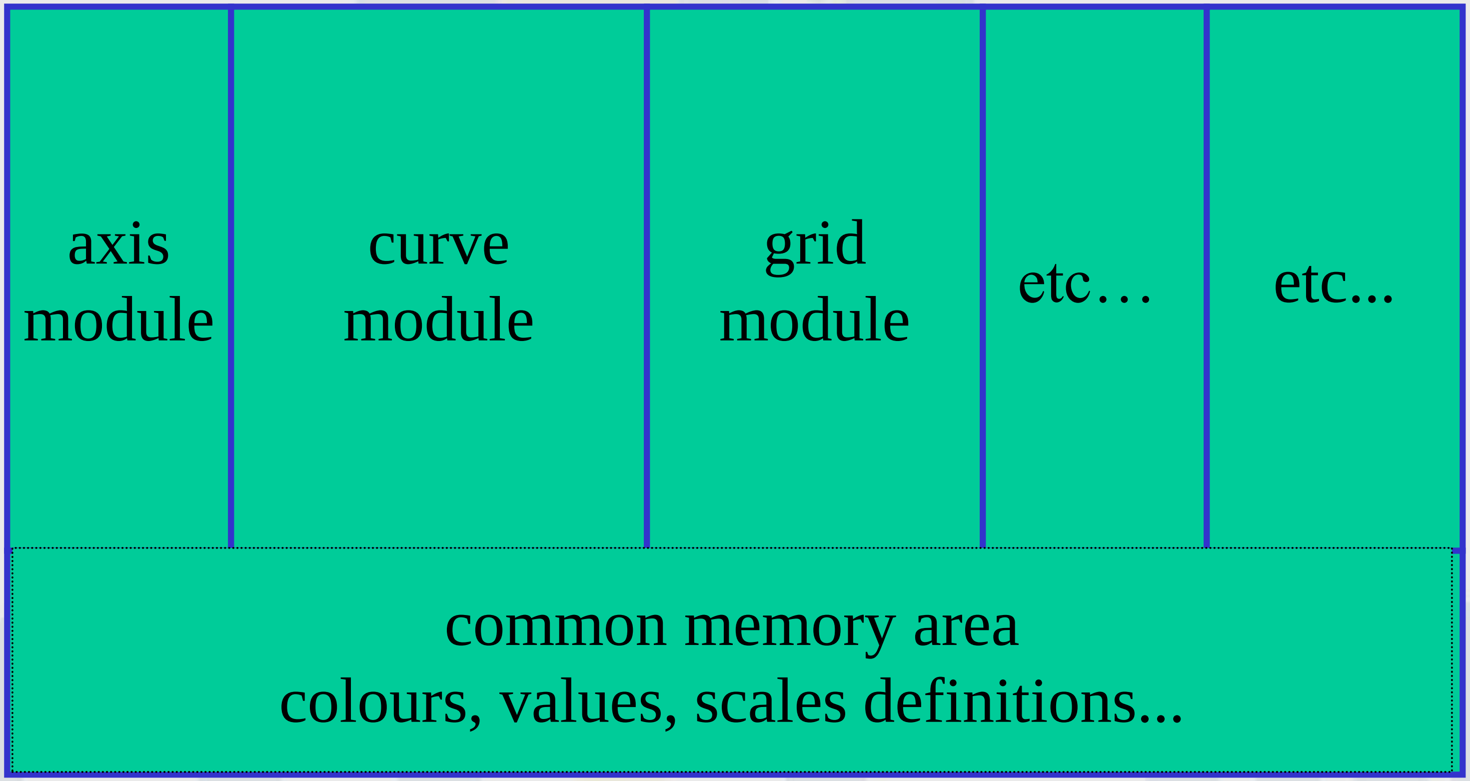
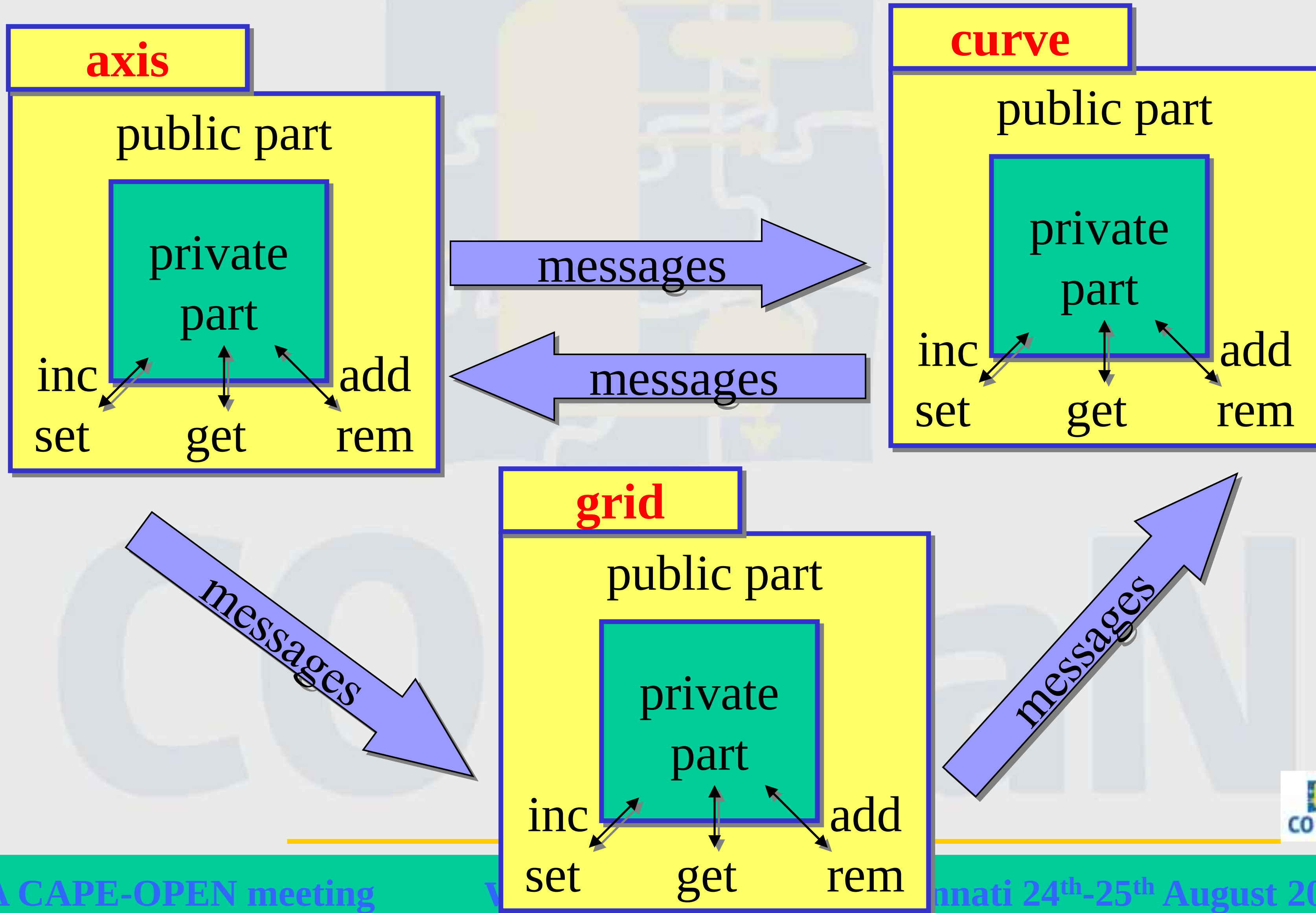
# CAPE OPEN Concepts

