Technische Universität Berlin

The MOSAIC Approach – Self-Made CO-UOs Without Programming Knowledge

Gregor Tolksdorf, M.Sc. | Faculty of Process Sciences | CAPE-OPEN 2015 Annual Meeting

MOSAIC

Modular **MO**del **S**pecific**A**t**I**on on Do**C**umentation Level -

Application in a Web Based Modeling Environment.

www.mosaic-modeling.de

d|b|t|a
www.dbta.tu-berlin.de

Technische Universität Berlin

www.mosaic-modeling.de

MOSAIC

Fachgebiet Dynamik und Betrieb technischer Anlagen

News | MOSAIC | Documentation | Solvers & Environments | Publications | Contact | Impressum | Forums

Home » Update to Version 1.8.1

MOSAIC Hotlinks

- Download & Run MOSAIC
- Account and Feature Request

MOSAIC on YouTube

MOSAIC - The M...

MOSAIC

MOSAIC is a free, web-based modeling, simulation, and optimization environment. Based on a LaTeX-style entry method for algebraic and differential equations, equation systems can be built and

Update to Version 1.8.1

18 Sep, 2015 in News by Gregor Tolksdorf

A new version of MOSAIC (1.8.1) is available now. New features, improvements, and fixes in MOSAIC 1.8.1: improvement: Nicer images for the flowsheet module improvement: User-defined langspec tests in advance if it is applicable for code generation of the selected evalution new: code generation properties now available for user-defined language [...]

MOSAIC Workshop Bogota 2015

7 Sep, 2015 in News by Erik Esche

The latest MOSAIC workshop started on Tuesday, 8th of September 2015, in Bogota, Colombia. Access to the course material can be gained through TU Berlin's ISIS system: MOSAIC Workshop Bogota 2015 on ISIS 29 highly motivated students have worked their way through MOSAIC features such as function applications, user-defined discretisation of [...]

MOSAIC

MOSAIC-Team

Scientific supervisor:

Prof. Dr.-Ing. G. **Wozny**,

Head of Chair of Process Dynamics and Operation, TU Berlin

d|b|t|a

Current Developer:

Dr.-Ing.	E. **Esche**
Dipl.-Ing.	S. **Fillinger**
Dipl.-Ing.	V.A. **Merchan-Restrepo**
M.Sc.	G. **Tolksdorf**

d|b|t|a

www.dbta.tu-berlin.de

Agenda

1. Motivation – The MOSAIC Approach
2. Unit Operations & Physical Properties
3. Code Generation
4. Application Example
5. Summary & Outlook

Self-Made CO-UOs Without Programming Knowledge

MOTIVATION – THE MOSAIC APPROACH

MOSAIC – Why?

Models and their application change over time.

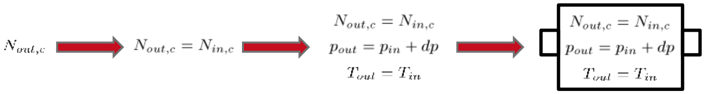
Problems:
- Documentation is outdated
- Reimplementation is error-prone

Eutopia: Process Engineer == Programming Expert

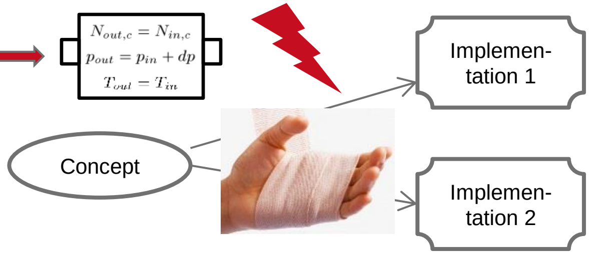
MOSAIC offers:
- Transparent model documentation
- Automatic code generation

MOSAIC + CAPE-OPEN Advantages

- Encouraged systematic modeling

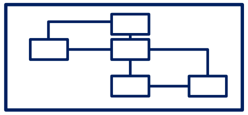


- Eliminates redundant, error-prone manual implementation



- Enables usage of plattform-independent models in virtually any simulation software based on
  - Equations (z.B. PSE, …)
  - Flowsheeting (z.B. Aspentech, Pro Sim SA, Honeywell Process Solutions, Amsterchem, …)

$$\boxed{\begin{aligned} N_{out,c} &= N_{in,c} \\ p_{out} &= p_{in} + dp \\ T_{out} &= T_{in} \end{aligned}}$$



Vision: « Model once, simulate anywhere »

*Image "Hand being bandaged as injury" courtesy of Stuart Miles / FreeDigitalPhotos.net

d|b|t|a
www.dbta.tu-berlin.de

Technische Universität Berlin

Agenda

1. Motivation – The MOSAIC Approach
2. Unit Operations & Physical Properties
3. Code Generation
4. Application Example
5. Summary & Outlook

Self-Made CO-UOs Without Programming Knowledge

UNIT OPERATIONS & PHYSICAL PROPERTIES

MOSAIC – Unit Operations

A Unit Operation in MOSAIC is…

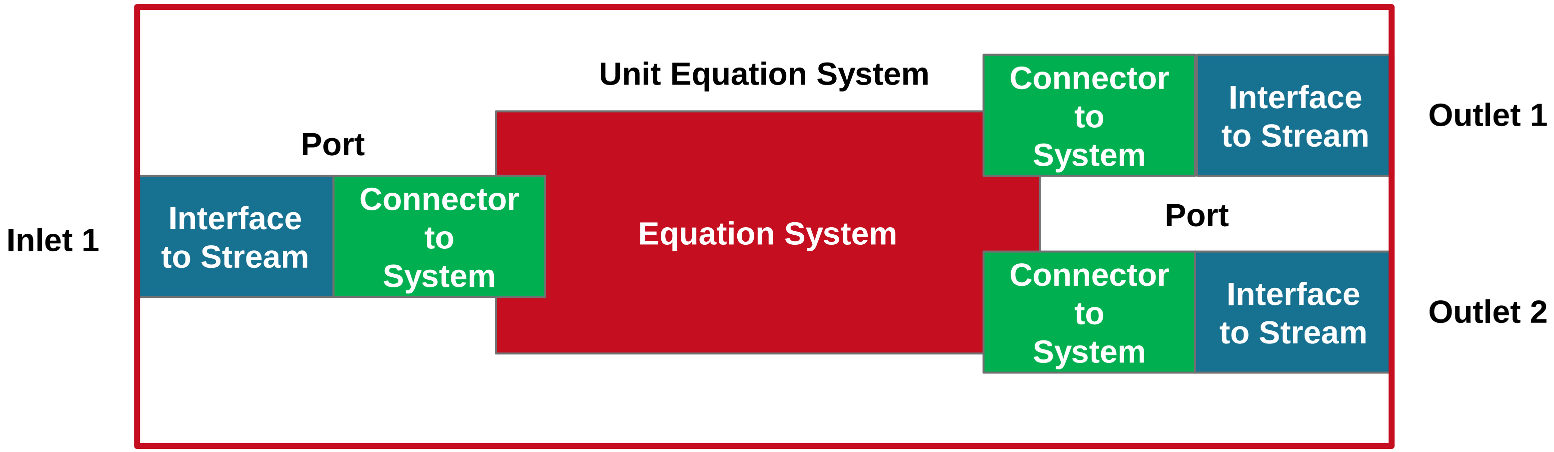
$$N_{out,c} \Rightarrow N_{out,c} = N_{in,c} \Rightarrow \begin{array}{c} N_{out,c} = N_{in,c} \\ p_{out} = p_{in} + dp \\ T_{out} = T_{in} \end{array} \Rightarrow \boxed{\begin{array}{c} N_{out,c} = N_{in,c} \\ p_{out} = p_{in} + dp \\ T_{out} = T_{in} \end{array}}$$

…an equation system with ports.

MOSAIC-Ports translate variables and are connection points for streams.

