

## **Expanding the CAPE-OPEN Thermodynamic interfaces: version 1.1**

Michel Pons  
CAPE-OPEN Laboratories Network  
32 rue Raulin, F-69007 Lyon, France

A revision of the CAPE-OPEN Thermodynamic and Physical Properties interface specification has been publicly released in October 2006 by the CAPE-OPEN Laboratories Network as version 1.1. This revision is part of the continuous improvement and extension process of the CAPE-OPEN standards. The specific aims of this revision are to extend the applicability of the CAPE-OPEN thermodynamic interfaces, to ease their implementation and to clarify a number of issues found in version 1.0 implementation and use.

### **1. Evolution process of the CAPE-OPEN standards**

As described by Belaud and Pons (2002), the CAPE-OPEN standards consist of a set of interface specifications that allows plug and play of process modelling software components into process modelling software environments. The scope encompasses the whole range of software components usually found in process simulation software. The set of interfaces results from the collaborative work of many organizations regrouped initially within European funded projects (CAPE-OPEN and Global CAPE-OPEN) then within a not-for-profit organization, the CAPE-OPEN Laboratories Network (CO-LaN). The interface specification for Thermodynamic and Physical Properties was designed very early on within the CAPE-OPEN standards. Consequently it has been implemented by many organisations and has had its limitations clearly identified.

As described in Pons (2003), version 1.0 has been implemented in AspenTech, SimSci-Esscor, Process Systems Enterprise, ProSim, Infochem Computer Services, TÜV-NEL, ProSim, Honeywell, HTRI, AmsterCHEM, RSI commercial software products. So a wide experience of implementation and usage has developed.

The missions of CO-LaN are:

- User priorities for CAPE-OPEN standards
- Exploitation and dissemination
- CAPE-OPEN interface specifications life cycle management
- Testing, interoperability facilitation
- Training/Migration facilitation

Under its third mission, CO-LaN has to organize the maintenance, evolution, and expansion of the specifications. CO-LaN has decided to launch a revision process for the Thermodynamics and Physical Properties interface specification.

### 1.1 Process applied to Thermodynamics and Physical Properties specification

Within CO-LaN a number of Special Interest Groups (SIGs) are set up with specific objectives. The SIGs gather experts in various fields from the CO-LaN membership, which is almost 60 organizations strong nowadays. The Thermodynamics SIG has the following objectives:

- Maintain and manage existing interface specifications (revisions to improve design, performance/speed and robustness based on user input)
- Assess and prioritize on expansions of interface specifications
- Manage the development of the agreed upon expansions
- Help organizations to develop implementations of the Thermodynamics standards

The Thermodynamics SIG delivered in October 2006 a revised version of the Thermodynamics and Physical Properties interface specification under the version number 1.1.

This is the outcome of a process involving a review of the 1.0 version, an analysis of the new requirements to be met and a design phase leading to a draft specification. This draft was used to develop a prototype plug software component as well as a prototype socket software environment. Following this prototyping and additional comments on the specification, a Request For Comments was issued in June 2006 and was successful in getting approval for public release from the CO-LaN Management Board on September 26, 2006.

### 1.2 Objectives of the revision

Clarity, efficiency, reliability, openness have been the main drivers of the revision.

The revision is meant to *improve clarity*. This has been done by removing ambiguities and better defining the purpose and functionality of the various software components involved.

The number and distribution of methods among interfaces have been heavily modified from version 1.0 to version 1.1. Version 1.0 specified the same methods in different interfaces so that there was a one-to-one correspondence between software components and interfaces. In 1.1 there are more interfaces but each one may be implemented by several software components.

This design is also meant to *simplify implementation* by reuse of entire sets of methods on different software components.

There is a general error handling mechanism defined in the CAPE-OPEN standards. A way to *improve the reliability*, hence to ensure a better interoperability is by defining and documenting precisely exceptions for each method. This has been done in the textual specification for 1.1.

Another way to *improve reliability* is to provide specific methods to check the capabilities of a software component before launching any actual calculation. A

common structure, leading to a common workflow, has been designed: enquiry methods as one step (to return a list of supported properties), check methods in another step (to verify possibility to conduct a specific calculation), calculate methods in a further step (to perform calculations).

The preeminence of phase equilibrium has been reduced. In version 1.0, the Material Object needed to comply with the requirement that all existing phases have the same temperature and pressure. In Thermo 1.1 this constraint has been dropped, recognizing that the equilibrium state is one case but not the only one. The basic concept is that phases are independent entities. The phases may or may not be in equilibrium.

