i_j, F_i_j, G_i_j, alpha1_i_j, alpha2_i_j (i,j=1...2)
Pitzer activity	Pitzer model for activity		beta_0, beta_1, beta_2,
coefficient model	coefficients of aqueous systems (Aspen [7], page 3-63)		beta_3, C_p, theta_c_cs, theta_a_as, psi_c_cs_a, psi_c_a_as
Redlich-Kister	Redlich-Kister model for calculating activity coefficients (Aspen[7], page 3-66)		a_i_j, b_i_j, c_i_j, d_i_j, e_i_j, f_i_j, g_i_j, h_i_j, m_i_j, n_i_j, v_i
Scatchard-Hildebrand	Scatchard-Hildebrand model (Aspen[7], page 3-67)		criticalTemperature-i, delta_i, V_i_CVT, V_i_l
Margules	Margules equation for calculating liquid activity coefficients (DDB[8], page XVI)	$\ln(\gamma_i) = [A_{ij} + 2*(A_{ji}-A_{ij})*x_i](1-x_i)^2$	A_i_j
extended Margules	Margules equation for calculating liquid activity coefficients with temperature-independent parameters (Hysys[9], page A-24)	$\ln(\gamma_i) = (1-x_i)*+2*[A_i + 2+x_i*(B_i-A_i)]$; $A_i = \Sigma_{(j)}[x_j*(a_{ij}+b_{ij}*T)/(1-x_i)]$; $B_i = \Sigma_{(j)}[x_j*(a_{ji}+b_{ji}*T)/(1-x_i)]$	a_i_j, b_i_j (i,j=1...2)
three-suffix Margules	extended Margules equation for calculating liquid activity coefficients (Aspen[7], page 3-68)		a_i_j, b_i_j, c_i_j, d_i_j (i,j=1...2)

short name of method	full name of method	equation	parameters
van Laar	van Laar equation for calculating liquid activity coefficients (DDB[8], page XVI)		A_i_j (i,j=1...2)
extended van Laar (Aspen)	extended van Laar equation for calculating liquid activity coefficients with temperature-independent parameters (Aspen[7], page 3-75)		a_i_j, b_i_j, c_i_j, d_i_j (i,j=1...2)
extended van Laar (Hysys)	extended van Laar equation for calculating liquid activity coefficients with temperature-independent parameters (Hysys[9], page A-28)		a_i_j, b_i_j (i,j=1...2)
Wilson	Wilson equation for calculating liquid activity coefficients (DDB[8], page XVI)		A_i_j (i,j=1...2)
extended Wilson (Aspen)	extended Wilson equation for calculating liquid activity (Aspen[7], page 3-78)		a_i_j, b_i_j, c_i_j, d_i_j (i,j=1...2)
extended Wilson (Hysys)	extended Wilson equation for calculating liquid activity coefficients with temperature-independent parameters (Hysys[9], page A-29)		a_i_j, b_i_j (i,j=1...2)
UNIQUAC	UNIQUAC equation for calculating liquid activity coefficients (DDB[8], page XVII)		u_i_j (i,j=1...2)
extended UNIQUAC (Aspen)	extended UNIQUAC equation for calculating liquid activity coefficients with temperature-independent parameters (Aspen[7], page 3-74)		a_i_j, b_i_j, c_i_j, d_i_j (i,j=1...2)
extended UNIQUAC (Hysys)	extended UNIQUAC equation for calculating liquid activity coefficients with temperature-independent parameters (Hysys[9], page A-26)		a_i_j, b_i_j (i,j=1...2)
DIPPR107	DIPPR equation for the ideal heat capacity	$property = A + B[C/T/\sinh(C/T)]^2 + D[E/T/\cosh(E/T)]^2$	A, B, C, D, E
heat capacity (ASPEN)	Aspen[7]-equation for the solid heat capacity (page 3-102)	$C_p = c_1 + c_2{*}T + c_3{*}T^2 + c_4/T + c_5/T^2 + c_6/\sqrt{(T)}$	c1, c2, c3, c4, c5, c6
Barin	Barin equations for thermophysical property data	$G = a + b{*}T + c{*}(T{*}\ln(T)) + d{*}T^2 + e{*}T^3 + f{*}T^4 + g/T + h/T^2$	a, b, c, d, e,f, g, h
Andrade	Andrade equation for calculating the liquid viscosity	$\ln(\eta) = A + B/T + C{*}\ln(T)$	A, B, C
liquid viscosity (DIPPR)	DIPPR equation for the liquid viscosity	$\ln(\eta) = c_1 + c_2/T + c_3{*}\ln(T) + c_4{*}T^{c_5}$	c1, c2, c3, c4, c5

short name of method	**full name of method**	**equation**	**parameters**
viscosity mixing rule	ASPEN[7] mixing rule for the liquid viscosity (listed under the heading Andrade/DIPPR, page 3122)	$\ln(\eta) = \Sigma_{(i)}[x\_i*\ln(\eta_i)] + \Sigma_{(i,j)}[(a_{ij} + b_{ij}/T)*x_i*x_j + (c_{ij}+d_{ij}/T)*x_i^2*x_j^2]$	a_i_j, b_i_j, c_i_j, d_i_j
DIPPR102	DIPPR equation for the gas viscosity at 0 atm pressure and the gas thermal conductivity	$property = A*T^B/(1 + C/T + D/T^2)$	A, B, C, D
Chung-Lee-Starling	Chung-Lee-Starling correlation of the viscosity and thermal conductivity of liquid or gaseous mixtures (Aspen[7], page 3-127, 3138))		criticalTemperature_i, V_crit_i, dipole_moment_i, omega_i, kappa_i, xi_i_j, zeta_i_j
suface tension (DIPPR)	DIPPR correlation for surface tension	$\sigma = c1*(1-T_r)$^$(c_2 + c_3*T_r + c_4*T_r^2 + c_5*T_r^3)$; $T_r = T/T_{crit}$	c1, c2, c3, c4, c5, criticalTemperature
Hakim-Steinberg-Stiel	Hakim-Steinberg-Stiel equation for the surface tension (Aspen[7], page 3-155)		chi
DIPPR105		$property = A/B$^$[1+(1-T/C)^D]$	A, B, C, D
DIPPR101		$property = \exp(A + B/T + C*\ln(T) + D*T^E)$	A, B, C, D, E
DIPPR106		$property = A*(1-T_r)$^$(B + C*T_r + D*T_r^2)$; $T_r = T/T_{crrit}$	A,B,C,D, criticalTemperature
DIPPR104		$property = A + B/T + C/T^3 + D/T^8 + E/T^9$	A, B, C, D, E

Glossary of the symbols used in the column "equation"

cp heat capacity d density

d_{crit} critical density p

pressure p_{crit} critical pressure

R gas constant T

temperature T_{crit} critical

temperature v_m volume of a

mixture x_1 mole fraction of

compound 1 Z

compressibility factor Z_m

compressibility factor of a

mixture γ_i activity

coefficient of compound i

κthermal conductivity

ηviscosity η_0 viscosity at

zero concentration σsurface

tension

6.1.2. Standardized Units (CAPE-OPEN)

unit	**SI-unit**	**remarks**	**factor unit -> SI-unit (only for information)**
		no dimension	1.
rad			0.15915475
(BTU.ft3)^(1/2)	(J.m3)^(1/2)		5.4658892
(kJ.m3)^(1/2)	(J.m3)^(1/2)		31.6227766
(J/cm3)^(1/2)	(J/c3)^(1/2)		31.6227766
(BTU/ft3)^(1/2)	(J/m3)^(1/2)		1.9302605E2
(cal/cm3)^(1/2)	(J/m3)^(1/2)		2046.1671
(J/m3)^(1/2)	(J/m3)^(1/2)		1.
(kcal/m3)^(1/2)	(J/m3)^(1/2)		64.70548
(g.l)^(1/2)/min	(kg.m3)^(1/2)/s		1.6666667E-5
(kg.m3)^(1/2)/s	(kg.m3)^(1/2)/s		1.
(lb.ft3)^(1/2)/hr	(kg.m3)^(1/2)/s		3.1481311E-5
(lb.gal)^(1/2)/min	(kg.m3)^(1/2)/s		6.9061846E-4
bar.m6/mol2	Pa.m6/mol2		1.E5
bar.m9.K2/mol3	Pa.m9.K2/mol3		1.E5
bar.m9/mol3	Pa.m9/mol3		1.E5
A	A	Ampere	1.
mA	A		1.E-3
A/cm2	A/m2		10000.
A/m2	A/m2		1.
mA/cm2	A/m2		10.
mA/m2	A/m2		0.001

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
amagat	amagat		1.
atomic %	atomic fraction		0.01
atomic fraction	atomic fraction		1.
Debye	Coul.m		3.33564E-30
Coul	Coul		1.
Coul.m	Coul.m		1.
Coul/mol	Coul/mol		1.
Farad	Farad		1.
mFarad	Farad		0.001
nFarad	Farad		1.E-9
pFarad	Farad		1.E-12
Farad/m	Farad/m		1.
oz	g		28.349523
H	H		1.
H/m	H/m		1.
hr-1	Hz		2.7777778E-4
Hz	Hz		1.
min-1	Hz		1.6666667E-2
s-1	Hz		1.
BTU	J	Britsh thermal unit	1055.0559
cal	J		4.1868
erg	J		1.E-7
eV	J		1.6021892E-19
GJ	J		1.00E9
J	J		1.
kcal	J		4.1868E3
kJ	J		1.E3
kp.m	J		9.80665
kW.hr	J		3.6E6
mJ	J		0.001
MMBTU	J	million BTUs	1.0550559E9
MMkcal	J	million kilocalories	4.1868E9
erg.mK/cm3	J.K/m3		0.1
J.K/m3	J.K/m3		1.
kJ/kg.degC	J.kg.K		1.0E3
J.s	J.s		1.