MOSAIC – Units and Flowsheets

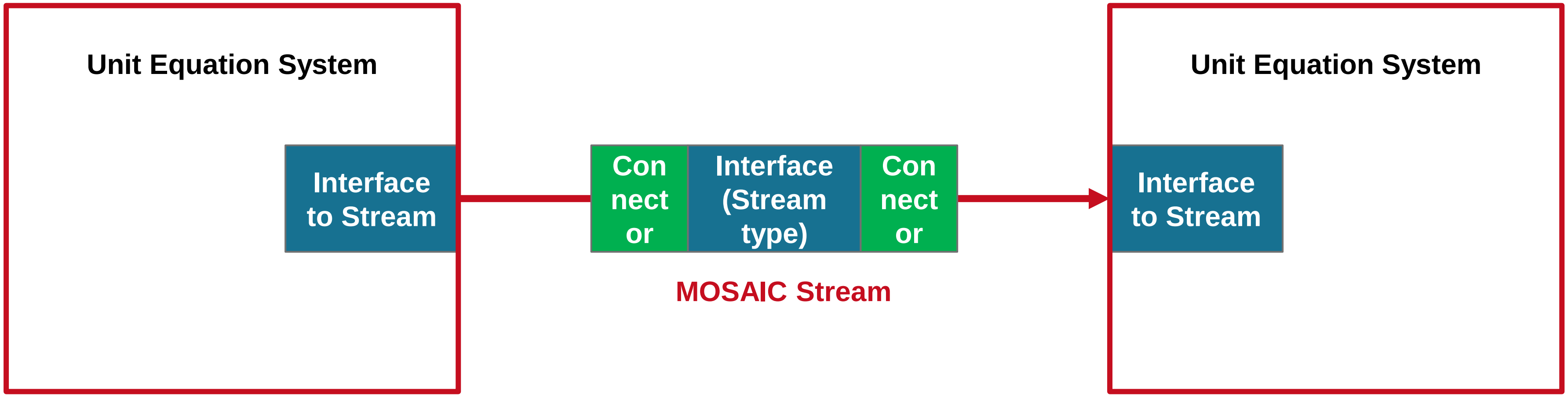
The Definition of Ports:



Connector connects variables from the System to the Interface

Interface defines the Stream Type (Material, Energy, etc.)

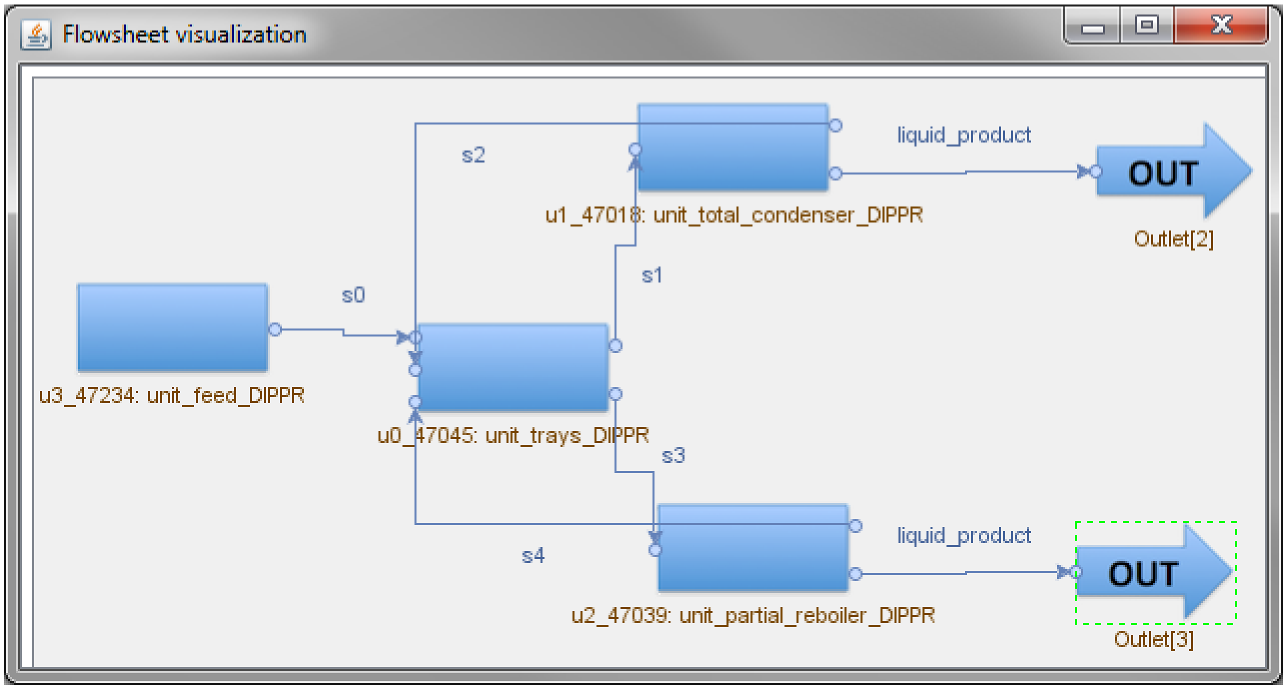
MOSAIC – Units and Flowsheets



Optional Connectors for Conversion

MOSAIC – Units and Flowsheets

MOSAIC Unit Operations: „Equation-based flowsheeting“



Technische Universität Berlin

MOSAIC – Physical Properties

(External) physical properties in MOSAIC are…

Function

Location: 12513: CO vapor pressure function.mosfun

$p_i^{\mathrm{LV}}(T) = \mathrm{CO\ Calculate\ Vapor\ Pressure}$

Description: Vapour pressure with CAPE OPEN

No. of usages: 3

…variables that are calculated by external, language-specific functions.

Example code (using Amsterchem‘s „Matlab CAPE-OPEN Thermo Import“):

```
function[std_p_LV_iALL] = fun_12513__co_vapor_pressure_function(std_T)
                    global co_handle;
                    std_p_LV_iALL = capeOpenTDepProp(co_handle,'vaporPressure',std_T);
end
```

Agenda

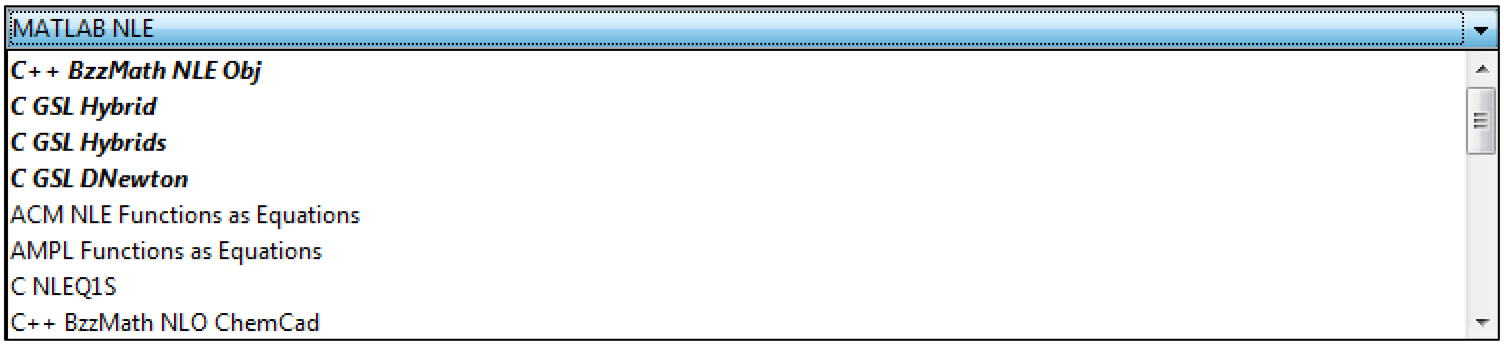
1. Motivation – The MOSAIC Approach
2. Unit Operations & Physical Properties
3. Code Generation
4. Application Example
5. Summary & Outlook

Self-Made CO-UOs Without Programming Knowledge

CODE GENERATION

MOSAIC – Code Generation

MOSAIC offers code generation for several **predefined** languages:

