Methods and Tools
Special Interest Group Report
CAPE-OPEN 2019 Annual Meeting
Amsterdam

Michael Hlavinka

Bryan Research & Engineering, LLC

22 October 2019

SIG Membership

Bill Barrett
US EPA

Jasper van Baten
AmsterCHEM

Tony Garratt
ANSYS

Daniel Wagner
DWSIM

Mark Stijnman
Shell Global Solutions

Michael Hlavinka
Bryan Research & Engineering, LLC

Loic d'Anterroches
Céondo GmbH

David Jerome and Krishna Penukonda
AVEVA

M&T SIG Ongoing Activities

- Typically three or four conference calls per month. Calls are held on Wednesdays at 1700 CET, 1100 US Eastern Time, 1000 US Central Time.
  - General SIG business on the 1st and 4th Wednesday of the month
    - Development and Maintenance of Common Interface Specifications
    - COBIA development and distribution
  - Threading conference call on 3rd Wednesday of the month
- Joint Conference call with Interop SIG, typically 3rd Wednesday of the month at 2100 CET, 1500 US Eastern Time, 1400 US Central Time.
- Join? Please contact either SIG Leader or CTO
  - Bill Barrett – barrett.williamm@epa.gov
  - Michel Pons - technologyofficer@colan.org

M&T SIG Charter

- **Improve integration and expand utilization of Computer-Aided Process Engineering (CAPE) applications within the enterprise through identification and resolution of existing cross-cutting issues with the CAPE-OPEN platform, develop mechanisms for use of CAPE within other application domains, and incorporate advances in information technology into the CAPE-OPEN platform.**

- **Key responsibilities**
  - **Resolve issues with the common interface specifications.**
  - **Develop and maintain standards and protocols for CAPE-OPEN implementations.**
  - **Incorporate advances in information technology into the CAPE-OPEN protocols.**
  - **Identify novel uses of CAPE and provide standards for utilizing CAPE within these applications.**

No Change to vision and responsibilities.

M&T SIG 2018/2019 Summary of Activities

- **Update Interface Specifications**
  - Simulation Context List of Named Values
  - Flowsheet Monitoring
  - Revised Persistence Interfaces
  - Revised Parameters Interfaces
  - Economics/Currency Interfaces
  - Revised Reporting Interface

- **COBIA Development and Testing**
  - End-User Redistributable and SDK installers
  - Threading Model
  - Transitioning of persisted objects from COM to COBIA.

- **Interaction with Interop SIG**
  - Installer Support
  - Versioning

Simulation Context

- Named Value List have been added to CO-LaN web site
  - The NamedValues are defined within the CAPE-OPEN COSE Interface Specification, PME or other CAPE-OPEN Special Interest Group.
  - Contact Michel Pons regarding adding NamedValues to this table.
  - http://www.colan.org/cose-simulation-context-common-interface-specification/

NamedValue	Type	Definition	Scope
FreeFORTRANChannel	CapeLong	The COSE will return a different FORTRAN channel each time this NamedValue is called for this property. The COSE may not use any of the returned FORTRAN channels for any internally used FORTRAN module.	Standard
HTMLReportSupport	CapeBoolean	PME advertises to PMCs that HTML format is accepted	Standard
DefaultThermoVersion		Select thermodynamic version to match that of the Property Package in use so that the PME does not need to translate, for Unit Operations that support both thermodynamic versions	COCO
SimplifiedModelRequest	CapeBoolean	Inform Unit Operations that they should not run with full physics, but give an answer suitable for initializing a guess for the flowsheet.	COCO
AbortCalculation	CapeBoolean	Unit Operations should poll for this during lengthy calculations	UNIT
ResetOnValidate	CapeBoolean	Used to wipe internal guess values for a Unit Operation upon reset	UNIT