In 1.0 the handling of phases apart from handling a vapour phase and a liquid phase, was limited. Version 1.1 provides support for any number of phases and any types of phases. This *adds new facilities* and *supports future extensions*. A central concept is now the phase label that is distinct from the state of aggregation. The phase label is an arbitrary string, e.g. "AqueousLiquid" that is used to identify a particular phase and to refer to its attributes. The attributes provide the necessary information for a client software or a human being to understand what sort of phase the label refers to.

In 1.0 phase equilibrium calculations were limited to a set of cases defined by a string, e.g. "PT" for a phase equilibrium calculation with pressure and temperature imposed. Version 1.1 does not impose anymore fixed identifiers for the type of flash but instead two specification variables are used to identify the constraints and the actual *values* of the constraints are retrieved using the Material Object.

The above objectives have led to a complete re-design of the 1.0 version while retaining basic concepts like the role of the Material Object as a container of data in between a Property Package and its final client. Consequently moving from a 1.0 implementation to a 1.1 implementation is not meant to be a major development task.

## 2. The interface specification

The Thermodynamic and Physical Properties interface specification (at <http://www.colan.org/index-37.html>) defines the following primary CAPE-OPEN components:

- Physical Property Calculator – a software component that can calculate a number of physical properties, possibly restricted to mixtures of particular compounds existing in particular phases. A Physical Property Calculator is not called directly by a Process Modelling Environment: it is called via a CAPE-OPEN Property Package. The purpose of a Physical Property Calculator is to extend or to override the list of calculations that a CAPE-OPEN Physical Property Package can perform. A CAPE-OPEN Property Calculator can only be used with a Property Package which supports the use of Physical Property Calculators.
- Equilibrium Calculator – a software component that can calculate the composition of mixtures, possibly restricted to mixtures of particular compounds existing in particular phases and subject to certain constraints. For example a specific

Equilibrium Calculator may not be able to deal with Materials with solid phases, or those containing polymeric compounds. An Equilibrium Calculator is not called directly by a Process Modelling Environment. Like a Physical Property Calculator, it is only called via a Property Package and the purpose of an Equilibrium Calculator is to extend or to override the list of equilibrium calculations that a CAPE-OPEN Property Package can perform.

- Property Package – a software component that combines the functions of a Property Calculator and an Equilibrium Calculator for a fixed set of compounds and phases. It combines the functions of a Physical Property Calculator and an Equilibrium Calculator.
- Property Package Manager – a software component that manages a set of Property Packages. It is responsible for instantiating Property Packages on request and may allow Property Packages to be edited and/or created.

The specification describes the following interfaces:

- ICapeThermoEquilibriumRoutine – methods implemented by software components which can perform an equilibrium calculation.
- ICapeThermoPropertyRoutine – methods implemented by software components that can calculate values for physical properties.
- ICapeThermoUniversalConstants – methods implemented by software components that can supply the values of universal constants.
- ICapeThermoPhases – methods implemented by software components that need to describe the phases that are present or could be present in a material.
- ICapeThermoCompounds – methods implemented by software components that need to describe the compounds that occur or can occur in a material.
- ICapeThermoMaterial – methods implemented by software components that need to provide access to the physical properties of a particular material.
- ICapeThermoMaterialContext – methods implemented by software components that need to be given a particular material as the context for a calculation.
- ICapeThermoPropertyPackageManager – methods implemented by a software component that can list and create instances of available Property Package components.

The defined software components support all the interfaces required to provide the functionality they implement. There is not a one-to-one correspondence between particular interfaces and particular software components. For example, a software component that implements the ICapeThermoEquilibriumRoutine may be an Equilibrium Calculator component or a Property Package component.

A Material Object implements CAPE-OPEN interfaces but a Material Object is not a CAPE-OPEN component. A Material Object is a software object that is responsible for holding the data describing the state of a Material. Each client software that uses CAPE-OPEN Thermodynamic and Physical Properties components must provide its own implementation of a Material Object because the configuration and data storage used in a Material Object will be different for each client.

### 3. Prototyping the proposed interface specification

#### 3.1 Need for prototyping

A CAPE-OPEN interface specification is an abstract specification: no code is delivered just a design that is meeting the requirements described in use cases. It is necessary, before releasing the specification for it to be implemented in any piece of code, that a proper verification of its operability be made. CO-LaN conducted the verification: Infochem Computer Services Ltd was contracted to deliver a plug implementation of version 1.1 while Universitat Politècnica de Catalunya, UPC was contracted to deliver a socket within the CAPE-OPEN Tester Suite.