Bertrand Braunschweig

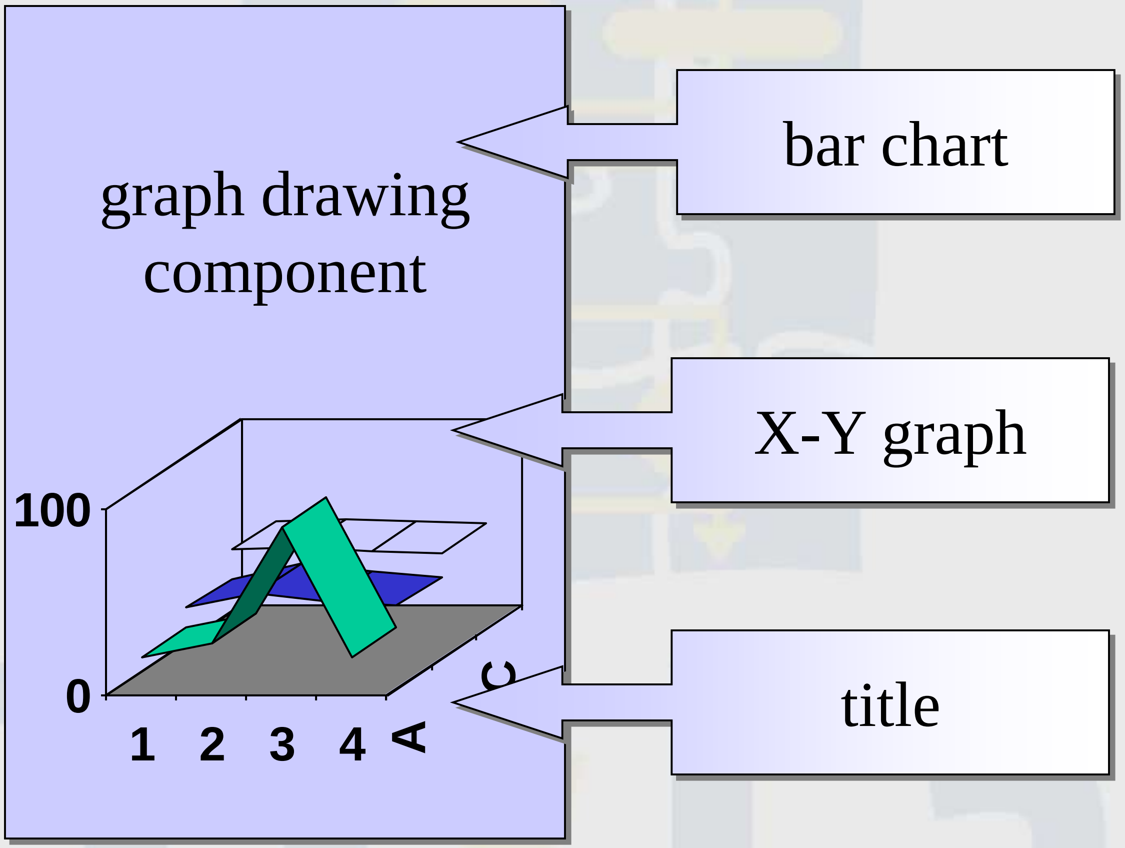
# Monolithic “COMMON”