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
J/Hz	J.s		1.
J.s/mol	J.s/mol		1.
BTU/cycle	J/cycle		1054.35
cal/cycle	J/cycle		4.1868
GJ/cycle	J/cycle		1.0E9
J/cycle	J/cycle		1.
kcal/cycle	J/cycle		1000.
MMBTU/cycle	J/cycle		1.0550559E9
MMkcal/cycle	J/cycle		4.1868E9
J/kg.degC	J/g.K		1.E-3
J/kg.K	J/g.K		1.E-3
kcal/g.degC	J/g.K		4.1868E3
BTU/degF	J/K	British thermal unit per degrees Fahrenheit	1.899101E3
cal/K	J/K		4.1868
J/K	J/K		1.
kcal/K	J/K		4.1868E3
kJ/K	J/K		1.E3
kcal/degC.hr	J/K.s		1.163
BTU/lb	J/kg	Britsh thermal unit per pound avdp.	2.32600E3
cal/g	J/kg		4.1868E3
cal/g	J/kg		4186.8
cal/kg	J/kg		4.1868
J/g	J/kg		1.E3
J/kg	J/kg		1.
kcal(th)/kg	J/kg		4184.
kcal/g	J/kg		4186.8E3
kcal/kg	J/kg		4186.8
kJ/g	J/kg		1.E6
kJ/g	J/kg		1.E6
kJ/kg	J/kg		1.0E3
kW.hr/ton	J/kg		3600.
mJ/kg	J/kg		0.001
MJ/kg	J/kg		1.E6
MMBTU/lb	J/kg	million BTU per pound (avdp)	2.32444E9
MMkcal/kg	J/kg		4.1868E9

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
BTU/lb.degF	J/kg.K		4.1868E3
BTU/lb.Rnk	J/kg.K		4.1868E3
cal(th)/g.K	J/kg.K		4148.
cal/g.degC	J/kg.K		4.1868E3
cal/g.K	J/kg.K		4186.8
cal/kg.degC	J/kg.K		4.1868
cal/kg.K	J/kg.K		4.1868
J/g.degC	J/kg.K		1.E3
J/g.K	J/kg.K		1.E3
J/kg.K	J/kg.K		1.
kcal/g.K	J/kg.K		4186.8E3
kcal/kg.degC	J/kg.K		4186.8
kcal/kg.K	J/kg.K		4.1868E3
kcal/kg.K	J/kg.K		4186.8
kJ/g.degC	J/kg.K		1.E6
kJ/g.K	J/kg.K		1.E6
kJ/kg.K	J/kg.K		1000.
BTU/lb.Rnk2	J/kg.K2		7536.2403
J/kg.K2	J/kg.K2		1.
kcal/kg.K2	J/kg.K2		4186.8
kJ/g.K2	J/kg.K2		1.E6
kJ/kg.K2	J/kg.K2		1000.
BTU/lb.Rnk3	J/kg.K3		13565.2326
kcal/kg.K3	J/kg.K3		4186.8
kJ/g.K3	J/kg.K3		1.E6
kJ/kg.K3	J/kg.K3		1000.
BTU/lb.Rnk4	J/kg.K4		24417.4187
kcal/kg.K4	J/kg.K4		4186.8
kJ/g.K4	J/kg.K4		1.E6
kJ/kg.K4	J/kg.K4		1000.
BTU/lb.Rnk5	J/kg.K5		43951.3537
kcal/kg.K5	J/kg.K5		4186.8
kJ/g.K5	J/kg.K5		1.E6
kJ/kg.K5	J/kg.K5		1000.
J/g.bar	J/kg.Pa		0.01
J/kg.Pa	J/kg.Pa		1.

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
J/g.bar.K	J/kg.Pa.K		0.01
J/kg.Pa.K	J/kg.Pa.K		1.
J/g.bar2.K	J/kg.Pa2.K		1.E-7
J/kg.Pa2.K	J/kg.Pa2.K		1.
BTU/ft.hr	J/m.s		0.96150757
cal/m.s	J/m.s		4.1868
J/m.s	J/m.s		1.
MMBTU/hr.ft	J/m.s		0.96150757E6
BTU.ft/ft2.hr.Rnk	J/m.s.K		0.09495505
BTU/ft.hr.degF	J/m.s.K		1.730734744
cal.cm/s.cm2.K	J/m.s.K		418.68
kcal/m.hr.degC	J/m.s.K		1.163
MMBTU/hr.ft2.Rnk	J/m.s.K		5.6782633E6
cal/cm2	J/m2		4.1868E4
erg/cm2	J/m2		1.E-3
J/cm2	J/m2		1.E4
J/m2	J/m2		1.
kcal/m2	J/m2		4.1868E3
kJ/m2	J/m2		1.E3
BTU/ft2.hr.Rnk	J/m2.K		5.6782636
cal/cm2.K	J/m2.K		4.1868E4
cal/m2.K	J/m2.K		4.1868
J/cm2.K	J/m2.K		1.E4
J/m2.K	J/m2.K		1.
kcal/m2.K	J/m2.K		4.1868E3
kJ/m2.K	J/m2.K		1.E3
BTU/ft2.hr	J/m2.s		3.1545909
BTU/ft2.hr.degF	J/m2.s.K		5.6782636
J/m2.s.K	J/m2.s.K		1.0
BTU/ft3	J/m3		29.875856
cal/cm3	J/m3		4.1868E6
J/m3	J/m3		1.
kcal/m3	J/m3		4.1868E3
kJ/cm3	J/m3		1000.
kJ/m3	J/m3		1.0E3
BTU/ft3.Rnk	J/m3.K		53.776541
cal/cm3.K	J/m3.K		4.1868E6

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
J/cm3.K	J/m3.K		1000000.
J/m3.K	J/m3.K		1.
kcal/m3.K	J/m3.K		4.1868E3
kJ/m3.K	J/m3.K		1000.
BTU/lbmol	J/mol	British thermal units per pound-moles	2.326E3
BTU/mol	J/mol		1055.056
cal(th)/mol	J/mol		4.184
cal/kmol	J/mol		4.1868E3
cal/mol	J/mol		4.1868
GJ/kmol	J/mol		1.0E12
J/kmol	J/mol		0.001
J/mol	J/mol		1.
kcal(th)/mol	J/mol		4.184E3
kcal/kmol	J/mol		4.1868
kcal/mol	J/mol		4.1868E3
kJ/kmol	J/mol		1.0
kJ/mol	J/mol		1000.
MJ/kmol	J/mol		1000.
MMBTU/lbmol	J/mol		2.32444E6
MMkcal/mol	J/mol		4.1868E9
BTU/lbmol.degF	J/mol.K		4.1868
BTU/lbmol.Rnk	J/mol.K		4.1868
BTU/mol.F	J/mol.K		1.8991006E3
BTU/mol.Rnk	J/mol.K		1.8991006E3
cal(th)/mol.K	J/mol.K		4.184
cal/kmol.degC	J/mol.K		4.1868E3
cal/kmol.K	J/mol.K		4.1868E3
cal/mol.degC	J/mol.K		4.1868
cal/mol.K	J/mol.K		4.1868
cal/mol.K	J/mol.K		4.1868
J/kmol.degC	J/mol.K		0.001
J/kmol.K	J/mol.K		0.001
J/mol.degC	J/mol.K		1.
J/mol.K	J/mol.K		1
kcal/kmol.degC	J/mol.K		4.1868
kcal/kmol.K	J/mol.K		4.1868

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
kcal/mol.degC	J/mol.K		4.1868E3
kcal/mol.K	J/mol.K		4.1868E3
kJ/kmol.degC	J/mol.K		1.
kJ/kmol.K	J/mol.K		1.
kJ/mol.degC	J/mol.K		1000.
kJ/mol.K	J/mol.K		1.E3
BTU/lbmol.Rnk2	J/mol.K2		7.53624
cal(th)/mol.K2	J/mol.K2		4.184
cal/mol.K2	J/mol.K2		4.1868
J/mol.K2	J/mol.K2		1.
kcal/kmol.K2	J/mol.K2		4.1868
kJ/kmol.K2	J/mol.K2		1.
kJ/mol.K2	J/mol.K2		1000.
cal(th)/mol.K3	J/mol.K3		4.184
cal/mol.K3	J/mol.K3		4.1868
J/mol.K3	J/mol.K3		1.
kJ/mol.K3	J/mol.K3		1000.
cal(th)/mol.K4	J/mol.K4		4.184
J/mol.K4	J/mol.K4		1.
cal/mol.atm	J/mol.Pa		4.13205E-5
J/mol.Pa	J/mol.Pa		1.
cal/mol.Torr.K	J/mol.Pa.K		0.0314035751
J/mol.Pa.K	J/mol.Pa.K		1.
BTU/hr	J/s		0.29307107
cal/hr	J/s		1.163E-3
cal/s	J/s		4.1868
GJ/hr	J/s		2.7777778E5
J/s	J/s		1.
kcal/hr	J/s		1.1629833
kJ/hr	J/s		0.2777777778
kJ/min	J/s		16.666667
kJ/s	J/s		1.0E3
MJ/hr	J/s		277.77778
MMBTU/day	J/s		1.221129458E4
MMBTU/hr	J/s		0.29307107E6
MMkcal/day	J/s		4.84576375E4
MMkcal/hr	J/s		1.1629833E6