#### 3.2 Results of prototyping

The conclusions of this prototyping work have been reported by Benqlilou et al. (2004). The prototyping proved that the overall design proposed is working well. However it showed that a number of modifications in the textual specification needed to be made in order to fully explain the purpose of a number of methods. Minor modifications to the signature of some methods proved to be necessary in order to simplify the design and to allow future extensions. While the CAPE-OPEN standards provide a general error handling mechanism, the prototyping showed that the use of this mechanism needed to be explained for each method in the textual specification. All this has been done plus taking into consideration additional comments sent by CO-LaN organizations.

### 4. Implementing the new interface specification

Having the interface specification released means that commercial and non-commercial software developers can use the published document to bring their code to compliance with version 1.1.

#### 4.1 Existing implementations

Infochem Computer Services Ltd has implemented the 1.1 interface specification in MultiFlash 3.6. MultiFlash is a thermodynamic server, hence a plug component.

AmsterCHEM has implemented version 1.1 in COFE 1.05 (process simulator) as well as in TEA 1.05 (thermodynamic server). So COFE can use a third-party 1.1 plug and TEA can be used as a plug in a 1.1 compliant Process Modelling Environment.

ProSim SA has developed a FORTRAN Thermodynamic Library that can be used to develop the socket calls to a 3<sup>rd</sup> party 1.1 plug.

Technische Universität Hamburg-Harburg has implemented a 1.1 Thermo socket in SolidSim so SolidSim process simulator is able to make use of a 3<sup>rd</sup> party thermodynamic server compliant with 1.1.

The following interoperability tests have been conducted successfully: MultiFlash has been used in SolidSim and COFE, MultiFlash and TEA have been used in a main program developed with the Thermodynamic Library developed by ProSim.

#### 4.2 Planned implementations

Aspentech is implementing version 1.1 in Aspen Hysys with a release scheduled for the end of 2007. Apart from a potential business interest of Aspentech in this implementation, this action is also a consequence of the consent order (Docket n°9310 – Dec 21, 2004) issued by the Federal Trade Commission on the Aspentech-Hyprotech merger. Aspentech was ordered to support the upgrading of Aspen Hysys to CAPE-OPEN Thermo 1.1.

TÜV-NEL Ltd has expressed interest in bringing their thermodynamic server PPDS to 1.1 compliance and work on that matter is scheduled for 2007.

At the University of Tulsa, a piece of software called TUWAX, that calculates the rate of wax deposit from oil and gas mixtures, is being brought to 1.1 compliance.

#### 5. Conclusion

A long and fruitful development process has been concluded successfully by immediate implementations of the new version of the Thermodynamics and Physical Properties interface specification. Additional implementations are under way with a strong recommendation from CO-LaN to all developers: use version 1.1!

#### 6. Acknowledgements

This work has been made possible through the commitment and collaboration of many people. Among them: Werner Drewitz (BASF AG), Suphat Watanasiri (Aspentech), Eric Hendriks (Shell Global Solutions bv), Jasper van Baten (AmsterCHEM), Matthias Pogodda (Technische Universität Hamburg-Harburg), Richard Szczepanski (Infochem Computer Services Ltd), Alain Vacher (Prosim SA), Alan Scott (TÜV NEL Ltd), Michael Halloran (then from Aspentech), Tom Williams (Process Systems Enterprise Ltd).

#### 7. References

- Belaud J.P. and M. Pons (2002), *Computer-Aided Chemical Engineering, vol. 10: Proceedings of the 12<sup>th</sup> European Symposium on Computer Aided Process Engineering*, pp 847-852, Eds J. Grievink and J. van Schijndel, Elsevier Science B.V., Amsterdam.
- Pons, M. (2003). Industrial Implementations of the CAPE-OPEN standard. *AIDIC Conference Series, vol. 6: Selected papers of the Sixth Italian Conference on Chemical and Process engineering*, pp. 253-262. Ed. S. Pierucci, Milan, Reed Business Information S.p.A., Milan.
- Benqlilou C., S. Bel. M. Gonzalez, M. Pons, R. Szczepanski, A. Espuna and L. Puigjaner, *Computer-Aided Chemical Engineering, vol. 18: European Symposium on Computer-Aided Process Engineering-14*, Eds. A. Barbosa-Povoa and H. Matos, Elsevier, Amsterdam. In [CD-ROM].