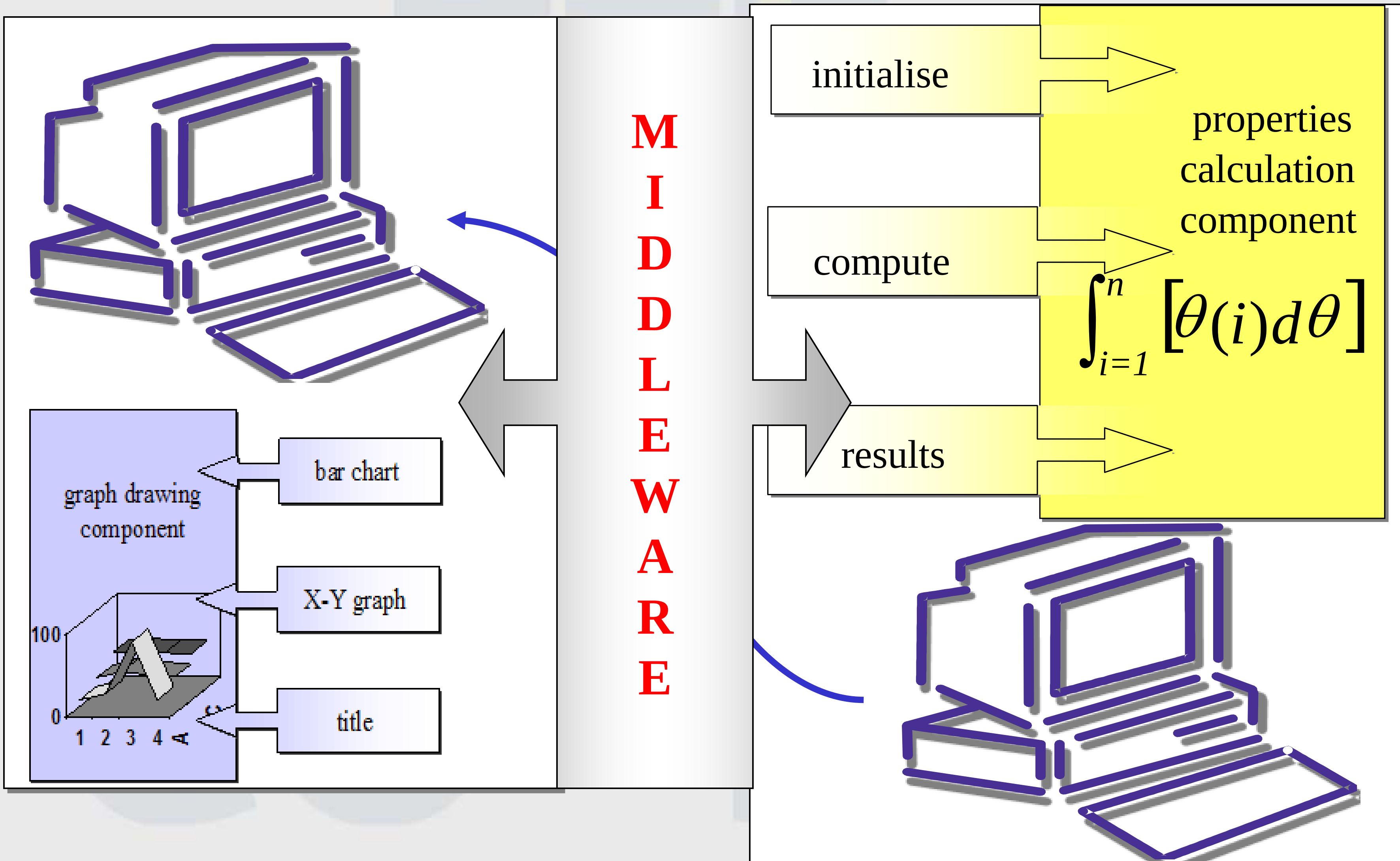
# software objects



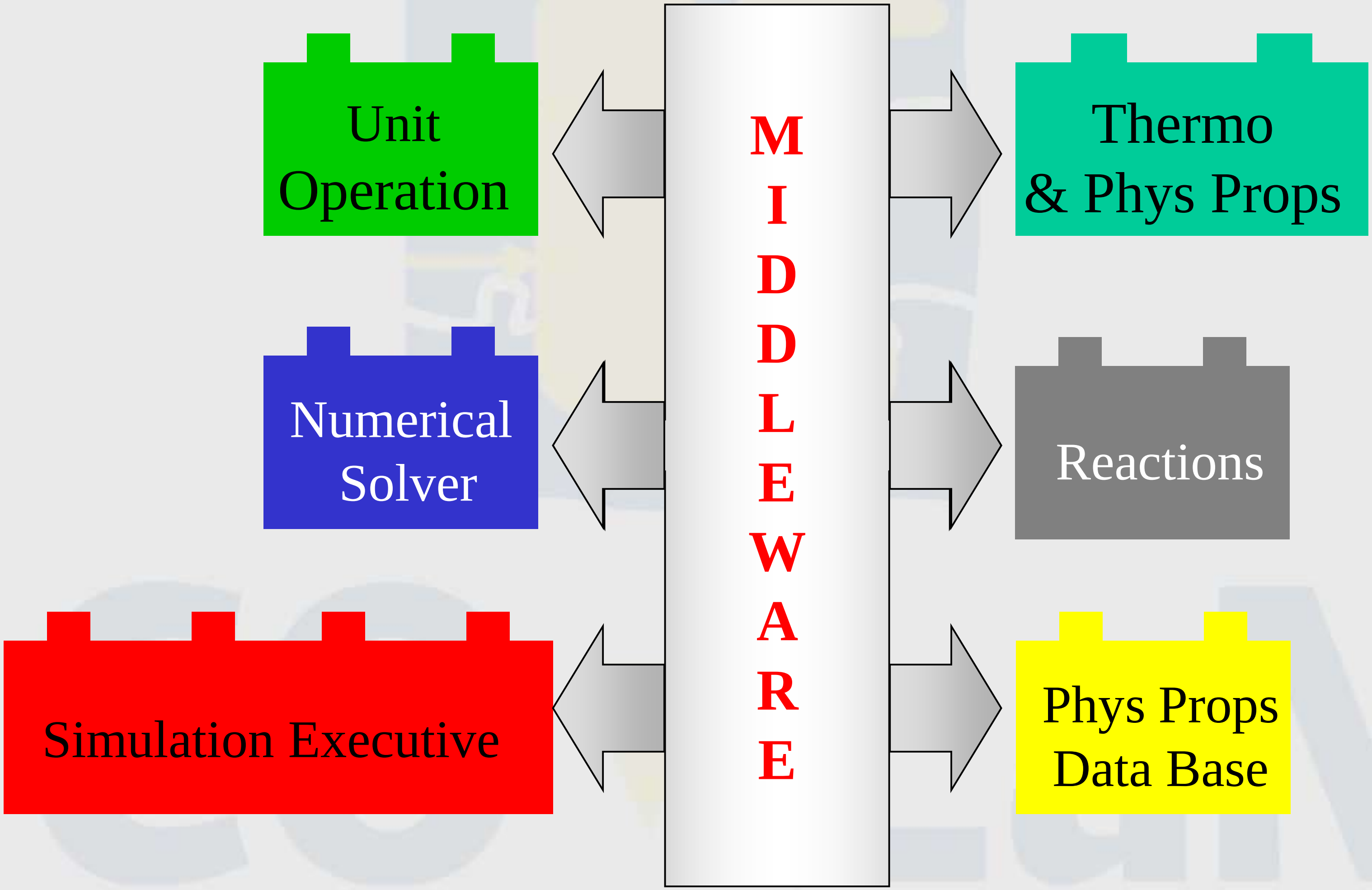
# software components



# distributed software components



# CAPE-OPEN Components



# The CAPE-OPEN Architecture
# Main Interfaces

**Common Services**

**Unit Operations**

**Physical and Thermodynamic Properties**

**Solvers**

# Common Services

# Common Interfaces

Persistence