Flowsheet Monitoring Interface

- **Overview of Interface**
  - Provides the FMC the ability to access all elements in a flowsheet without interfering with the flowsheet
  - Event notifications provided to the FMC for changes to the state of the flowsheet.
- **Publication process has been completed.**
  - Comments from RFC taken into consideration in finalizing specification.
  - Thanks to reviewers!!!
  - Published to CO-LaN website on July 2019
- **IDL developed and ready for distribution through TLB/PIA installers**
  - Led to modifications in textual specification
- **Once released, the COCO/USEPA WAR Algorithm plug in will be updated.**

Revising Persistence Common Interfaces

- Motivation
  - Use of CAPE-OPEN Data Types.
  - Allow use of human-readable formats such as JSON or XML
  - Required for COBIA.
    - COMBIA should handle COM/COBIA persistence interoperability.
    - Allow use of platform-native serialization mechanisms through COBIA language bindings.
  - Single set of Interfaces Defined to reduce complexity vs COM Persistence Interfaces
- Design decision
  - Explicit separation of object serialization from storage to persistent media.
  - Generic Persistence Host and Persistence Client allows broader application.
  - Tree-based dictionary used to read/write object data as any CAPE-OPEN data type or a byte array.
- Status
  - Implemented in the Beta version of COBIA.
  - Textual specification under development.
  - Considering implications of adding these persistence interfaces to the COM Type Library. Looking for input from others.

Persistence Common Interfaces, Cont'd.

- ***ICapePersist***
  - ***Save*** **– Asks the PMC to save itself using the *ICapePersistWriter* interface.**
  - ***Restore*** **– Restores the PMC using the *ICapePersistReader* interface.**
  - **Dirty status – indicates whether Persistable Object has been modified.**

- ***ICapePersistWriter***
  - **Exposed by Persistence Host to the Persistable Object.**
  - **Provides methods to write standard CAPE-OPEN data types and byte array.**
  - **Method to add nodes to tree.**

- ***ICapePersistReader***
  - **Exposed by Persistence Host to the Persistable Object.**
  - **Provides methods to read standard CAPE-OPEN data types and byte array.**
  - **Methods to determine the names of the nodes present, and their data type.**

Revised Parameter Common Interfaces

Motivations

- Strongly typed Value/Elimination of the VARIANT.
- Eliminate the separation of static and dynamic elements of the Parameter.
- New Design:
  - Common Parameter members:
    - Mode
    - Type
    - Validate current value
    - Validation Status
  - Type-specific Parameter members:
    - Value
    - Upper/Lower Bounds
    - Default Value
    - Validate potential value
    - Units of Measure only provided for Real and Real Array parameters
- Backwards compatible with existing COM-based CAPE-OPEN implementations provided by COMBIA interoperability.
  - Array compatibility limited to homogeneous arrays

Parameter Interfaces, cont'd

- Array Parameters:
  - Homogeneous array types.
  - Passed as a one-dimensional array
  - Parameter value will be CapeArray(type)
    - CapeArrayInteger
    - CapeArrayReal
    - CapeArrayBoolean
    - CapeArrayString
  - Can obtain and set the value of the entire array at one time.
  - Parameter values can be obtained/set by element using the ICapeArray(Type)Parameter interface (New Functionality).
- ICapeArrayParameter interface has two properties:
  - NumDimensions
  - Size
- NOTE: All rows in a dimension have the same length, not a jagged array.

Currency Interface (Proposed)

- Queried CO-LaN membership about the interest in adding an economic dimension to parameters.
  - Positive comments received on adding the dimension.
  - Applicable to more than Parameters.
- Currency Issue
  - Need the ability to convert currencies.
  - Identified ISO 4217 – Currency Code Services
    - Widely used in banking, commerce, and trade.
    - Three letter code for currencies.
    - Web services available to acquire conversion factors using these codes.
- Two Interfaces proposed:
  - ICapeCurrency exposed by the PMC.
    - One property, Currency that returns the currency code supported by the PMC.
  - ICapeCurrencyExchange exposed by the PME.
    - Members include default currency for the PME, list of currencies known by the PME, and currency conversion methods.
- Currently not implemented in COBIA.