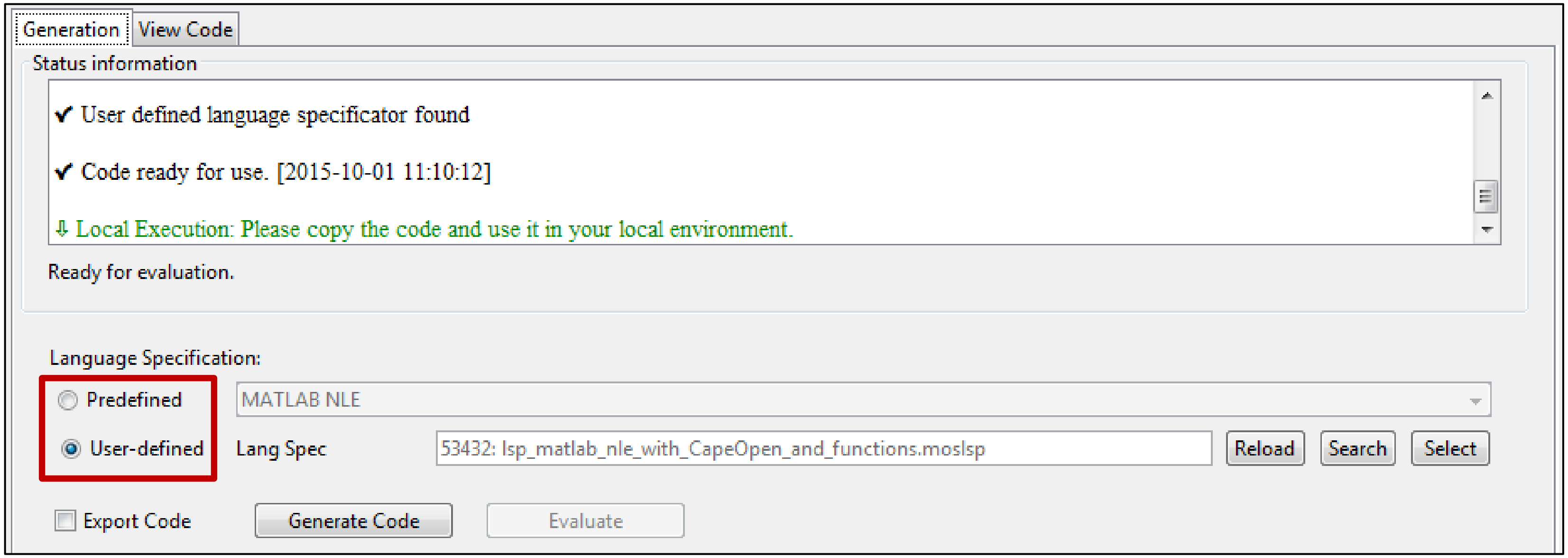


The predefined languages cover

- general purpose programming languages
- specific environments for simulation
- optimization languages

MOSAIC – Code Generation

MOSAIC users can define code generators for **new** languages.



Example:
Code generator for Amsterchem's „Scilab CAPE-OPEN Unit Operation"

Agenda

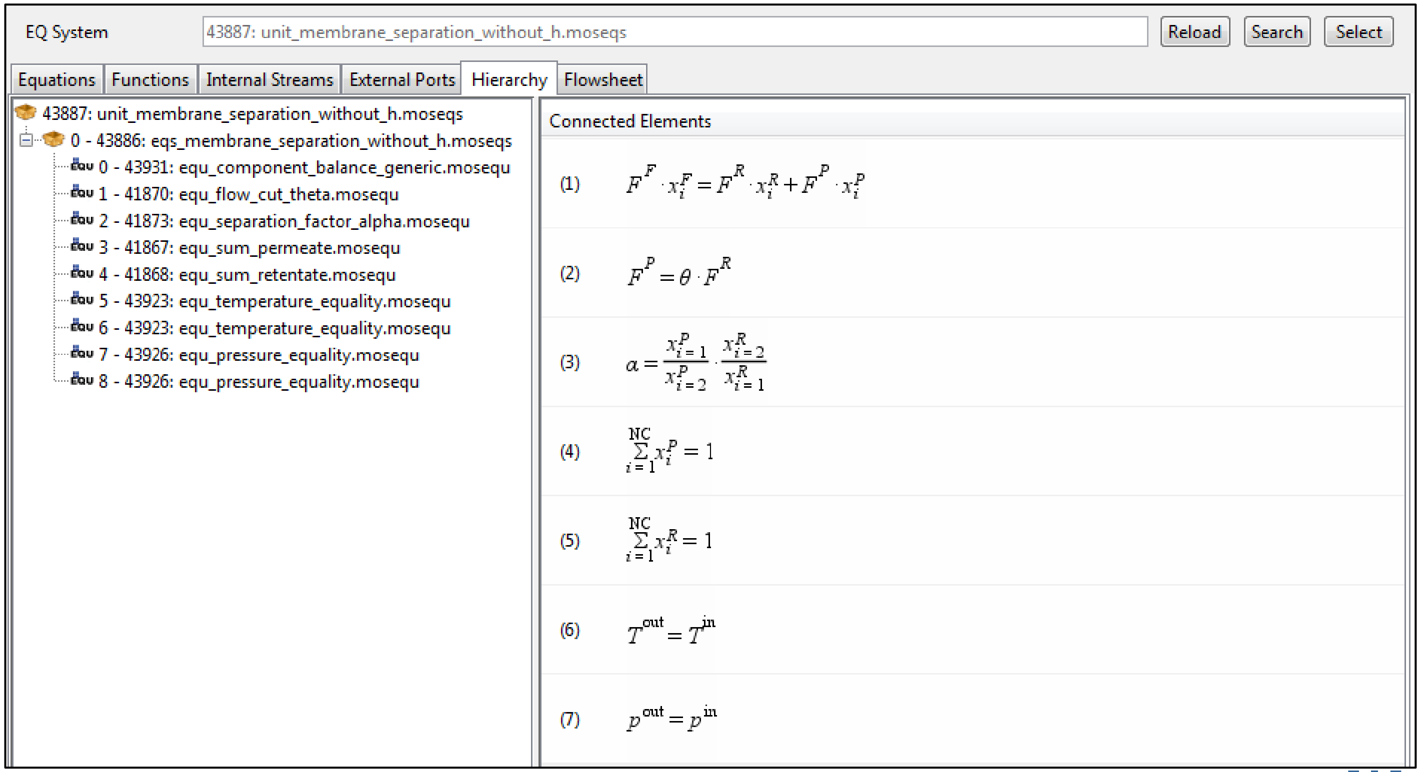
1. Motivation – The MOSAIC Approach
2. Unit Operations & Physical Properties
3. Code Generation
4. Application Example
5. Summary & Outlook