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
BTU/hr.degF	J/s.K		0.52753056
BTU/hr.Rnk	J/s.K		.527527926
cal/s.K	J/s.K		4.1868
J/s.K	J/s.K		1.
kcal/hr.K	J/s.K		1.1629833
kcal/s.K	J/s.K		4186.8
kJ/hr.degC	J/s.K		0.27777778
kJ/s.degC	J/s.K		1.0E3
kJ/s.K	J/s.K		1.0E3
kcal/hr.m2	J/s.m2		1.1629833
kJ/hr.m2	J/s.m2		0.2777777778
kJ/s.m2	J/s.m2		1000.
kJ/s.m2.K	J/s.m2		1.0E3
kcal/hr.m2.degC	J/s.m2.K		1.1629833
kcal/hr.m2.K	J/s.m2.K		1.1629833
kcal/s.m2.K	J/s.m2.K		4186.8
kJ/hr.m2.degC	J/s.m2.K		0.2777777778
kJ/s.m2.degC	J/s.m2.K		1000.
degC	K	degrees centigrade	1. (+273.15)
degF	K		0.55555556 (+255.37222)
K	K		1.
kK	K		1000.
mK	K		0.001
Reamur	K		1.25 (+273.15)
Rnk	K		0.55555556
degF/psia	K/Pa		8.05764E-5
K/atm	K/Pa		9.86923E-6
K/bar	K/Pa		1.E-5
K/MPa	K/Pa		1.E-6
K/Pa	K/Pa		1.
degC-1	K-1		1.
degF-1	K-1		1.8
K-1	K-1		1.
Rnk-1	K-1		1.8
K2	K2		1.
degF2/lbmol2	K2/mol2		1.500111E-6
degF2/lbmol3	K2/mol3		3.3071795E-9

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
K2/Pa	K2/Pa		1.
K2/Torr	K2/Pa		7.500615E-3
K3	K3		1.
g	kg		0.001
kg	kg		1.
lb	kg	pound avdp.	0.45359237
Mlb	kg	1000 pounds (avdp)	453.59237
ton	kg		1000.
ton(long)	kg		1016.0469
ton(short)	kg		907.18474
kg.m2	kg.m2		1.
lb.in2	kg.m2		2.9263961E-4
g/cycle	kg/cycle		0.001
kg/cycle	kg/cycle		1.
lb/cycle	kg/cycle		0.45359237
Mlb/cycle	kg/cycle		453.59237
ton(short)/cycle	kg/cycle		907.18474
ton/cycle	kg/cycle		1000.
kg/J	kg/J		1.
g/100g solvent	kg/kg solvent		0.01
g/kg solvent	kg/kg solvent		0.001
kg/kg solvent	kg/kg solvent	kg solute per kg solvent	1.
mg/kg solvent	kg/kg solvent		1.E-6
kg/m	kg/m		1.
lb/ft	kg/m		1.4881639
kg/m.hr	kg/m.s		2.7777778E-4
slug/hr.ft	kg/m.s		0.013268619
kg/m.hr2	kg/m.s2		7.7160494E-8
kg/m.s2	kg/m.s2		1.
lb/in.s2	kg/m.s2		17.857967
lb/m.s2	kg/m.s2		0.45359237
g.cm2	kg/m2		10.
kg/cm2	kg/m2		1.E-4
kg/m2	kg/m2		1.
lb.ft2	kg/m2		0.04214
poundals/ft2	kg/m2		0.15175047
kg/m2.atm.hr	kg/m2.Pa.s		2.7414535E-9

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
lb/ft2.atm.hr	kg/m2.Pa.s		1.3384948E-8
g/cm2.s	kg/m2.s		10.
kg/m2.hr	kg/m2.s		2.7777778E-4
kg/m2.s	kg/m2.s		1.
lb/ft2.hr	kg/m2.s		1.3562298E-3
lb/ft2.s	kg/m2.s		4.8824276
g/cm2.s.atm	kg/m2.s.Pa		9.8692326E-5
g/cm2.s.Pa	kg/m2.s.Pa		10.
kg/m2.s.Pa	kg/m2.s.Pa		1.
g/100ml	kg/m3		10.
g/cm3	kg/m3		1000.
g/dm3	kg/m3		1.
g/l	kg/m3		1.
g/m3	kg/m3		1.E-3
g/ml	kg/m3		1.E3
kg/cm3	kg/m3		1.E6
kg/dm3	kg/m3		1.E3
kg/l	kg/m3		1.E3
kg/m3	kg/m3		1.
lb/ft3	kg/m3		16.018463
lb/gal	kg/m3		119.82643
lb/in3	kg/m3		27.6799E3
mg/cm3	kg/m3		1.
mg/l	kg/m3		1.E-3
mg/m3	kg/m3		1.E-6
g/100cm3 solvent	kg/m3 solvent		10.
g/l solvent	kg/m3 solvent		1.
kg/m3 solvent	kg/m3 solvent	kg solute per m3 solvent	1.
kg/m3(0 C, 1 atm)	kg/m3(0 C, 1 atm)	kg per norm cubic m	1.
g/cm3.K	kg/m3.K		1.E3
kg/m3.K	kg/m3.K		1.
g/mol	kg/mol		0.001
kg/mol	kg/mol		1.
lb/lb(force).hr.ft2	kg/N.s.m2		3.0489259E-4
g/atm.hr	kg/Pa.s		2.7414535E-6
g/bar.hr	kg/Pa.s		2.7777778E-6
g/kPa.hr	kg/Pa.s		2.7777778E-4

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
g/kPa.min	kg/Pa.s		1.6666667E-8
g/kPa.s	kg/Pa.s		1.E-6
g/mmHg.hr	kg/Pa.s		2.0835042E-3
kg/atm.hr	kg/Pa.s		2.7414535E-9
kg/atm.s	kg/Pa.s		9.8692327E-6
kg/bar.hr	kg/Pa.s		2.77777778E-9
kg/bar.s	kg/Pa.s		1.E-5
kg/cmH2O.hr	kg/Pa.s		2.8326244E-6
kg/kPa.hr	kg/Pa.s		2.77777778E-7
kg/kPa.min	kg/Pa.s		1.66666667E-5
kg/kPa.s	kg/Pa.s		1.E-3
kg/mmH2O.hr	kg/Pa.s		2.8326244E-7
kg/mmHg.hr	kg/Pa.s		2-0835042E-6
kg/Pa.hr	kg/Pa.s		2.77777778E-4
lb/atm.hr	kg/Pa.s		1.2435024E-9
lb/atm.s	kg/Pa.s		4.4766086E-6
lb/inH2O.hr	kg/Pa.s		5.0585892E-7
lb/inHg(32F).hr	kg/Pa.s		3.7207138E-8
lb/psi.hr	kg/Pa.s		1.827444E-8
lb/psi.min	kg/Pa.s		1.0964664E-6
lb/psi.s	kg/Pa.s		6.5787985E-5
g/s	kg/s		1.E-3
kg/day	kg/s		1.157407417E-5
kg/hr	kg/s		0.27777778E-3
kg/min	kg/s		0.0166666667
kg/s	kg/s		1.
lb/day	kg/s		5.249911667E-6
lb/hr	kg/s		0.12599788E-3
lb/s	kg/s		453.59237E-3
Mlb/hr	kg/s	1000 pounds (avdp) per hour	0.12599788
ton/day	kg/s		0.0115741
ton/hr	kg/s		0.27777778
ton/year	kg/s		3.1709792E-5
kg/kW.hr	kg/W.hr		2.777778E-7
lb/hp.hr	kg/W.s		1.689659E-7
kg-1.m-1.s-1	kg-1.m-1.s-1		1.
kJ/cycle	kJ/cycle		1000.

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
l/Val	l/Val		1.
Ang	m	Angstrøm	1.E-10
cm	m		0.01
dm	m		0.1
ft	m		0.3048
in	m		0.0254
km	m		1000.
m	m		1.
micron	m		1.E-6
mile	m		1609.344
mm	m		1.E-3
nm	m		1.E-9
um	m	mikro-meter	1.E-6
yd	m		0.9144
m/K	m/K		1.
ft/lb	m/kg		6.719690E-1
m/kg	m/kg		1.
bbl/ft2.hr	m/s	barrel per square foot and hour	4.75335474E-4
cm/hr	m/s		2.7777778E-6
cm/s	m/s		0.01
cm2/cm.s	m/s		0.01
ft/hr	m/s		8.46666667E-5
ft/min	m/s		5.08E-3
ft/s	m/s		0.3048
ft2/ft.hr	m/s		8.4666667E-5
ft2/ft.s	m/s		0.3048
ft3/ft2.s	m/s		0.3048
ft3/ft2.hr	m/s		8.46666667E-5
gal/ft2.min	m/s		6.79097E-4
km/hr	m/s		0.27777778
km/s	m/s		1000.
l/m2.hr	m/s		2.7778E-7
l/m2.s	m/s		1.E-3
m/hr	m/s		2.77777778E-4
m/min	m/s		0.0166666667
m/s	m/s		1.
m/s	m/s		1.

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
m2/m.hr	m/s		2.7777778E-4
m2/m.s	m/s		1.
m3/m2.s	m/s		1.
m3/m2.min	m/s		1.6666667E-2
mile/hr	m/s		0.44704
l/hr.rpm	m/s.rpm		2.7777778E-7
m3/min.rpm	m/s.rpm		1.6666667E-2
cm-1	m-1		100.
cm2/cm3	m-1		100.
ft-1	m-1		3.280840
ft2/ft3	m-1		3.2808
in-1	m-1		39.370079
in2/in3	m-1		0.39370079
m-1	m-1		1.
m2/m3	m-1		1.
mm-1	m-1		1000.0
mm2/mm3	m-1		1000.
m-1.s-2	m-1.s-2		1.
cm12/mol4	m12/mol4		1.E-24
l4/mol4	m12/mol4		1.E-12
m12/mol4	m12/mol4		1.
cm15/mol5	m15/mol5		1.E-30
m15/mol5	m15/mol5		1.
ft-2	m-2		10.76364864
cm2	m2		1.E-4
dm2	m2		0.01
ft2	m2		0.09290304
in2	m2		0.00064516
m2	m2		1.
mm2	m2		1.E-6
nm2	m2		1.E-18
m2.K/kW	m2.K/W		1.0E-3
m2.K/W	m2.K/W		1.
m2/g	m2/kg		0.001
m2/kg	m2/kg		1.
ft2/lb(force)	m2/N		2.0885434E-2
m2/Ohm	m2/Ohm		1.