Parameters

Collections | Utilities

Identification

Error Handling

Types and undefined values

# Unit Operations

# Overview of CO 1.0 Interfaces: Unit Operations

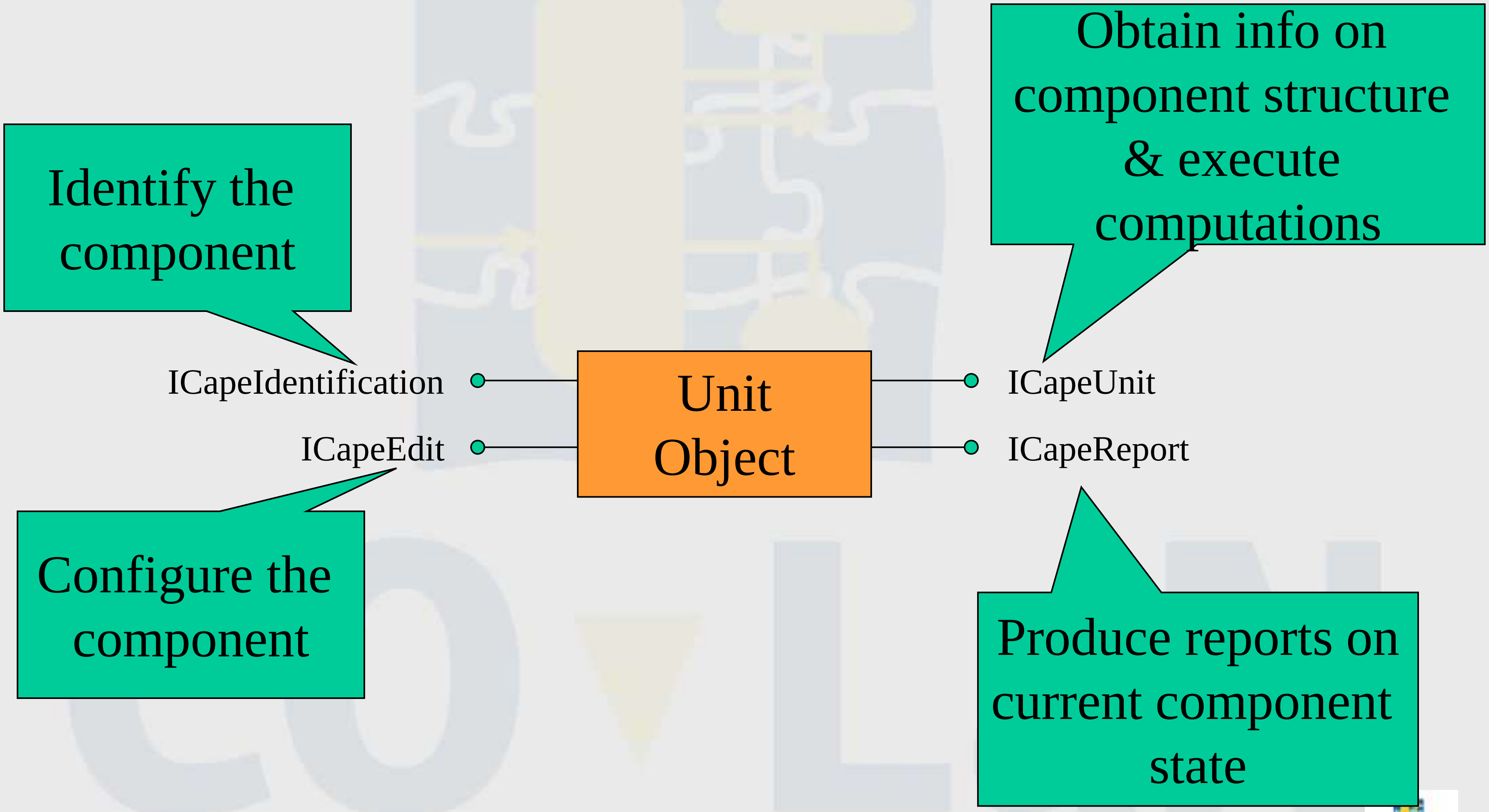
Hybrid Units

Unit Operations
Steady-State (& Dynamic)

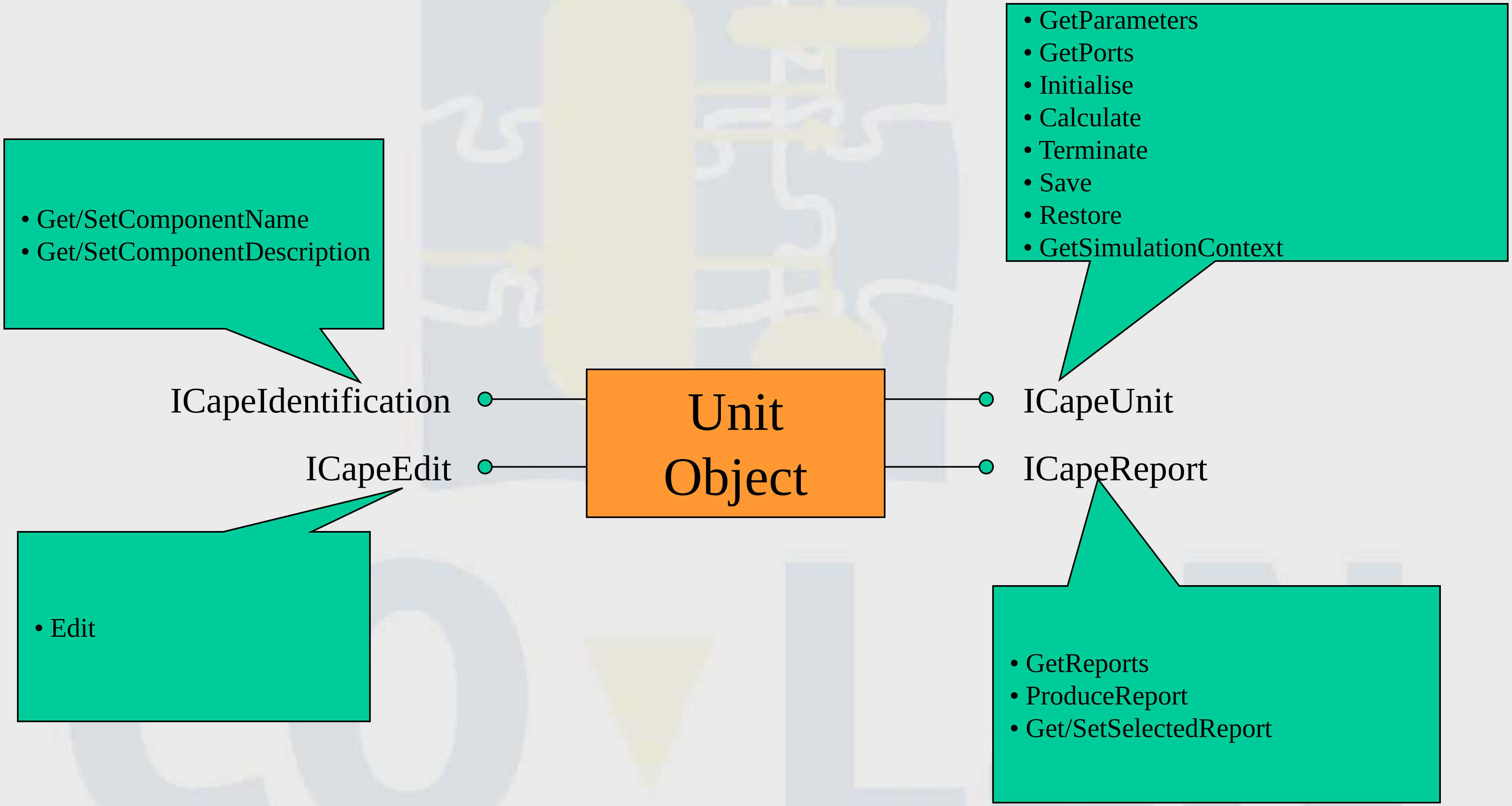
# CAPE-OPEN UNIT OBJECTS

- A "CO unit object" may be
  - a single unit operation
  - a plant sub-section
  - a whole plant
- …connected to its environment via "ports" carrying material, energy or information
  - input ports
  - output ports
- …and characterised by "parameters"
  - input parameters
  - output parameters
- Unit object behaviour
  - given input port information and input parameter values
  - compute output port information and output parameter values