Self-Made CO-UOs Without Programming Knowledge

APPLICATION EXAMPLE

Application Example - Simple Membrane Separation

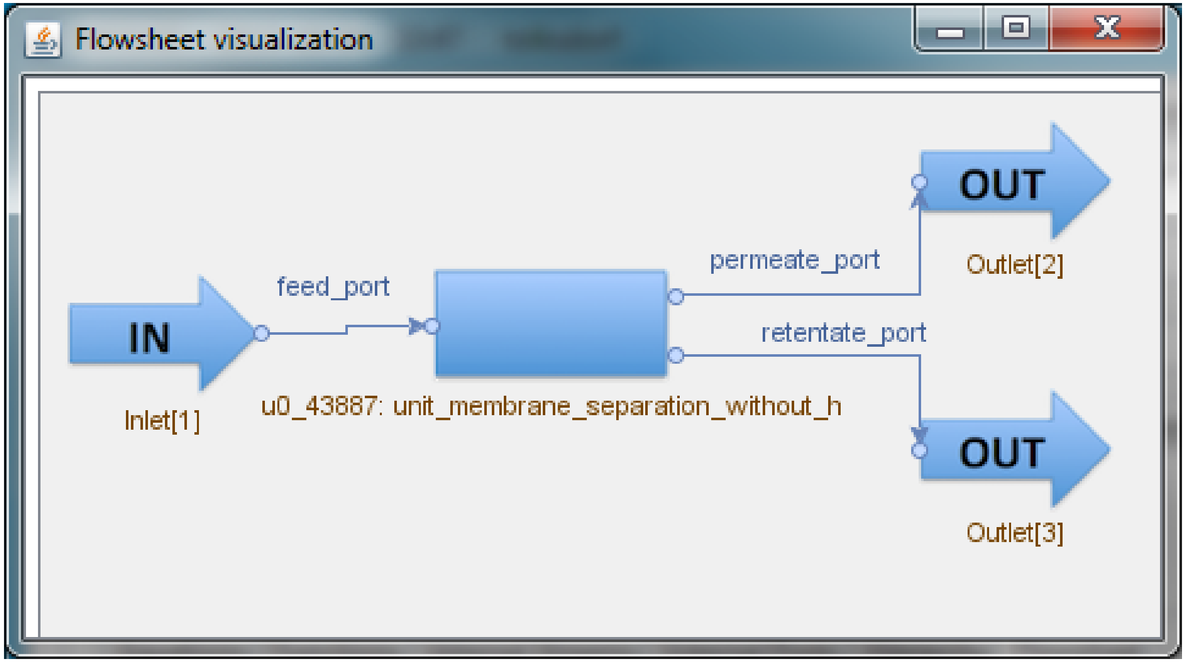
MOSAIC:



***van Baten, Taylor, Kooijman: „Using Chemsep, COCO and other modeling tools for versatility in custom process modeling“ ,AIChE 2010**

Application Example - Simple Membrane Separation

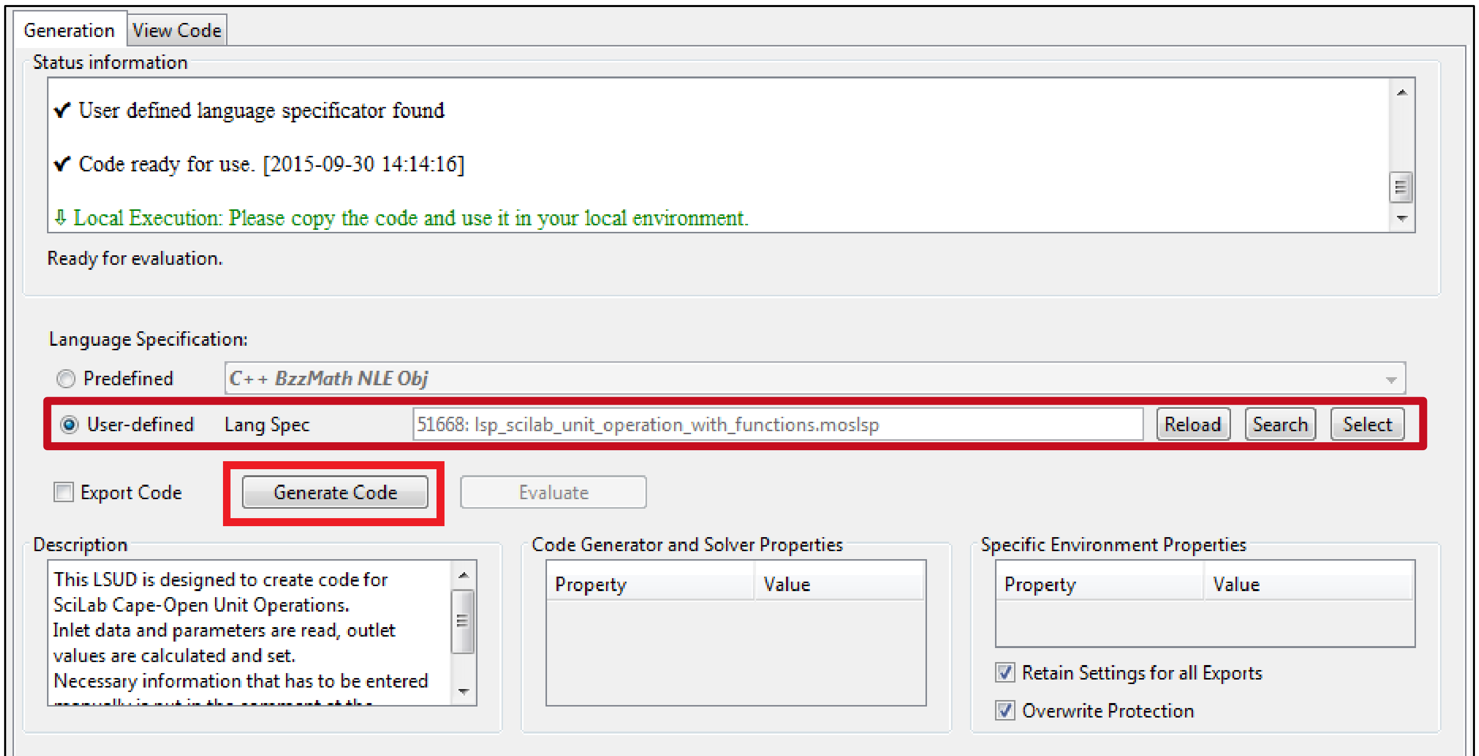
MOSAIC:



***van Baten, Taylor, Kooijman: „Using Chemsep, COCO and other modeling tools for versatility in custom process modeling" ,AIChE 2010**

Application Example - Simple Membrane Separation

MOSAIC:



*van Baten, Taylor, Kooijman: „Using Chemsep, COCO and other modeling tools for versatility in custom process modeling“ ,AIChE 2010

Application Example - Simple Membrane Separation

MOSAIC:

Generation | View Code

Generated Code: Export Code

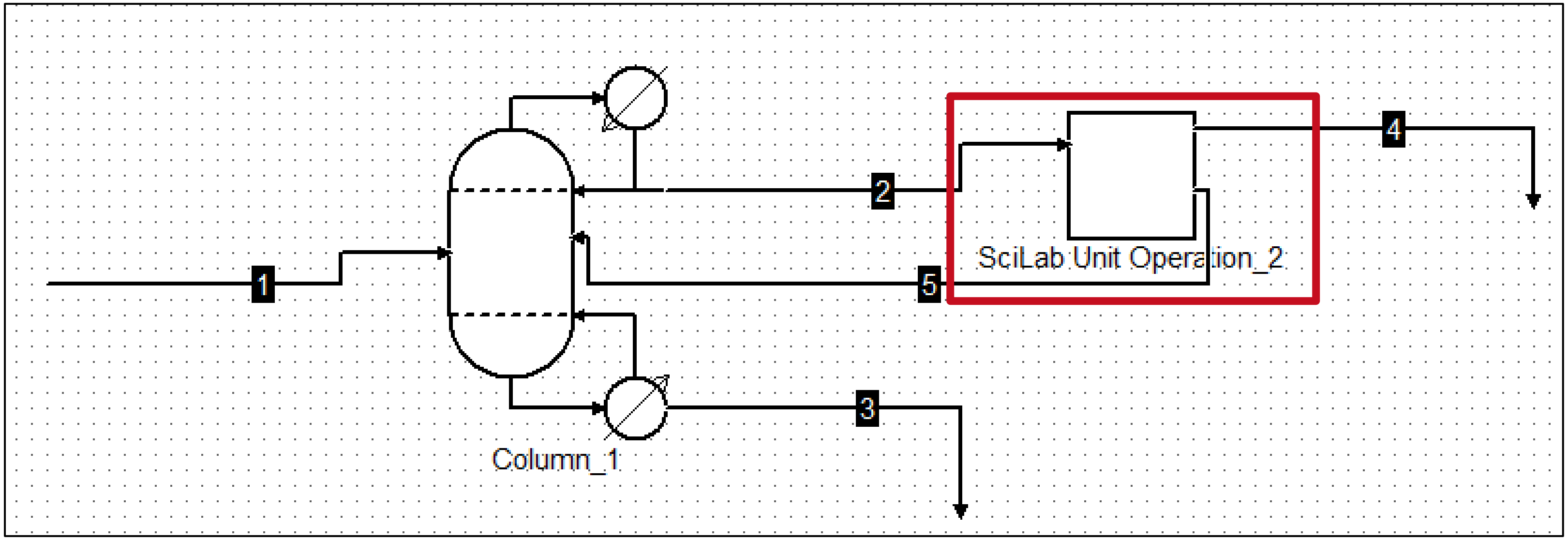
```
//****************************************************************
// Scilab output for algebraic equation systems
//
// Copy this code into Amsterchems SciLab Unit Operation
//
// Code generated by the help of MOSAIC
//
// Please enter the following inlet ports in the Ports tab:
//                    feed_port
//
// Please enter the following outlet ports in the Ports tab:
//                    permeate_port
//                    retentate_port
//
// Please enter the following parameters in the Parameters tab:
//                    f0_greek_theta                    default value:  0.6
//                    f1_greek_alpha                    default value:  70.0
//
//****************************************************************


function[Y] = getFunVal(X_ITER,PARAMS)
//
// Calculate the function value of a normalized equation system.
//
                    // read out variables
```