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
cm2/Ohm.mol	m2/Ohm.mol		1.E-4
m2/Ohm.mol	m2/Ohm.mol		1.
Sie.cm2/mol	m2/Ohm.mol		0.01
cm2/Ohm.val	m2/Ohm.val		1.E-4
m2/Ohm.val	m2/Ohm.val		1.
bbl/ft.hr	m2/s	barrel/(foot and hour)	9.1134442E-4
cm2/s	m2/s		1.E-4
cSt	m2/s	centistokes	1.E-6
ft2/hr	m2/s		2.58064E-5
ft2/min	m2/s		1.548385E-3
ft3/ft.hr	m2/s		2.58064E-5
l/hr.m	m2/s		2.7777778E-7
m2/day	m2/s		1.1574074E-5
m2/hr	m2/s		2.7777778E-4
m2/s	m2/s		1.
m2/year	m2/s		3.1688087E-8
m3/m.min	m2/s		1.6666667E-2
m3/m.s	m2/s		1.
mm2/s	m2/s		1.E-6
mSt	m2/s		1.E-7
St	m2/s		1.E-4
cm2/s.mol fraction	m2/s.mol fraction		1.
m2/s.mol fraction	m2/s.mol fraction		1.
m2/s2	m2/s2		1.
cm2/V.s	m2/V.s		1.E-4
m2/V.s	m2/V.s		1.
bbl	m3	barrel	0.15898729
cm3	m3		1.0E-6
dm3	m3		1.E-3
ft3	m3		2.8316847E-2
gal	m3		3.7854118E-3
in3	m3		1.6387064E-5
l	m3	liters	1.E-3
m3	m3		1.
ml	m3		1.E-6
cm3.K/mol	m3.K/mol		1.E-6
ft3.Rnk/lbmol	m3.K/mol		3.468220043E-5

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
m3.K/kmol	m3.K/mol		1.E-3
bbl/cyle	m3/cycle	barrel/cycle	0. 15898729
ft3/cycle	m3/cycle		2.8316847E-2
gal/cycle	m3/cycle		3.7854118E-3
kbbl/cycle	m3/cycle	1000 barrels per cycle	0.15898729
l/cycle	m3/cycle		0.001
m3/cycle	m3/cycle		1.
MMft3/cycle	m3/cycle	million cubic feet per cycle	2.8316847E4
kbbl/day	m3/day		1.840131E-6
cm3/kg	M3/g		1.E-3
m3/K	m3/K		1.
cm3/100g	m3/kg		1.E-5
cm3/g	m3/kg		1.E-3
dm3/g	m3/kg		1.
dm3/kg	m3/kg		1.E-3
ft3/lb	m3/kg		0.062427962
l/g	m3/kg		1.
l/kg	m3/kg		1.E-3
m3/kg	m3/kg		1.
ml/g	m3/kg		1.E-3
cm3/g solvent	m3/kg solvent		1.E-3
m3/kg solvent	m3/kg solvent		1.
m3/kg.s	m3/kg.s		1.
m3/kg.s2	m3/kg.s2		1.
cm3/cm3 solvent	m3/m3 solvent		1.
cm3/l solvent	m3/m3 solvent		1.E-3
l/l solvent	m3/m3 solvent		1.
m3/m3 solvent	m3/m3 solvent		1.
ml/ml solvent	m3/m3 solvent		1.
cm3/cm3(sat.solut.)	m3/m3(sat.solution)		1.
cm3/l.atm	m3/m3.Pa		0.0098692327E-6
m3/m3.Pa	m3/m3.Pa		1.
dm3/mol	m3/mol		1.E-3
ft3/lbmol	m3/mol		6.242796029E-5
l/mol	m3/mol		1.E-3
m3/kmol	m3/mol		0.001
m3/mol	m3/mol		1.

unit	**SI-unit**	**remarks**	**factor unit -> SI-unit (only for information)**
ml/mol	m3/mol		1.E-6
cm3/mol2	m3/mol2		1.E-6
cm3/bar.g	m3/Pa.kg		1.E-8
m3/Pa.kg	m3/Pa.kg		1.
bbl/psi.hr	m3/Pa.s	barrel/(pounds per square inch and hour)	6.4053188E-9
ft3/psia.min	m3/Pa.s		6.8450163E-8
ft3/psia.s	m3/Pa.s		4.107010E-6
l/atm.hr	m3/Pa.s		2.7414535E-12
l/kPa.hr	m3/Pa.s		2.7777778E-10
l/mmHg.hr	m3/Pa.s		2.0835042E-9
m3/kPa.hr	m3/Pa.s		2.7777778E-7
m3/kPa.min	m3/Pa.s		1.6666667E-5
m3/kPa.s	m3/Pa.s		1.E-3
ml/atm.hr	m3/Pa.s		2.7414535E-15
ml/kPa.hr	m3/Pa.s		2.7777778E-13
bbl/day	m3/s	barrel/day	1.8401307E-6
bbl/hr	m3/s	barrel/hour	0.04415E-3
ft3/day	m3/s		3.2774128E-7
ft3/hr	m3/s		7.8657907E-6
ft3/min	m3/s		4.7194744E-4
ft3/s	m3/s		0.028316846
gal/hr	m3/s		1.0515033E-6
gal/min	m3/s		6.3090196E-5
l/day	m3/s		1.157407E-8
l/hr	m3/s		2.7777778E-7
l/min	m3/s		1.6666667E-5
l/s	m3/s		0.001
l/year	m3/s		3.1688087E-11
m3/day	m3/s		1.1574074E-5
m3/hr	m3/s		2.7777778E-4
m3/min	m3/s		1.6666667E-2
m3/s	m3/s		1.
m3/year	m3/s		3.1688087E-8
Mft3/day	m3/s	1000 cubic feet per day	0.327741279E-3
MMft3/day	m3/s	million cubic feet per day	0.327741279
MMft3/hr	m3/s	million cubic feet per hour	7.8657907

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
cm3/hr.K	m3/s.K		2.7777778E-10
l/min.m	m3/s.m		1.6666667E-5
l/s.m	m3/s.m		0.001
m3/hr.m	m3/s.m		2.7777778E-4
l/min.m2	m3/s.m2		1.6666667E-5
l/s.m2	m3/s.m2		0.001
m3/hr.m2	m3/s.m2		2.7777778E-4
ft3/hr.psi	m3/s.Pa		1.140836E-9
ft3/min.rpm	m3/s.rpm		4.7194744E-4
ft3/s.rpm	m3/s.rpm		0.028316846
gal/hr.rpm	m3/s.rpm		1.0515033E-6
gal/min.rpm	m3/s.rpm		6.3090196E-5
l/day.rpm	m3/s.rpm		1.157407E-8
l/min.rpm	m3/s.rpm		1.6666667E-5
l/s.rpm	m3/s.rpm		0.001
m3/day.rpm	m3/s.rpm		1.1574074E-5
m3/hr.rpm	m3/s.rpm		0.2777778E-3
m3/s.rpm	m3/s.rpm		1.
m3/year.rpm	m3/s.rpm		3.1688087E-8
ft3/s2	m3/s2		0.028316846
m3/s2	m3/s2		1.
cm2.l/Ohm.mol2	m5/Ohm.mol2	square cm*liters per Ohm and mole squared	1.E-7
m5/Ohm.mol2	m5/Ohm.mol2		1.
cm2.l/s.mol	m5/s.mol		1.E-7
m5/s.mol	m5/s.mol		1.
cm6/mol2	m6/mol2		1.E-12
dm6/mol2	m6/mol2		1.E-6
ft6/lbmol2	m6/mol2		3.8972502E-9
l2/mol2	m6/mol2		1.E-6
m6/mol2	m6/mol2		1.
cm9/mol3	m9/mol3		1.E-18
dm9/mol3	m9/mol3		1.E-9
ft9/lbmol3	m9/mol3		2.4329738E-13
l3/mol3	m9/mol3		1.E-9
m9/mol3	m9/mol3		1.
mass fraction	mass fraction		1.

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
kmol	mol		1000.
lbmol	mol	pound-mole	453.592368
mol	mol		1.
mol %	mol fraction		1.E-2
mol fraction	mol fraction		1.
mol/mol	mol fraction		1.
kmol/cycle	mol/cycle		1000.
lbmol/cycle	mol/cycle		453.59237
mol/cycle	mol/cycle		1.E-3
lbmol/day	mol/day		5.249911667E-3
lbmol/day.ft	mol/day.m		1.722412E-2
mol/100g	mol/kg		10.
mol/kg	mol/kg		1.
molon	mol/kg		1.
umol/100 g	mol/kg	micro-mol per 100 grams	1.E-5
mol/m.min	mol/m.s		1.6666667E-2
mol/m.s	mol/m.s		1.
dmol/m3	mol/m3		0.1
kmol/m3	mol/m3		1.E3
lbmol/ft3	mol/m3		1.6018463E4
lbmol/gal	mol/m3		1.1982643E5
mmol/cm3	mol/m3		0.001
mmol/l	mol/m3		1.
mol/cm3	mol/m3		1.E6
mol/dm3	mol/m3		1.E3
mol/l	mol/m3		1.E3
mol/m3	mol/m3		1.
mol/l solvent	mol/m3 solvent		1000.
mol/m3 solvent	mol/m3 solvent		1.
mol/m3.K	mol/m3.K		1.
kmol/m3.atm	mol/m3.Pa		9.86923E-3
mol/m3.atm	mol/m3.Pa		9.86923E-6
mol/m3.Pa	mol/m3.Pa		1.
lbmol/ft3.hr	mol/m3.s		4.4495732
lbmol/ft3.min	mol/m3.s		2.6697439E2
lbmol/ft3.s	mol/m3.s		1.6018563E4