# CAPE-OPEN Unit Object Interfaces



# CAPE-OPEN Unit Object Interfaces



# Thermo & Phys Props

# Overview of CO 1.0 Interfaces: Physical Properties Services

Reactions

Petroleum Fractions

Thermodynamic and Physical Properties

Physical Properties Data Bases

# The main elements of 'Thermo"

- Thermo systems
  - can create ...
- Property packages
  - containing ...
- Equilibrium servers
- Calculation routines

  - to serve ...
- Material objects

# Thermo System

- Manages creation and selection of Property Packages
- Configuration of a PP is outside the scope of CO specs
  - Selection of components models and phases is done by proprietary methods

# Property Package

- Self-contained collection of
  - Chemical components (pure substances)
    - Defined through basic constants eg. Tc
    - Optionally data/correlations on T-dependent properties, eg. Vapour pressure
  - Models/correlations/data for mixture properties
    - Model parameters/BIPs
  - Phases
  - Equilibrium calculation procedures
- Targeted to a particular application
- May be stand-alone or created by a Property System
- Normally a small subset of all the components and models available in a Property System

# Calculation Routine

- A routine that can calculate one or more physical properties
- Typically dedicated to a specific application
- May be supplied as an add-on for a PP
  - Could be an implementation of an in-house method
  - Specialist 3-rd party model, eg. gas sweetening

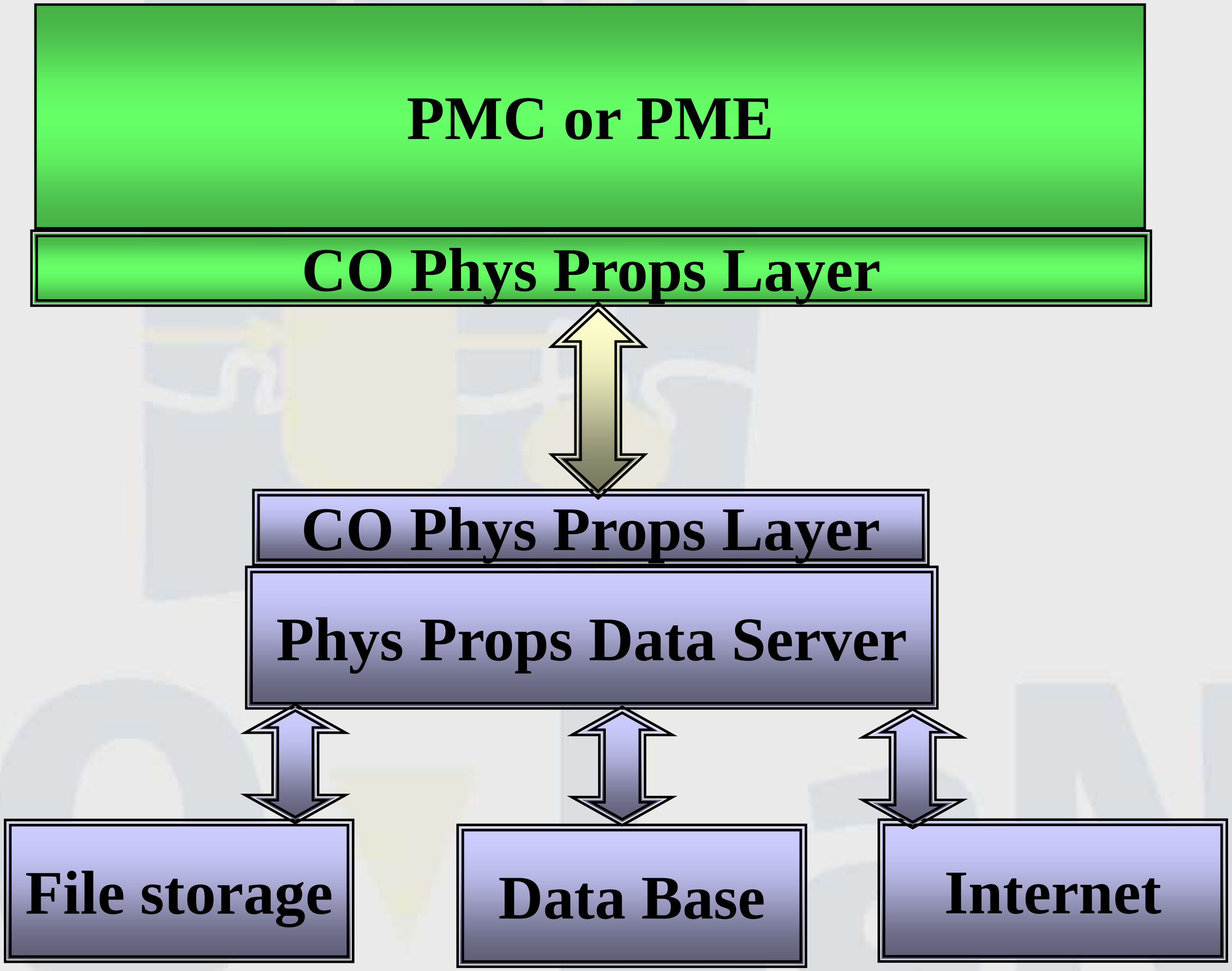
# Equilibrium Server

- Software component that carries out phase equilibrium calculations
- May be implemented as part of PP or stand-alone
- Could be an add-on for a PP
  - Provide specialist capabilities, eg. Hydrate calculations