Code Export

*van Baten, Taylor, Kooijman: „Using Chemsep, COCO and other modeling tools for versatility in custom process modeling“ ,AIChE 2010

Application Example - Simple Membrane Separation

COCO/COFE:

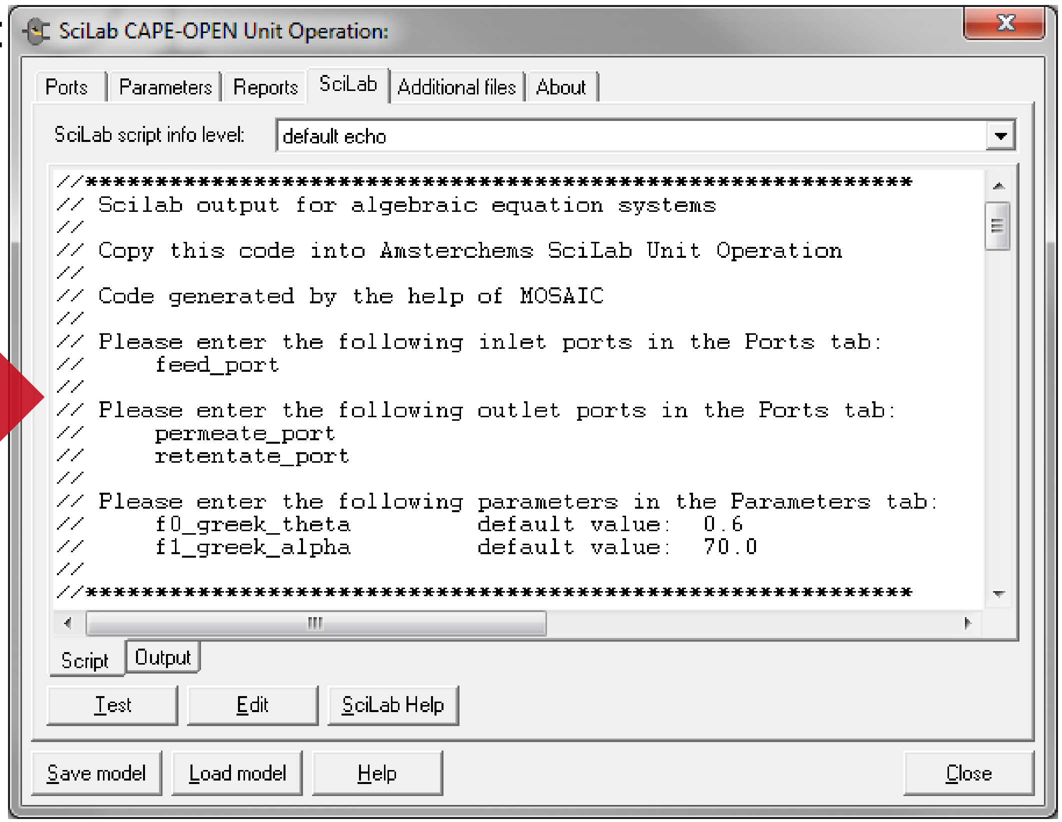


*van Baten, Taylor, Kooijman: „Using Chemsep, COCO and other modeling tools for versatility in custom process modeling" ,AIChE 2010

Application Example - Simple Membrane Separation

COCO/COFE:

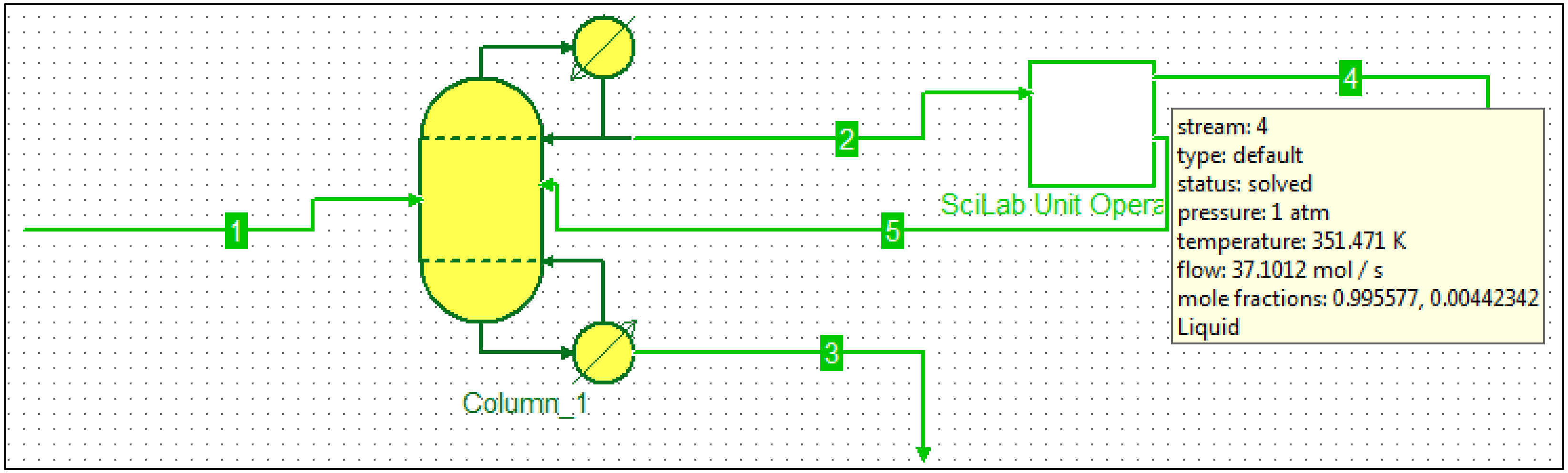
Code Export



***van Baten, Taylor, Kooijman: „Using Chemsep, COCO and other modeling tools for versatility in custom process modeling" ,AIChE 2010**

Application Example - Simple Membrane Separation

COCO/COFE:



*van Baten, Taylor, Kooijman: „Using Chemsep, COCO and other modeling tools for versatility in custom process modeling" ,AIChE 2010

Technische Universität Berlin

Agenda

1. Motivation – The MOSAIC Approach
2. Unit Operations & Physical Properties
3. Code Generation
4. Application Example
5. Summary & Outlook

Technische Universität Berlin

Self-Made CO-UOs Without Programming Knowledge

SUMMARY & OUTLOOK

d|b|t|a
www.dbta.tu-berlin.de

Summary & Outlook

MOSAIC

- is an equation-based modeling and code generation tool
- covers unit operations and physical property calls
- automatically generates code for various programming languages, including
  - MatLab/SciLab Cape-Open UnitOperations by Amsterchem

Summary & Outlook

Vision:

« Model once, simulate anywhere »

by creating CO-UOs with MOSAIC

Next steps:

- **C++ code generation for Cape-Open unit operations**
- **Direct delivery of a Cape-Open unit operation shared library (DLL)**

d|b|t|a
www.dbta.tu-berlin.de

Thank you very much for your kind attention.

www.mosaic-modeling.de

Acknowledgement:

This project is

- supported by the Cluster of Excellence 'Unifying Concepts in Catalysis'
- coordinated by the Technical University of Berlin and
- funded by the German Research Foundation.