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
mol/cm3.hr	mol/m3.s		3.6E9
mol/cm3.s	mol/m3.s		1.E6
mol/l.hr	mol/m3.s		0.277777778
mol/l.s	mol/m3.s		1.E3
mol/mol solvent	mol/mol solvent		1.
lbmol/atm.s	mol/Pa.s		0.0044766086
lbmol/psi.min	mol/Pa.s		1.0964664E-3
lbmol/psi.s	mol/Pa.s		6.5787985E-2
mol/atm.hr	mol/Pa.s		2.7414535E-9
mol/bar.hr	mol/Pa.s		2.7777778E-9
mol/kPa.hr	mol/Pa.s		2.7777778E-7
mol/kPa.min	mol/Pa.s		1.66666667E-5
mol/kPa.s	mol/Pa.s		1.E-3
kmol/day	mol/s		1.157407417E-5
kmol/hr	mol/s		0.27777778
kmol/s	mol/s		0.001
lbmol/hr	mol/s		0.12599788
lbmol/s	mol/s		0.45359237E3
mol/min	mol/s		1.6666667E-2
mol/s	mol/s		1.E-3
kmol/day.m	mol/s.m		1.157407417E-5
kmol/hr.m	mol/s.m		0.27777778E-6
kmol/s.m	mol/s.m		0.001
lbmol/hr.ft	mol/s.m		0.41337887
lbmol/s.ft	mol/s.m		1.4881639E3
kmol/hr.m3	mol/s.m3		0.27777778E-6
kmol/s.m3	mol/s.m3		0.001
kmol/hr.atm	mol/s.Pa		2.7414535E-6
kmol/hr.bar	mol/s.Pa		2.77777778E-6
kmol/hr.kPa	mol/s.Pa		2.77777778E-4
kmol/hr.mmH2O	mol/s.Pa		0.0283262437
kmol/hr.Pa	mol/s.Pa		0.27777778
kmol/min.kPa	mol/s.Pa		0.0166666667
kmol/s.atm	mol/s.Pa		9.8692327E-9
kmol/s.bar	mol/s.Pa		1.E-8
kmol/s.kPa	mol/s.Pa		1.E-6
mol-1	mol-1		1.

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
mol2/l2	mol2/m6		1.E6
mol2/m6	mol2/m6		1.
mol/100g solvent	molal		10.
mol/g solvent	molal		1000.
mol/kg solvent	molal		1.
molal	molal		1.
molal-1	molal-1		1.
molar	molar		1.
dyn	N		1.0E-5
lb(force)	N		4.4482216
N	N		1.
dyn^(1/4).cm^(11/4)/ml	N^(1/4).m^(11/4)/mol		1.7782794E-7
N^(1/4).m^(11/4)/mol	N^(1/4).m^(11/4)/mol		1.
lb(force).ft	N.m		1.3558179
lb(force).ft/lb	N.m/kg		2.989067
lb(force).ft/s	N.m/s		1.3558179
mN.s/m2	N.s/m2		0.001
dyn/cm	N/m		1.E-3
lb(force)/ft	N/m		14.593903
mN/m	N/m		1.E-3
N/m	N/m		1.
N/m2.m	N/m		1.
lb(force)/ft2	N/m2		47.880259
lb(force)/ft3	N/m3		157.08747
mmH2O/m	N/m3		9.806348
N/m3	N/m3		1.
Ohm	Ohm		1.
Ohm.cm	Ohm.m		0.01
Ohm.m	Ohm.m		1.
Ohm-1	Ohm-1		1.
Ohm-1.cm-1	Ohm-1.m-1		100.
Ohm-1.m-1	Ohm-1.m-1		1.
Sie/cm	Ohm-1.m-1		100.
Sie/m	Ohm-1.m-1		1.
Ohm-1.m-3	Ohm-1.m-3		1.
Sie/ml	Ohm-1.m-3		1.E6
ata	Pa	absolute atmospheres	98066.5

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
atm	Pa		101325.
bar	Pa		1.E5
cmH2O	Pa	cm of water column (pressure)	98.063754
Gpa	Pa		1.E9
Hpa	Pa		100.
inH2O	Pa	inches of water columns (pressure)	249.0889
inHg(32F)	Pa		3386.39
inHg(60F)	Pa		3332.69
kbar	Pa		1.E8
kg(force)/cm2	Pa		98066.5
kN/m2	Pa		1000.
kp/cm2	Pa		98066.5
kp/m2	Pa		9.80665
kPa	Pa		1.E3
lb(force)/in2	Pa		6.89476E3
lb/in2	Pa		6.89476E3
mbar	Pa		100.
mmH2O	Pa	mm of water column	9.806348
mmHg	Pa	mm of Hg column (Torr)	133.3223684
MN/m2	Pa		1.E6
mPa	Pa		1.E-3
MPa	Pa		1.E6
mTorr	Pa		0.1333224
N/m2	Pa		1.
Pa	Pa		1.
psi	Pa	pounds per square inch	6894.7573
Torr	Pa		133.3224
ata.K	Pa.K		98066.5
atm.K	Pa.K		101325.
kPa.K	Pa.K		1000.
Pa.K	Pa.K		1.
kg(force).m/kg	Pa.m/kg		9.80665
bar.m/s	Pa.m/s		1.E5
Pa.m/s	Pa.m/s		1.
atm.ft3/lbmol	Pa.m3/mol		6.3255133
Pa.cm3/mol	Pa.m3/mol		1.E6

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
Pa.dm3/mol	Pa.m3/mol		1.E3
Pa.m3/mol	Pa.m3/mol		1.
psi.ft6	Pa.m6		5.5285184
atm.dm6.K/mol2	Pa.m6.K/mol2		0.101325
Pa.m6.K/mol2	Pa.m6.K/mol2		1.
atm.l2.K2/mol2	Pa.m6.K2/mol2		0.101325
bar.m6.K2/mol2	Pa.m6.K2/mol2		1.E5
kPa.m6.K2/mol2	Pa.m6.K2/mol2		1000.
Pa.m6.K2/mol2	Pa.m6.K2/mol2		1.
bar.cm6/g.K	Pa.m6/kg.K		1.E2
Pa.m6/kg.K	Pa.m6/kg.K		1.
atm.l2/mol2	Pa.m6/mol2		0.101325
kPa.m6/mol2	Pa.m6/mol2		1000.
Pa.m6/mol2	Pa.m6/mol2		1.
psi.ft6/lbmol2	Pa.m6/mol2		2.68706E-5
psi.ft9	Pa.m9		0.15655021
kPa.m9.K2/mol3	Pa.m9.K2/mol3		1000.
kPa.m9/mol3	Pa.m9/mol3		1000.
Pa.m9/mol3	Pa.m9/mol3		1.
psi.ft9/lbmol3	Pa.m9/mol3		1.6774764E-9
kbar.mol.K/cm3	Pa.mol.K/m3		1.E14
Pa.mol.k/m3	Pa.mol.K/m3		1.
cP	Pa.s	centipoise	0.001
g/cm.s	Pa.s		0.1
kg(force).s/m2	Pa.s		9.80665
kg/m.s	Pa.s		1.
kp.s/m2	Pa.s		9.807
lb(force).s/ft2	Pa.s		47.880262
lb/ft.hr	Pa.s		4.1337887E-4
lb/ft.s	Pa.s		1.48816
mP	Pa.s		1.E-4
mPa.s	Pa.s		1.E-3
N.s/m2	Pa.s		1.
P	Pa.s		0.1
Pa.s	Pa.s		1.
psi.hr	Pa.s		2.4821136E7
ug/cm.s	Pa.s	mico-gram per cm and s	1.E-7

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
uP	Pa.s	micropoise	1.E-7
uPa.s	Pa.s	micro-Pascal-seconds	1.E-6
mPa.s.cm3/g	Pa.s.m3/kg		1.E-6
Pa.s.m3/kg	Pa.s.m3/kg		1.
uPa.s.cm3/g	Pa.s.m3/kg	micro-Pascal.s.cm3/g	1.E3
Pa.s/K	Pa.s/K		1.
Pa.s/K2	Pa.s/K2		1.
Pa.s/K3	Pa.s/K3		1.
cP/mol %	Pa.s/mol fraction		0.1
cP/mol fraction	Pa.s/mol fraction		0.001
Pa.s/mol fraction	Pa.s/mol fraction		1.
lb/ft.hr2	Pa.s2		1.1482746E-7
lb/ft.s2	Pa.s2		1.4881662
ata/K	Pa/K		98066.5
atm/K	Pa/K		101325.
bar/K	Pa/K		1.E5
kbar/K	Pa/K		1.E8
kPa/K	Pa/K		1.E3
mbar/K	Pa/K		1.E2
mPa/K	Pa/K		1.E-3
MPa/K	Pa/K		1.E6
Pa/K	Pa/K		1.
psi/F	Pa/K		12410.568
Torr/K	Pa/K		133.3224
ata/m	Pa/m		98066.5
atm/m	Pa/m		101325.
bar/km	Pa/m		100.
bar/m	Pa/m		1.E5
cmH2O/m	Pa/m		98.063754
inH2O/ft	Pa/m		817.2208
inHg(32F)/ft	Pa/m		11110.203
kPa/km	Pa/m		1.E6
kPa/mm	Pa/m		1.
mbar/m	Pa/m		1.E2
mH2O/m	Pa/m		98.063754
mmHg/ft	Pa/m		437.4094
Pa/m	Pa/m		1.