# MaterialObject

- Container for properties of a material
  - Components
  - Phases
  - Pressure, temperature compositions
  - Other properties…
  - Reference to a PP
- Used by client of Property Package to specify input and collect output from calculations
- Key methods
  - SetProp
    - Property
    - Phase
    - Components
    - calcType (mixture/pure/EMPTY)
    - Basis (mole/mass/EMPTY)
  - GetProp

# Physical Properties Data Bases (PPDB)



# Physical Properties Data Bases (PPDB)

- **Interface for PP Data Bases:**
  - Physical property data at discrete values of the state variables (temperature, pressure, composition)
    - measured, correlated or estimated values
  - Parameters of model equations to be used for calculating data at any desired state

- **The PPDB standard is made of three interfaces**
  - DB management
  - Access to properties in tables
  - Access to model parameters

# Reactions

- Consistent reaction modeling across CAPE-OPEN PMEs
- Reaction Modeling Scope:
  - Support kinetic and equilibrium reactions
  - Support electrolyte reactions
  - Support for any reaction model
  - Support formulation of reaction equations by a client
  - Support Reaction model parameter estimation

# Reactions Interfaces Design

- **Primary CAPE-OPEN Components:**
  - **Reactions Package**
    - defines system of reactions involving specified compounds existing in specific phases
    - able to calculate reaction properties for these reactions
  - **Reactions Package Manager**
    - Manages Reaction Packages
    - May allow Reactions Packages to be created and/or edited
- **PME software objects**
  - **Reactions Object**
    - supports exchange of reaction property values between clients and components

# Petroleum Fractions Interfaces

- Extensions of Thermo and Unit interfaces
- ICapeThermoPetroFractions implemented on the Material Objects
  - Allows setting and getting refinery properties (bulk, curves and component properties)
  - Allows characterising a set of petroleum fractions (e.g. estimate Tc, Pc, etc)
- ICapeUnitTypeInfo implemented on the UNITs
  - Informs COSEs whether a UNIT will require characterization of petroleum fractions (e.g. FCC)

# Numerical Solvers

# Overview of CO 1.0 interfaces: solvers services and clients

PEDR

| Optimisation MILP, MINLP | PDAE Solvers |
|---|---|

Solvers
LAE, NLAE, DAE

# CAPE-OPEN scope for numerical solvers

- Variety of core model-based activities
  - Steady-state & dynamic simulation
  - Steady-state optimisation
  - Parameter estimation and data reconciliation
- Both “modular” and “equation-orientated” systems
- Emphasis on “large-scale” problems

# A wide variety of mathematical problem types depending on application

- **Steady-state simulation**
  - Sets of nonlinear algebraic equations
  - Sets of integral, partial differential and algebraic equations
- **Dynamic simulation**
  - Sets of differential & algebraic equations
  - Sets of integral, partial differential and algebraic equations
- **Plant design**
  - (mixed integer) nonlinear programming
  - (mixed integer) dynamic optimisaiton
- **Plant operation and control**
  - Linear programming
  - Nonlinear programming
  - Dynamic optimisation
- **Production planning & scheduling**
  - (mixed integer) linear programming

# A hierarchy of solvers

- To do steady-state simulation of a distillation column
  - we need to solve sets of nonlinear algebraic equations…
    - which involves solving linear equations…
- To optimise the grade transition in a polymerisation reactor
  - we need to solve a dynamic optimisation problem
    - which involves solving sets of DAEs
      - which involves solving linear algebraic equations
      - …and also solving nonlinear algebraic equations (2 different types)
        - which involves solving linear equations
    - .. and also nonlinear programming problems
      - which involves solving (more) linear equations…

# CAPE-OPEN Problem Objects

- **Fundamental principle: complete separation between**
  - **the description of the problem being solved**
  - **the code used for its solution**
- **Describe different types of mathematical problems as different classes with formally defined interfaces**

| Mathematical Problem Type | CAPE-OPEN Problem Object |
|---|---|
| Nonlinear algebraic equations | Equation Set Object (**ESO**) |
| Differential-algebraic equations | Differential Algebraic ESO (**DAESO**) |
| Partial differential-algebraic equations | Partial Differential Algebraic ESO(**PDAESO**) |
| Mixed integer nonlinear programming problems | **minlp** object |

# CAPE-OPEN Systems

$$\left\{\begin{matrix}\text{CAPE}-\text{OPEN}\\ \text{System}\end{matrix}\right\} \equiv \left\{\begin{matrix}\text{Problem}\\ \text{being solved}\end{matrix}\right\} + \left\{\begin{matrix}\text{Numerical code}\\ \text{used for solution}\end{matrix}\right\}$$

- **Each CAPE-OPEN System has method(s) for solving the problem**
- **Final solution of the problem is placed in the CO Problem Object**

# The CAPE-OPEN Standard:
# What it permits

# Example of use 1

- **A physical and thermodynamic properties calculations PMC** developed by a supplier**, can be used the same way within several CO-PMEs**.
- e.g. Infochem's , Multiflash, can be used the same way in Aspen+, gPROMS or Hysys.
- The user saves the time needed to configure the properties calculations parameters for those environments, and gets consistent results by using the same methods and data.
- This is simply obtained by wrapping the thermo server with CAPE-OPEN standard interfaces.