Reporting Interface (Proposed)

- Issues leading to updating reporting interface :
  - Currently Reporting is handled by ICapeUnitReport
    - Reporting is only supported by unit operation PMCs.
  - Format of reports available is currently limited.
    - Work around using Simulation Context.
- Objective to develop a general reporting interface: *ICapeReport*
  - Usable by all types of PMCs.
  - Expand the media types (formats) of reports that may be generated.
    - Standard MIME types.
    - Potential for XML/JSON format that can be used in templated reporting.
    - Minimum reporting requirement is plain text format.
  - Provide means to create reports in different languages.
- Overview of interface design provided to membership for comment.
- This interface has not been implemented in COBIA, and is currently not anticipated to be available in COM-based CAPE-OPEN.

COBIA Project Roadmap

- Phase I – Proof of Concept **Completed**
  - Core technical components
  - Demonstrate COM/COBIA interoperability with Thermo 1.1 interface set
- **Phase II – Full Windows Native Completed**
  - **Expanding COBIA to all interfaces of business value**
  - **Support for C/C++ development.**
  - **Allow development of fully functional COBIA-based PMEs and PMCs**
- **Release of Phase II - Pending**
- **Phase III – Interoperability (Future)**
  - **Microsoft .NET is planned – 2020/2021**
  - **Other platforms as identified by CO-LaN membership**

COBIA Timeline

- October 2016 - Phase I completed
- October 2017 – Phase II status presented and demonstrated
- October 2018 –Beta version of COBIA
  - Testing of COBIA
  - COBIA Training
- October 2019 – Completed Beta testing of COBIA
- In Progress – Implementation of COBIA by KBC and HTRI
- FUTURE
  - Release Phase II Product (Awaiting Management Board Approval of License).
  - Scope and Create Required COBIA Documentation
    - E.g. Transitioning and Threading
  - Develop .NET language bindings
  - Other language bindings (Develop if there is a business case).
  - CO-LaN will maintain COBIA codebase and provide updates as needed.

COBIA 2018/2019 Activities

- End-User Redistributable and SDK installers
  - Specification
  - Implementation
- COBIA Question and Answer session on 21 October.
- Transitioning of persisted objects from COM to COBIA.
- Threading Model
- Revised Interfaces Included in COBIA (Reported above):
  - Parameters Interfaces
  - Persistence Interfaces
  - Reporting Interface

Development of COBIA Threading Model

- **Objective – Enable efficient use of COBIA in multi-threaded applications.**

- **Proposal – Components declare themselves safe for multi-threading**
  - **Restricted**
    - **Only called from a single thread**
    - **Only called from thread that created the object.**
    - **Can use persistence to move from one thread to another**
    - **Can be created in dedicated thread and calls marshaled – Similar to COM calling STA from MTA**
  - **Unrestricted**
    - **Can be called from any thread**
    - **Only one thread can call the object at a time (No re-entrancy)**

- **Documentation still under development.**

Interaction with Interoperability SIG

- **Aiding in the design of CAPE-OPEN Versioning Scheme.**

- **Supported Development/Maintenance Installer Packages.**
  - Diagnosed the merge module issue in per-user installation of the Type Library/Primary Interop Assembly (TLB/PIA) .
    - Destination folder should be a Private Property of the Installer.
    - Can not redirect of TLB/PIA installation to a different directory.
  - Supported the Interop SIG to develop a new TLB/PIA installer incorporating the remedy.
  - This same solution will be applied, as appropriate, to additional CAPE-OPEN Installation package.

- **Provide Flowsheet Monitoring IDL to Interop SIG for inclusion in TLB/PIA.**

2019/2020 Deliverables

- **COBIA**
  - Testing of vendor-developed COBIA Components
  - Finalize and release production version
  - Scoping of Phase III.
    - Threading model development
    - Marshaling
    - Language bindings

- **Develop textual interface specifications and RFC.**
  - Parameters, Persistence, Reporting, Error Handling, Currency
  - Initially released as part of the COBIA IDL, port to COM, as appropriate.

- **Work with Interop SIG**
  - Testing and Evaluation of COBIA
  - Certification tools
  - Installation Packages
  - Versioning