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
psi/100ft	Pa/m		2.2620595E2
psi/ft	Pa/m		2.2620595E4
Torr/ft	Pa/m		437.40945
kbar/cm3.mol	Pa/m3.mol		1.E14
Pa/m3.mol	Pa/m3.mol		1.
ata/mol fraction	Pa/mol fraction		98066.5
atm/mol fraction	Pa/mol fraction		101325.
bar/mol fraction	Pa/mol fraction		1.E5
kPa/mol fraction	Pa/mol fraction		1.E3
MPa/mol fraction	Pa/mol fraction		1.E6
Pa/(mol/mol)	Pa/mol fraction		1.
Pa/mol fraction	Pa/mol fraction		1.
psi/mol fraction	Pa/mol fraction		6894.76
Torr/mol fraction	Pa/mol fraction		133.3224
ata/s	Pa/s		98066.5
atm/s	Pa/s		101325.
bar/s	Pa/s		1.E5
Pa/s	Pa/s		1.
atm/weight fraction	Pa/weight fraction		101325.
Pa/weight fraction	Pa/weight fraction		1.
MPa^0.5	Pa^0.5		1.E3
Pa^0.5	Pa^0.5		1.
ata-1	Pa-1		1.01972E-5
atm-1	Pa-1		9.86923E-6
bar-1	Pa-1		1.E-5
cm2/dyn	Pa-1		10.
cm2/kg(force)	Pa-1		1.0197162E-5
cm2/kp	Pa-1		1.01972E-5
Gpa-1	Pa-1		1.E-9
inH2O-1	Pa-1		4.0146309E-3
kbar-1	Pa-1		1.E-8
kPa-1	Pa-1		1.E-3
m2/kp	Pa-1		0.101972
m2/N	Pa-1		1.
mbar-1	Pa-1		1.E-2
mmH2O-1	Pa-1		0.1019748
mPa-1	Pa-1		1.E3

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
MPa-1	Pa-1		1.E-6
Pa-1	Pa-1		1.
psi-1	Pa-1		1.4503774E-4
Torr-1	Pa-1		7.500615E-3
TPa-1	Pa-1		1.E-12
atm-1.K-1	Pa-1.K-1		9.86923E-6
Pa-1.K-1	Pa-1.K-1		1.
P-1	Pa-1.s-1		10.
Pa-1.s-1	Pa-1.s-1		1.
ata-2	Pa-2		1.0398289E-10
atm-2	Pa-2		9.7401753E-11
bar-2	Pa-2		1.E-10
kPa-2	Pa-2		1.E-6
mbar-2	Pa-2		1.E-4
Pa-2	Pa-2		1.
Torr-2	Pa-2		5.6259226E-5
ata-3	Pa-3		1.0603225E-15
atm-3	Pa-3		9.6128057E-16
bar-3	Pa-3		1.E-15
kPa-3	Pa-3		1.E-9
mbar-3	Pa-3		1.E-6
Pa-3	Pa-3		1.
Torr-3	Pa-3		4.219788E-7
ata-4	Pa-4		1.081228E-20
atm-4	Pa-4		9.4871016E-21
bar-4	Pa-4		1.E-20
kPa-4	Pa-4		1.E-12
mbar-4	Pa-4		1.E-8
Pa-4	Pa-4		1.
Torr-4	Pa-4		3.1651005E-9
day	s		86400.
hr	s		3600.
min	s		60.
ms	s		0.001
ns	s		1.E-9
ps	s		1.E-12
s	s		1.

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
us	s	microseconds	0.000001
year	s		31536000.
hr.ft2.Rnk/BTU	s.m2.K/J		1.7611019E-1
hr.m2.K/kcal	s.m2.K/J		8.5984523E-1
s.cm2.K/cal	s.m2.K/J		2.3884590E-5
s.m2.K/J	s.m2.K/J		1.0
s.m2.K/kcal	s.m2.K/J		2.3884590E-4
s.m2.K/kJ	s.m2.K/J		1.0E-3
day-1	s-1		1.1574074E-5
GHz	s-1		1.E9
kHz	s-1		1000.
MHz	s-1		1000000.
rad/s	s-1		0.15915475
year-1	s-1		3.1688088E-8
kV	V		1.E3
mV	V		1.E-3
V	V		1.
V/cm	V/m		100.
V/m	V/m		1.
val/kg	val/kg		1.
val/l	val/m3		1.E3
val/m3	val/m3		1.
cm3/100cm3	volume fraction		0.01
cm3/cm3	volume fraction		1.
cm3/l	volume fraction		1.E-3
l/m3	volume fraction		1.E-3
m3/m3	volume fraction		1.
ml/m3	volume fraction		1.E-6
volume %	volume fraction		0.01
volume fraction	volume fraction		1.
BTU/s	W		1.0550559E3
erg/s	W		1.E-7
hp	W	hourse power	745.69987
kcal/s	W		4.1868E3
kp.m/s	W		9.80665
kW	W		1000.
PS	W		745.700

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
W	W		1.
hp.hr	W.s		2.6845195E6
kW/cycle	W/cycle		1000.
W/cycle	W/cycle		1.
W/degC	W/K		1
W/K	W/K		1.
W/kg	W/kg		1.
W/m	W/m		1.
BTU.in/ft2.hr.degF	W/m.K		0.14422789
BTU.in/ft2.hr.Rnk	W/m.K		0.14422789
BTU/ft.hr.degF	W/m.K		1.7307347
BTU/ft.s.degF	W/m.K		6230.6449
BTU/in.hr.degF	W/m.K		20.768816
BTU/in.s.degF	W/m.K		7.4767738E4
cal/cm.s.K	W/m.K		4.1868E2
cal/km.s.K	W/m.K		4.1868E-3
cal/m.hr.K	W/m.K		1.163E-3
cal/m.s.K	W/m.K		4.1868
erg/cm.s.K	W/m.K		1.E-5
J/cm.s.K	W/m.K		1.E2
J/m.s.K	W/m.K		1.
kcal.m/hr.m2.K	W/m.K		1.163
kcal/m.hr.K	W/m.K		1.163
kcal/m.s.K	W/m.K		4.1868E3
kW/m.K	W/m.K		1.0E3
mW/cm.degC	W/m.K		0.1
mW/cm.K	W/m.K		0.1
mW/m.K	W/m.K		0.001
uW/cm.K	W/m.K	micro-Watt per cm and K	0.0001
W/cm.K	W/m.K		1.E2
W/m.K	W/m.K		1.
W/m.K2	W/m.K2		1.
W/m.K3	W/m.K3		1.
W/m.K4	W/m.K4		1.
W/m2	W/m2		1
cal/cm2.s.K	W/m2.K		4.1868E4
kW/m2.K	W/m2.K		1.0E3

unit	SI-unit	remarks	factor unit -> SI-unit (only for information)
W/m2.degC	W/m2.K		1.
W/m2.K	W/m2.K		1.
W/m2.s.K	W/m2.s.K		1.
hp/ft3	W/m3		2.6334143E4
kW/l	W/m3		1.E6
kW/m3	W/m3		1.E3
W/m3	W/m3		1.
W/mol	W/mol		1.
hp/lbmol.hr	W/mol.s		4.566631EE-4
hp/lbmol.s	W/mol.s		1.64398709
kW/kmol.hr	W/mol.s		2.7777778E-4
kW/kmol.s	W/mol.s		1.
g/100g	weight fraction		1.E-2
g/g	weight fraction		1.
g/kg	weight fraction		0.001
kg/kg	weight fraction		1.
mg/kg	weight fraction		1.E-6
ppm	weight fraction		1.E-6
weight %	weight fraction		1.E-2
weight fraction	weight fraction		1.

bbl = petroleum barrel, USA (42 US liquid gallons)

cal(th) = thermochemical calory

gal = liquid gallons, USA hp =

mechanical horse power lb = pound

(avdp.) oz = ounces (avdp.) u... =

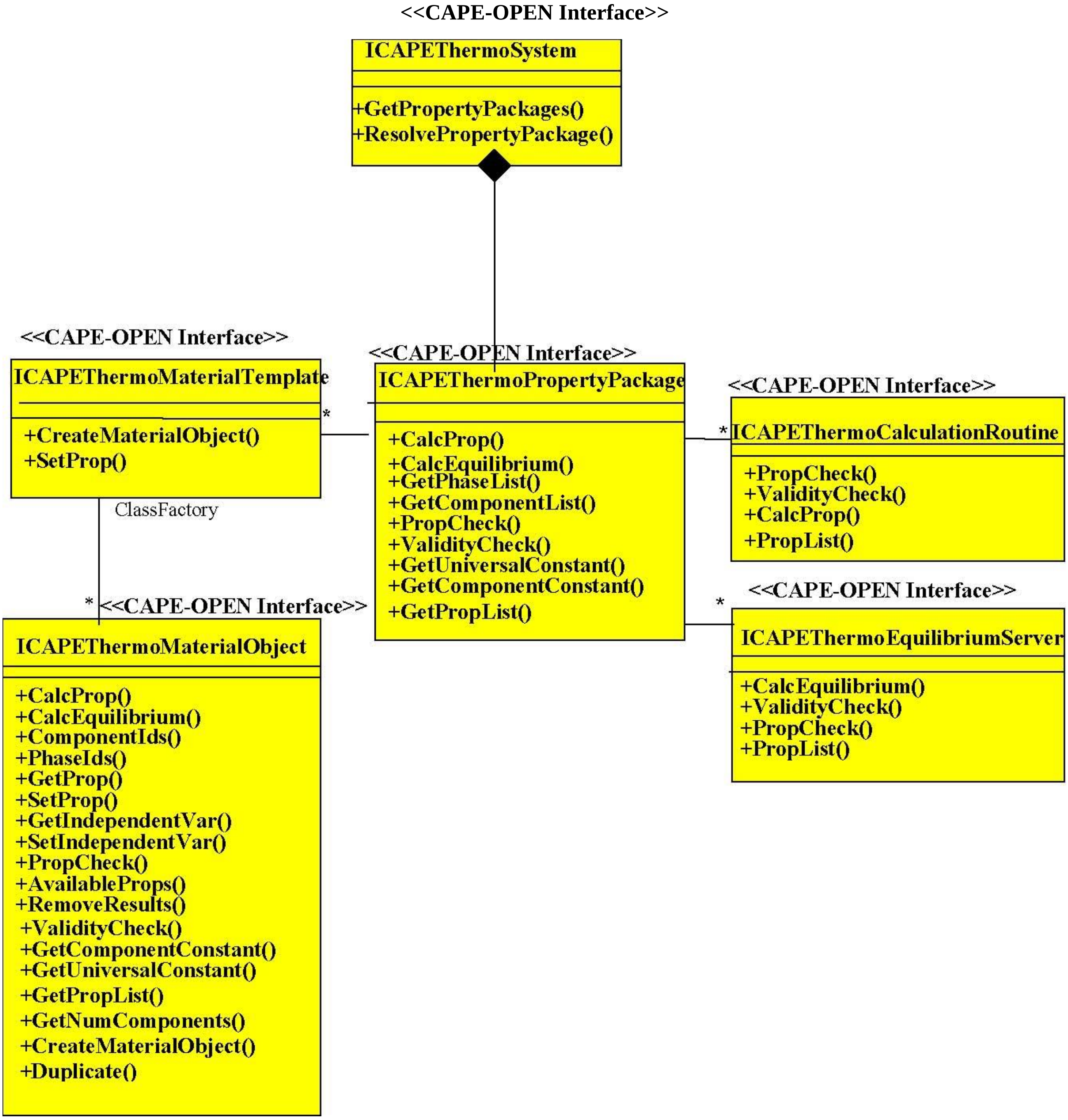
mikro... m... = milli... M... = Mega...