# Example of use 2

- A CO-compliant PME **can transparently use several physical properties and thermodynamic servers** for one model.
- e.g. Hysys can be configured to use Aspentech's Properties Plus, or Infochem 's Multiflash, or IFP 's SPIP proprietary thermo.
- This can be through replacing a single thermo server for the whole flowsheet, or even by combining different servers for different sections of the flowsheet (with precautions on the enthalpy basis).
- Thus, the modeller can easily try out diverse methods and choose the best
- This is obtained by **introducing the CO « Thermo » API** in the PME.

# Example of use 3

- **A Unit Operation model** such as a proprietary chemical reactor model, developed by an operator or a process licensor, **can be used transparently in CO-compliant PMEs**.
- e.g. IFP 's FIBER (FIxed BEd Reactor) generic reactor model can be used the same way in most commercial PMEs without any change, whitout any coding or compiling.
- The process licensor can easily serve clients who demand the use of a specific PME in their contracts.
- This is obtained from putting the reactor model to the Unit Operation standard: introduction in a flowsheet, connection of input-output ports, specification of parameters, validity checking, calculation, publication of results.

# Example of use 4

- Reciprocally, a modeller who uses a PME with CO Unit Operations « sockets » can **seamlessly include foreign unit operation models** by selecting from a list of available CO-compliant Unit Operations.

- This, the model designer can easily test several equipment models and choose the best equipment (a compressor, a heat exchanger, a pump etc.) for a specific process.

- This imposes that all equipment models are available on the user's machine. in the future, this will be possible though component identification services developed for the internet.

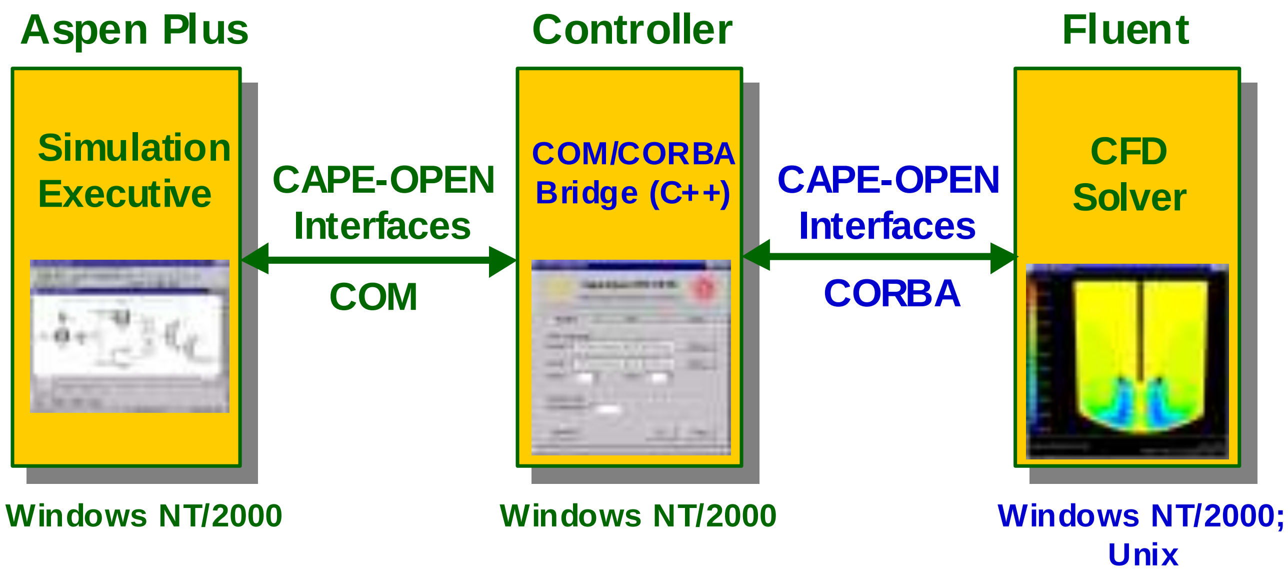
- Equipment manufacturers should take advantage of this facility.

# Example of use 5: GO:CAPE-OPEN

## gO:CAPE-OPEN overview

- Introduce advanced gPROMS models within CAPE-OPEN compliant steady-state flowsheeting packages e.g.
  - ASPEN PLUS™
  - HYSYS™
- Use consistent physical properties throughout
- No programming required
  - retain advantages of gPROMS-based modelling

# COM/CORBA Bridge




<u>Advantages</u>

- Fast bi-directional data exchange using inter-process communication
- Fluent process starts up and remains active
- Fluent runs on Windows and Unix systems

# Other uses

- **More than 10 published interfaces**
  - **numerical solvers**
  - **chemical reactions**
  - **physical properties data banks**
  - **etc.**

- **Same kind of facilities as presented in other examples**

# CAPE OPEN Concepts

Bertrand Braunschweig