6.1.3. Pure Component Constant Properties (CAPE-OPEN)

6.1.4. Relevant Thermo Interfaces

4.7 CAPE-OPEN Thermo Interface Diagram

This diagram does not describe how or when these interfaces are executed in the context of a working Open System / CAPE-OPEN Environment. The mechanisms by which these underlying associations are executed are proprietary to the simulation environments. This diagram represents a general abstract view of the interfaces. The full overview of the interfaces is described in both the Component Diagram and the Interface Diagram.



A more detailed view can be found in the corresponding IDL and code samples. It is very important to note that the interface diagrams expose the necessary interfaces for using plug and play components. These diagrams do not imply the internal traversal path for how these interfaces are executed. Note: The results are stored in the Material Object and accessed through the GetProp method.

4.9.2 ICapeThermoMaterialObject

4.9.2.1 ComponentIds

Interface Name **ICapeThermoMaterialObject**
Method Name ComponentIds
Returns CapeError

Description
Returns the list of components Ids of a given Material Object.

Arguments

Name	Type	Description
[out, retval] *compIds	CapeVariant	Component IDs (String Array)

4.9.2.2 PhaseIds

Interface Name ICapeThermoMaterialObject Method Name PhaseIds **Returns** CapeError

Description
It returns the phases existing in the MO at that moment. The Overall phase and multiphase identifiers cannot be returned by this method. See notes on Existence of a phase for more information.

Arguments

Name	Type	Description
[out, retval] *phaseIds	CapeVariant	List of phases (String Array)

4.9.2.3 GetUniversalConstant

Interface Name ICapeThermoMaterialObject Method Name GetUniversalConstant **Returns** CapeError

Description
Retrieves universal constants from the Property Package.
Arguments

Name	Type	Description
[in] props	CapeVariant (String Array)	List of universal constants to be retrieved
[out, retval] *propvals	CapeArrayVariant	Values of universal constants

4.9.2.4 GetComponentConstant

Interface Name ICapeThermoMaterialObject Method Name GetComponentConstant **Returns** CapeError

Description
Retrieve component constants from the Property Package. See Notes for more information.

Arguments Description

Name	Type	Description

[in] props	CapeVariant (String Array)	List of component constants
[in] compIds	CapeVariant (String Array)	List of component IDs for which constants are to be retrieved. emptyVariant for all components in the Material Object.
[out,retval] *propvals	CapeVariant (Variant Array)	Component Constant values returned from the Property Package for all the components in the Material Object It is a variant containing a 1 dimensional array of variants. If we call P to the number of requested properties and C to the number requested components the array will contain C*P variants. The C first ones (from position 0 to C-1) will be the values for the first requested property (one variant for each component). After them (from position C to 2*C-1) there will be the values of constants for the second requested property, and so on.

4.9.2.5 CalcProp

Interface Name **ICapeThermoMaterialObject**
Method Name CalcProp
Returns CapeError

This method is responsible for doing all property calculations and delegating these calculations to the associated thermo system. This method is further defined in the descriptions of the CAPE-OPEN Calling Pattern and the User Guide Section. See Notes for a more detailed explanation of the arguments and CalcProp description in the notes for a general discussion of the method.

Arguments

Name	Type	Description
[in] props	CapeVariant (String Array)	The List of Properties to be calculated.
[in] phases	CapeVariant (String Array)	List of phases for which the properties are to be calculated.
[in] calcType	CapeString	Type of calculation: Mixture Property or Pure Component Property. For partial property, such as fugacity coefficients of components in a mixture, use “Mixture” CalcType. For pure component fugacity coefficients, use “Pure” CalcType.

4.9.2.6 GetProp

Interface Name ICapeThermoMaterialObject Method Name GetProp **Returns** CapeError

Description
This method is responsible for retrieving the results from calculations from the MaterialObject. See Notesfor a more detailed explanation of the arguments.

Arguments

Name	Type	Description
[in] property	CapeString	The Property for which results are requested from the MaterialObject.
[in] phase	CapeString	The qualified phase for the results.
[in] compIds	CapeVariant (String Array)	The qualified components for the results. emptyVariant to specify all components in the Material Object. For mixture property such as liquid enthalpy, this qualifier is not required. Use emptyVariant as place holder.
[in] calcType	CapeString	The qualified type of calculation for the results. (valid Calculation Types: Pure and Mixture)
[in] basis	CapeString	Qualifies the basis of the result (i.e., mass /mole). Default is mole. Use NULL for default or as place holder for property for which basis does not apply (see also Specific properties.
[out, retval] *results	CapeVariant (Double Array)	Results vector containing property values in SI units arranged by the defined qualifiers.

4.9.2.7 SetProp

Interface Name ICapeThermoMaterialObject Method Name SetProp **Returns** CapeError

Description
This method is responsible for setting the values for properties of the Material Object. See Notes for a more detailed explanation of the arguments.

Arguments

Name	Type	Description
[in] property	CapeString	The property for which the values need to be set.
[in] phase	CapeString	Phase, if applicable. Use NULL for place holder.
[in] compIds	CapeVariant (String Array)	Components for which values are to be set. emptyVariant to specify all components in the Material Object. For mixture property such as liquid enthalpy, this qualifier is not required. Use emptyVariant as place holder.

[in] calcType	CapeString	The calculation type. (valid Calculation Types: Pure and Mixture)
[in] basis	CapeString	Qualifies the basis (mole / mass). See also Specific properties.
[in] values	CapeVariant (Double Array)	Values to set for the property.

4.9.2.8 CalcEquilibrium

Interface Name ICapeThermoMaterialObject Method Name CalcEquilibrium **Returns** CapeError

Description
This method is responsible for delegating flash calculations to the associated Property Package or Equilibrium Server. It must set the amounts, compositions, temperature and pressure for all phases present at equilibrium, as well as the temperature and pressure for the overall mixture, if not set as part of the calculation specifications. See CalcProp and CalcEquilibrium for more information.

Arguments

Name	Type	Description
[in] flashType	CapeString	Flash calculation type.
[in] props	CapeVariant (String Array)	Properties to be calculated at equilibrium. emptyVariant for no properties. If a list, then the property values should be set for each phase present at equilibrium.

4.9.2.9 SetIndependentVar

Interface Name ICapeThermoMaterialObject Method Name SetIndependentVar **Returns** CapeError

Description
Sets the independent variable for a given Material Object.

Arguments

Name	Type	Description
[in] indVars	CapeVariant (String Array)	Independent variables to be set (see **names for state variables** for list of valid variables)
[in] values	CapeVariant (Double Array)	Values of independent variables.

4.9.2.10 GetIndependentVar

Interface Name ICapeThermoMaterialObject Method Name GetIndependentVar **Returns** CapeError

Description
Returns the independent variables of a Material Object.

Arguments

Name	Type	Description
[in] indVars	CapeVariant (String Array)	Independent variables to be set (see **names for state variables** for list of valid variables)
[out, retval] *values	CapeArrayDouble	Values of independent variables.

4.9.2.11 PropCheck

Interface Name ICapeThermoMaterialObject Method Name PropCheck **Returns** CapeError

Description
Checks to see if given properties can be calculated.

Arguments

Name	Type	Description
[in] props	CapeVariant (String Array)	Properties to check.
[out, retval] *valid	CapeArrayBoolean	Returns Boolean List associated to list of properties to be checked.

4.9.2.12 AvailableProps

Interface Name **ICapeThermoMaterialObject**
Method Name AvailableProps
Returns CapeError

Description
Gets a list properties that have been calculated.

Arguments

Name	Type	Description
[out,retval] *props	CapeArrayString	Properties for which results are available.

4.9.2.13 RemoveResults

Interface Name **ICapeThermoMaterialObject**
Method Name RemoveResults
Returns CapeError

Description
Remove all or specified property results in the Material Object.

Arguments

Name	Type	Description
[in] props	CapeVariant (String Array)	Properties to be removed. emptyVariant to remove all properties.

Name

Type

Description

4.9.2.14 CreateMaterialObject

Interface Name **ICapeThermoMaterialObject**
Method Name CreateMaterialObject
Returns CapeError

Description
Create a Material Object from the parent Material Template of the current Material Object. This is the same as using the CreateMaterialObject method on the parent Material Template.

Arguments

Name	Type	Description
[out, retval] MaterialObject	ICapeInterface*	The created/initialized Material Object.

4.9.2.15 Duplicate

Interface Name **ICapeThermoMaterialObject**
Method Name Duplicate
Returns CapeError

Description
Create a duplicate of the current Material Object.

Arguments

Name	Type	Description
[out, retval] clone	ICapeInterface *	The created/initialized Material Object.

4.9.2.16 ValidityCheck

Interface Name **ICapeThermoMaterialObject**
Method Name ValidityCheck
Returns CapeError

Description
Checks the validity of the calculation.

Arguments

Name	Type	Description
[in] props	CapeVariant (String Array)	The properties for which reliability is checked.
[out, retval] *rellist	CapeArrayThermoReliability	Returns the reliability scale of the calculation.

4.9.2.17 GetPropList

Interface Name **ICapeThermoMaterialObject**
Method Name GetPropList
Returns CapeError

Description
Returns list of properties supported by the property package and corresponding CO Calculation Routines. The properties TEMPERATURE, PRESSURE, FRACTION, FLOW, PHASEFRACTION, TOTALFLOW cannot be returned by GetPropList, since all components must support them. Although the property identifier of derivative properties is formed from the identifier of another property, the GetPropList method will return the identifiers of all supported derivative and non-derivative properties. For instance, a Property Package could return the following list:

enthalpy, enthalpy.Dtemperature, entropy, entropy.Dpressure.

Arguments

Name	Type	Description
[out, retval] *props	CapeArrayString	String list of all supported properties of the property package.

4.9.2.18 GetNumComponents

Interface Name **ICapeThermoMaterialObject**
Method Name GetNumComponents
Returns CapeError

Description
Returns number of components in Material Object.

Arguments

Name	Type	Description
[out, retval] *num	CapeLong	Number of components in the Material Object.

Name

Type

Description

4.9.6 ICapeThermoEquilibriumServer

4.9.6.1 CalcEquilibrium

Interface Name ICapeThermoEquilibriumServer Method Name CalcEquilibrium **Returns** CapeError

Description
Calculates the equilibrium properties requested. It must set the amounts, compositions, temperature and pressure for all phases present at equilibrium, as well as the temperature and pressure for the overall mixture, if not set as part of the calculation specifications. See CalcProp and CalcEquilibrium for more information.

Arguments

Name	Type	Description
[in] *materialObject	CapeInterface	MaterialObject of the calculation
[in] flashType	CapeString	Flash calculation type.
[in] props	CapeVariant (String Array)	Properties to be calculated at equilibrium. emptyVariant for no properties. If a list, then the property values should be set for each phase present at equilibrium.

4.9.6.2 PropCheck

Interface Name ICapeThermoEquilibriumServer Method Name PropCheck **Returns** CapeError

Description
Checks to see if a given type of flash calculations can be performed and whether the properties can be calculated after the flash calculation.

Arguments

Name	Type	Description
[in] *materialObject	CapeInterface	The Material Object for the calculations.
[in] flashType	CapeString	Type of flash calculation to check
[in] props	CapeVariant (String Array)	List of Properties to check. emptyVariant for none.

[out, retval] *valid	CapeArrayBoolean	The array of booleans for flash and property. First element is reserved for flashType.

4.9.6.3 ValidityCheck

Interface Name ICapeThermoEquilibriumServer Method Name ValidityCheck **Returns** CapeError

Description
Checks the reliability of the calculation.

Arguments

Name	Type	Description
[in] *materialObject	CapeInterface	The material object for the calculations.
[in] props	CapeVariant (String Array)	The list of properties to check. NULL for none.
[out, retval] *rellist	CapeArrayThermo Reliability	The properties for which reliability is checked. First element reserved for reliability of flash calculations.

4.9.6.4 PropList

Interface Name ICapeThermoEquilibriumServer Method Name PropList **Returns** CapeError

Description
Returns the flash types, properties, phases, and calculation types that are supported by a given Equilibrium Server Routine.

Arguments

Name	Type	Description
[in,out] *flashType	CapeVariant (String Array)	Type of flash calculations supported.
[in,out] *props	CapeVariant (String Array)	List of supported properties.
[in,out] *phases	CapeVariant (String Array)	List of supported phases.
[in,out] *calcType	CapeVariant (String Array)	List of supported calculation types. (Pure & Mixture)

7 Schedule

	Step 1	Step 2	Step 3	Step 4	Step 5	Step 6
User Interface	**Graphical Drag & Drop User Interface**					
	Stream and Unit Operations Input Forms					
Units Converter	Basic Unit Conversions, T, P, Q, etc					
Basic Thermo	Component Database	Low Level Functions	Equation of State	Flash Calculations	Distillation Algorithm	
	Pure	PSat1		Bubble & Dew Points		
	Binary	CPIG		Isothermal Flash		
	Vicosity Database	HVIG		Constant H Flash		
	Thermal Conductivity	SVIG		Constant S Flash		
	Surface Tension	CPIGMX				
		HVIGMX				
		SVIGMX				
Transport Properties		Visc_L, Visc_V				
		K_L, K_V				
		Sigma				
Reactors					Stoicheometric	
					Rate Based	
					Equilibrium	
Other Unit Operations					Splitters Mixers Component Separator	
Equipment Sizing					Lines Drums/Vessels Trays Heat Exchangers	
Report Generation					Stream Report Unit Operations Report Other Reports	
Kernal Development						Sequencing & Solving

8 Structure Description

I. Global Variables: The following variables are GLOBAL. They are defined in the mdlThermoControl module

Variable	Description	Type	Units
N_Comp	Number of Components	Integer	---
T	Temperature	Double	K
P	Pressure	Double	Bar
Vfrac	Vapor Fraction	Double	---
ZL	Liquid Compressibility Factor	Double	---
ZV	Vapor Compressibility Factor	Double	---
HLMX	Liquid Molar Enthalpy	Double	J/gmole
HVMX	Vapor Molar Enthalpy	Double	J/gmole
HMX	Mixture Molar Enthalpy	Double	J/gmole
SLMX	Liquid Molar Entropy	Double	J/gmole.K
SVMX	Vapor Molar Enthalpy	Double	J/gmole.K
SMX	Mixture Molar Enthalpy	Double	J/gmole.K
Z()	Overall Mole Fraction	Double	---
Y()	Vapor Mole Fraction	Double	---

X()	Liquid Mole Fraction	Double	---

II. Equations of State

Every equation of state should be in a separate module. All subroutines and functions that related to the EOS should be PRIVATE (not accessible to other routines) except for the ones listed below. If the name of the equation of state is ABC, then the equation should have the following subroutines

Routine	Description	Arguments
ABCZMX	Compressibility Factor	T, P, X(), N, IPhase
ABCPHX	Fugacity Coefficients	Phi, T, P, X(), N, Z
ABCHDM	Enthalpy Departure	T, P, Z, X(), N
ABCHDM	Entropy Departure	T, P, Z, X(), N

Where

T, P and N are the temperature, pressure and number of components respectively.

IPhase = 0 is to call for a liquid compressibility factor

IPhase = 1 is to call for a vapor compressibility factor

X() containing the compositions

Z: Compressibility factor

Phi: Array for fugacity coefficients.

III. Streams

The streams are currently work in progress and will be modified as needed. The initial thinking is to define the streams in a Pascal "Record" so that the stream properties can be access as follows:

Stream(1).Temp. Stream(1).Pres, etc.

The following are the parameters stored in a stream record

Parameter	Description	Type
Tag	Stream Tag	String
Name	Stream Name	String
Source	The operation the stream is coming from	
Destination	The operation the steam is going to	
Temp	Temperature	Double
Pres	Pressure	Double
Vfrac	Vapor Fraction	Double
ZL	Liquid Compressibility Factor	Double
ZV	Vapor Compressibility Factor	Double
HLMX	Liquid Molar Enthalpy	Double
HVMX	Vapor Molar Enthalpy	Double
HMX	Mixture Molar Enthalpy	Double
SLMX	Liquid Molar Entropy	Double
SVMX	Vapor Molar Enthalpy	Double
SMX	Mixture Molar Enthalpy	Double

Z()	Overall Mole Fraction	Double
Y()	Vapor Mole Fraction	Double
X()	Liquid Mole Fraction	Double
ZM()	Overall Mass Fraction	Double
MW	Molecular Weight	Double
Flow	Molar Flow Rate	Double
MassFlow	Mass Flow Rate	Double
Has_Flow	Indicates if the steam has a flow rate	Boolean
Has_Comp	Indicates if the steam has composition	Boolean
Has_Temp	Indicates if the steam has Temperature	Boolean
Has_Pres	Indicates if the steam has Pressure	Boolean
Has_Vfrac	Indicates if the steam has Vapor Fraction	Boolean
Has_HMX	Indicates if the steam has Enthalpy	Boolean
IsSolved	Indicates if the stream is Solved	Boolean

IV. Units

The units are set by the pure component database I will be donating. The reference pressure for heats of formation and Gibbs free energy of formation in this database is 1 bar, same for heat capacity. The following is a list of all key units that will be used in the heart of the simulator. User entries should be converted to these units (e.g. steam temperature, pressure, etc) BEFORE they are stored in the appropriate location.

Property	Units
Temperature	K
Pressure	Bar
Molar Flow Rate	gmole/hr
Mass Flow Rate	grams/hr
Enthalpy	J/gmole
Entropy	J/gmole.K
Heat Duty/Power	J/hr

I agree, this is the most disgusting set of units I have ever seen but let’s put up with it for now.

V.
VI.