Self-Made CO-UOs Without Programming Knowledge

BACKUP SLIDES

Goals and Resulting Characteristics

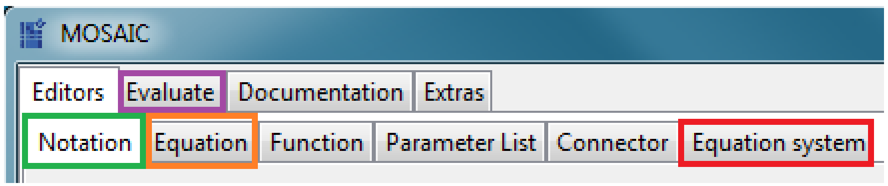
Main goals:

- Less errors
- Less effort
- More cooperative work
  - Improved reuse
  - Improved portability

Resulting characteristics:

- Highly **modular** modeling concept
- Define Platform Independent Models (PIM) in the documentation level using an **enhanced symbolic notation**
- Use of PIM and **code generation** to Platform Specific Models (PSM)
- Support web-cooperation
  - Store and share all model elements in a **web database**

Modular Modeling Concept – The Editors



MOSAIC editors and model elements:

Notation – What symbols/variables are allowed?

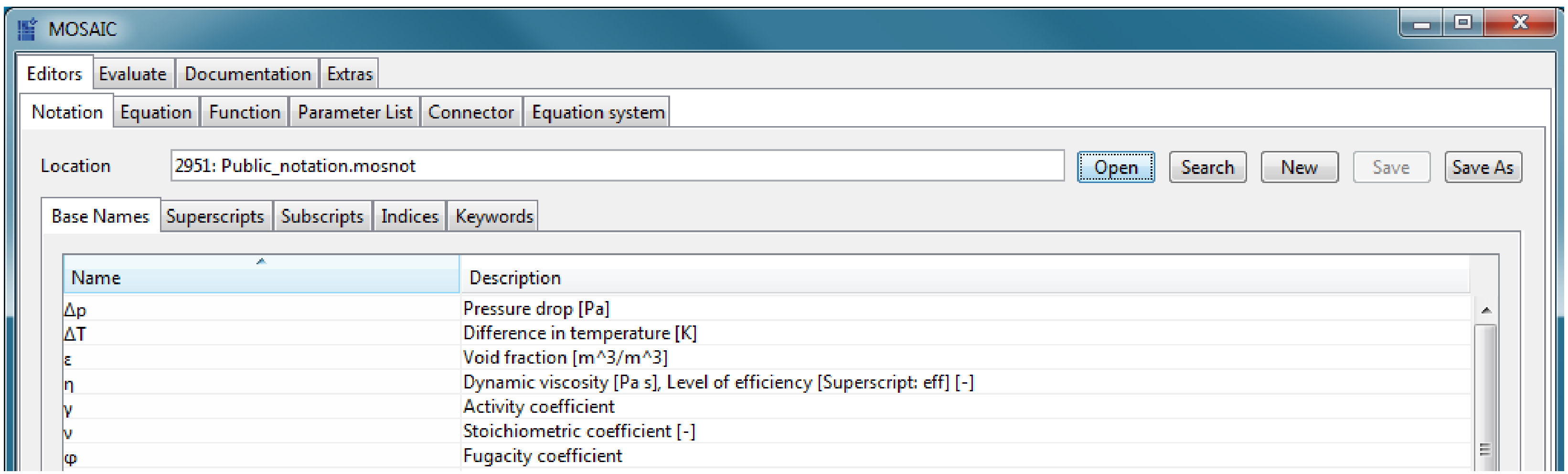
Equation – What equations will be used?

EquationSystem – How will the equations be combined? What functions will be used?

Evaluation – What are the design, state, and iteration values? How does the problem solving code look like?

Enhanced Symbolic Notation I - Variables

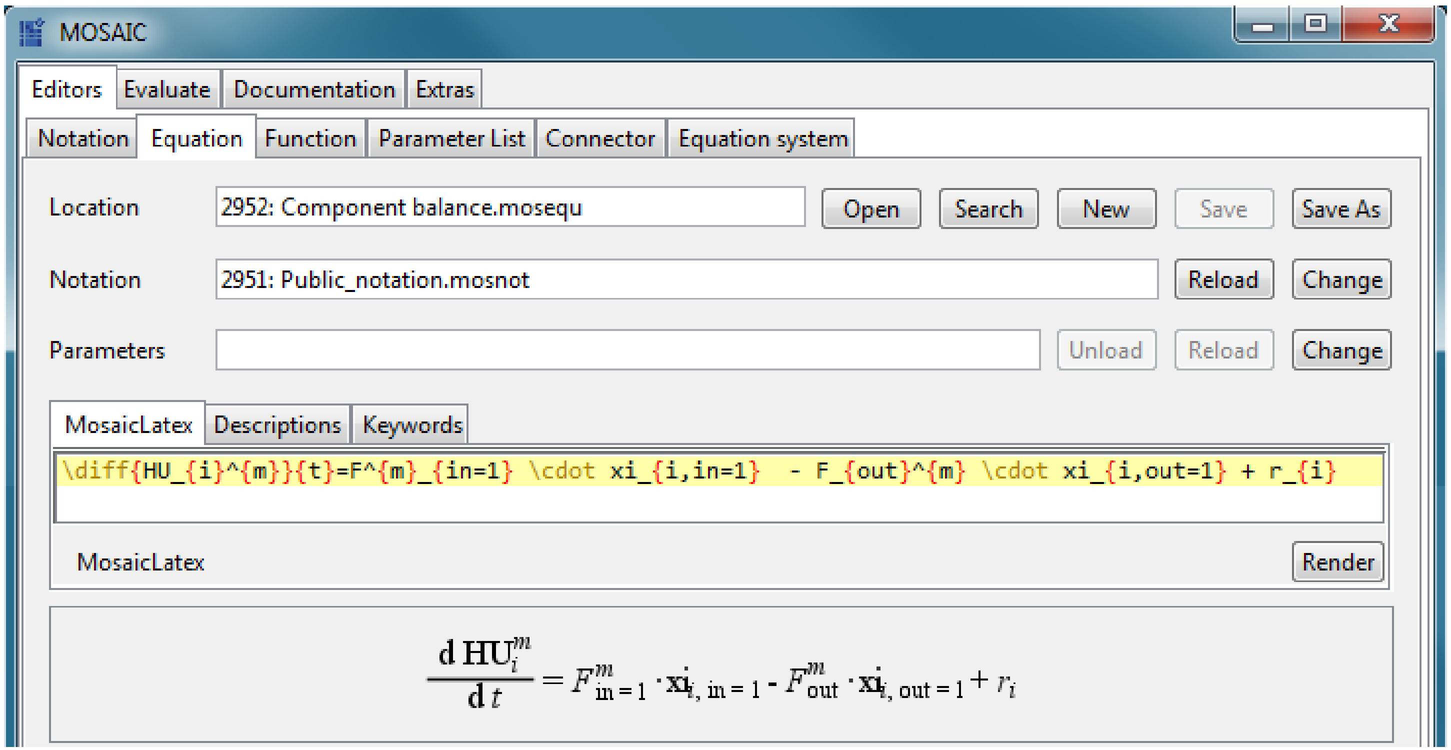
Notation editor:



Example: $p^{LV,I}_{o,i=2,j=4}$

Enhanced Symbolic Notation II - Equations

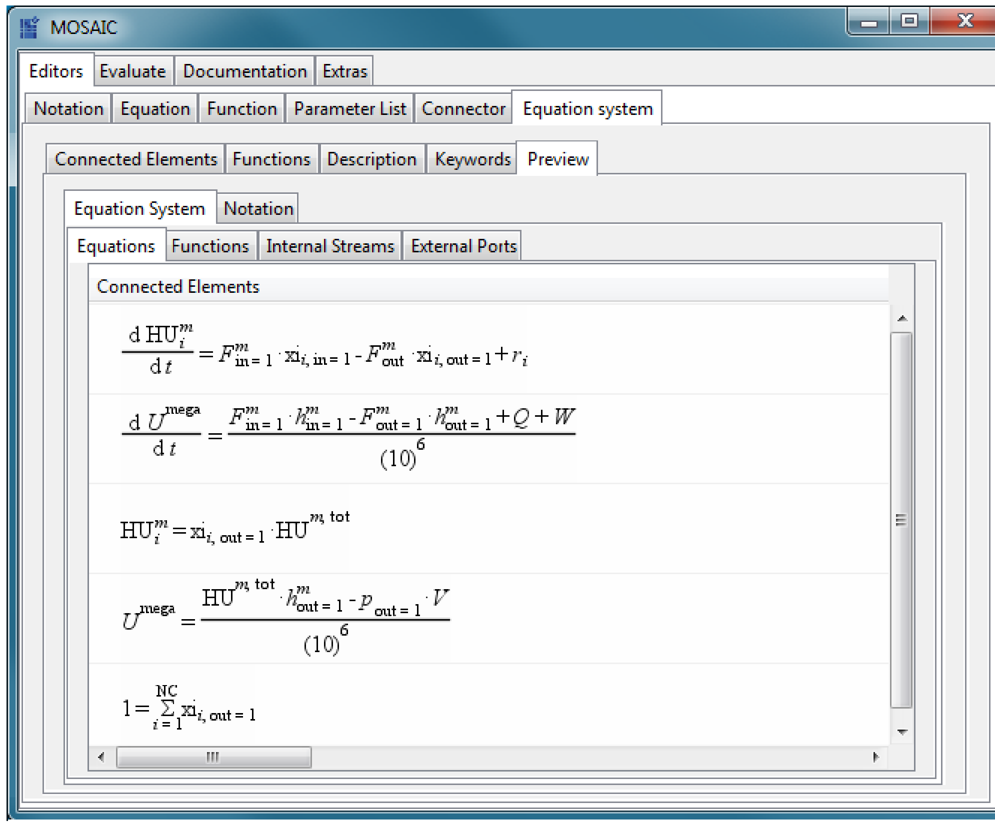
Equation editor:



d|b|t|a
www.dbta.tu-berlin.de

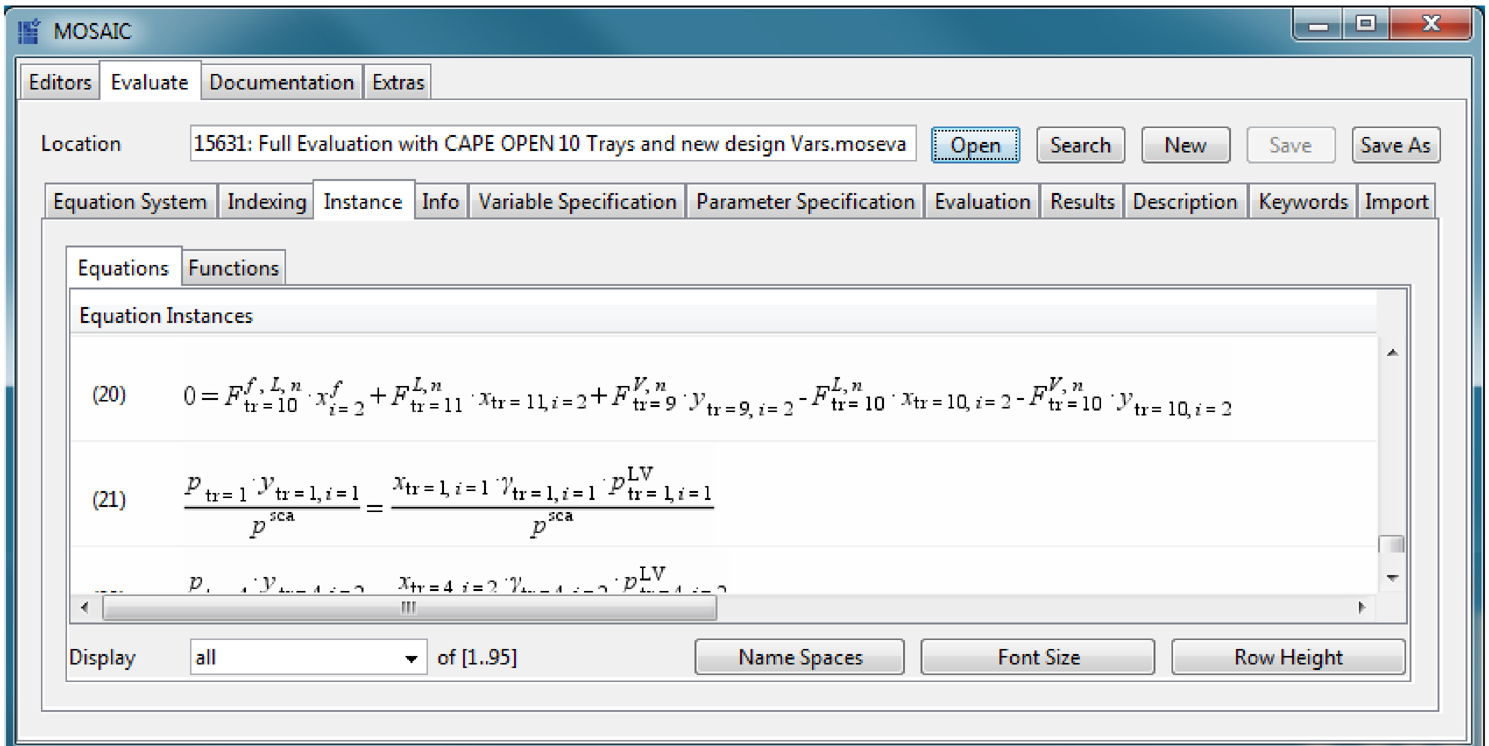
Enhanced Symbolic Notation III – Equation Systems

Equation system editor:



Enhanced Symbolic Notation IV – Instantiated Equations

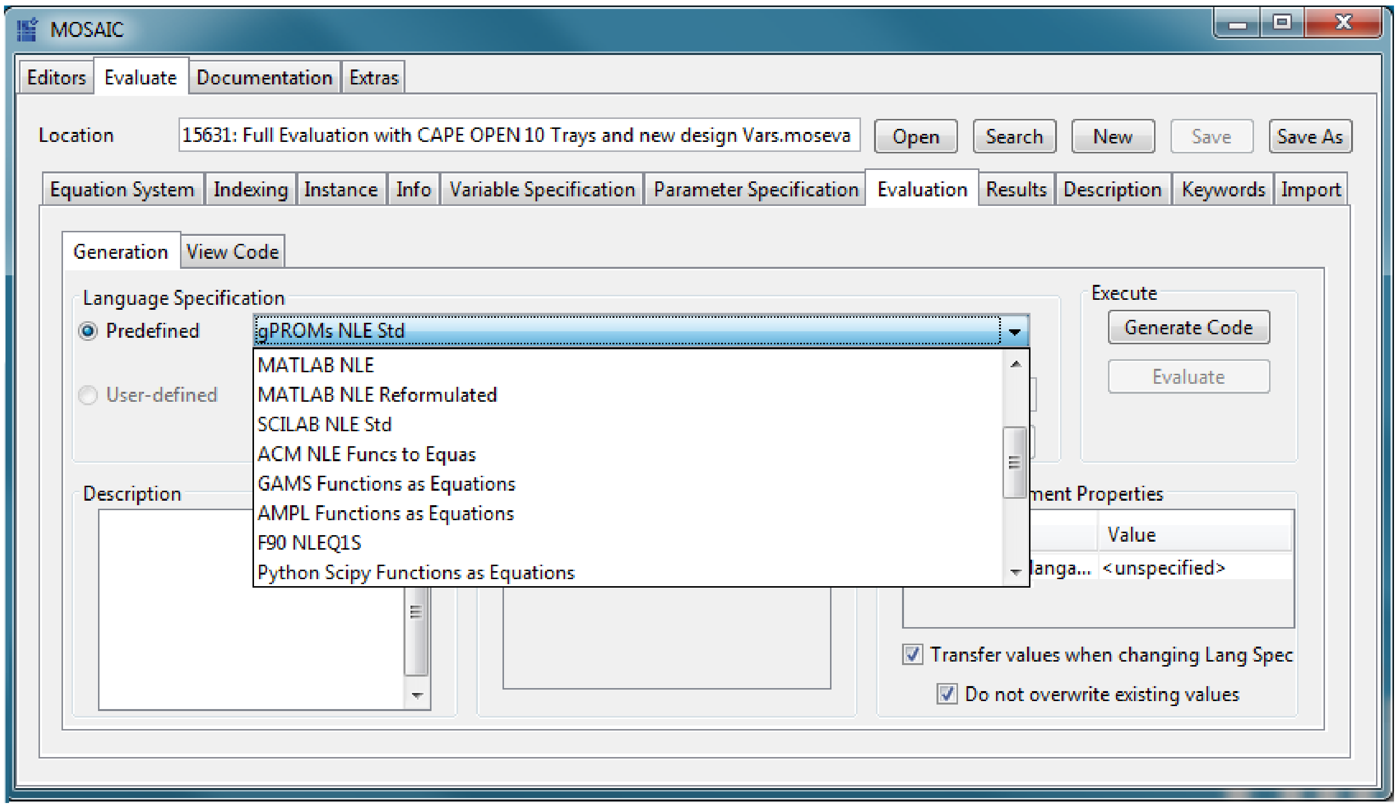
Evaluation editor:



d|b|t|a
www.dbta.tu-berlin.de

Code Generation I – What language do you prefer?

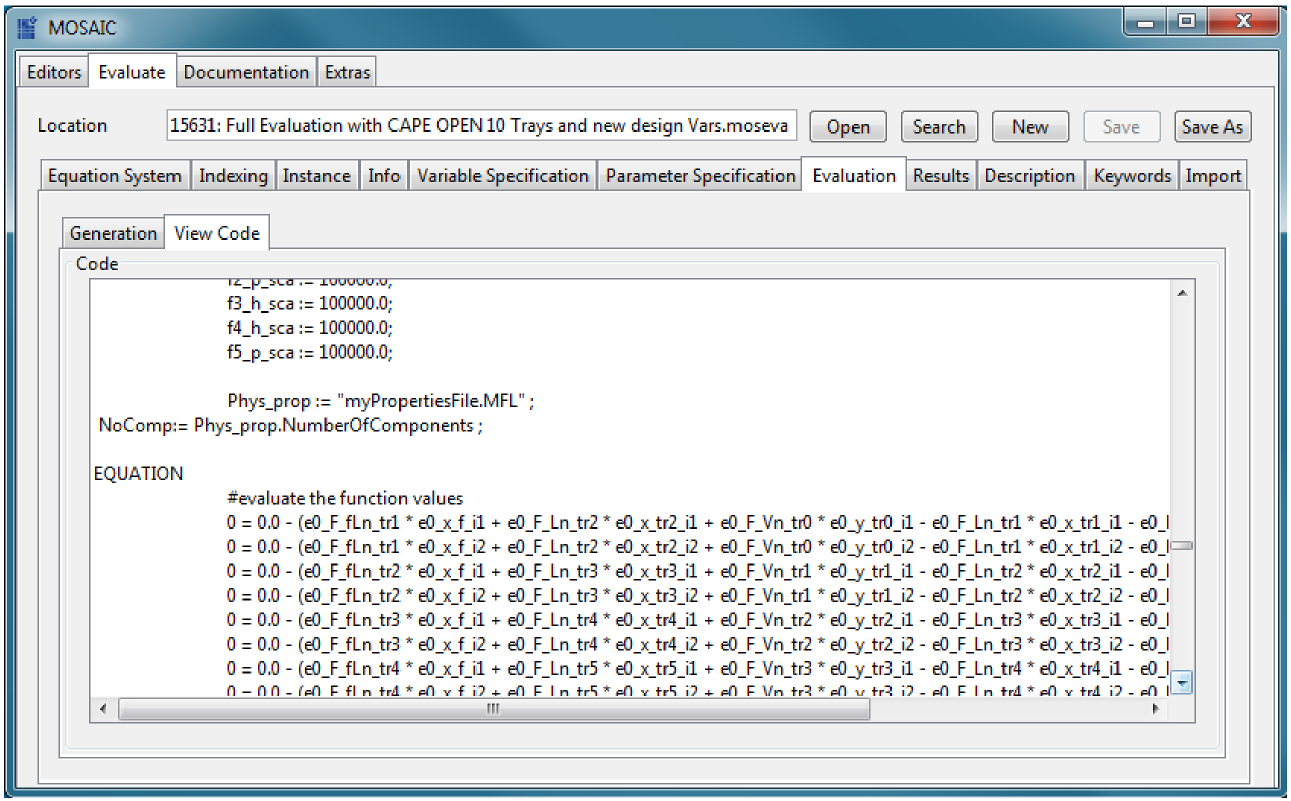
Choose of a list of supported target platforms:



d|b|t|a
www.dbta.tu-berlin.de

Code Generation II – Show me what you got!

Take a look at the generated code:

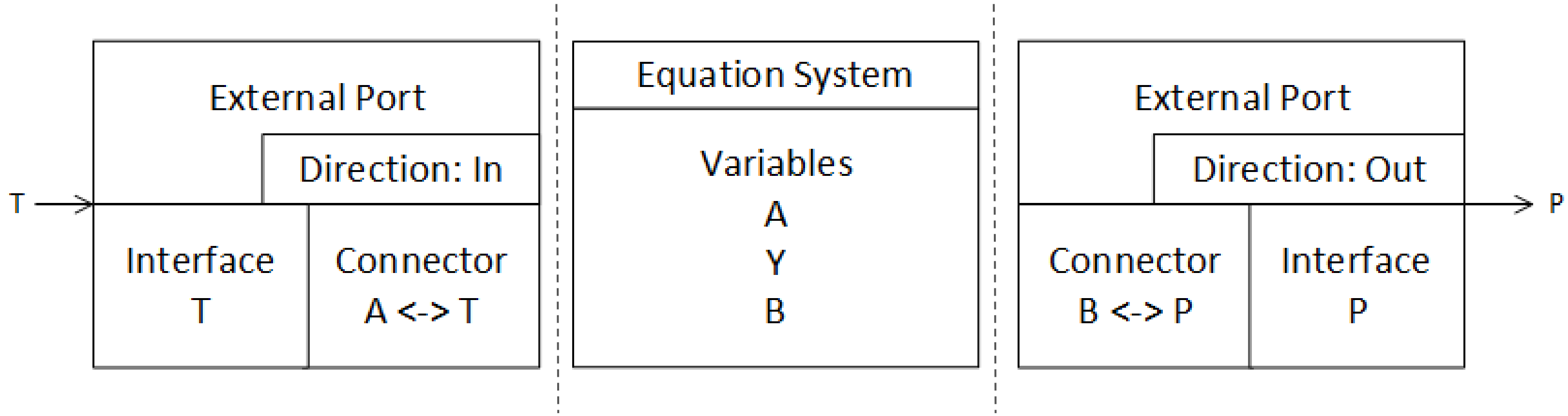


Technische Universität Berlin

d|b|t|a
www.dbta.tu-berlin.de

MOSAIC Ports – let's get connected

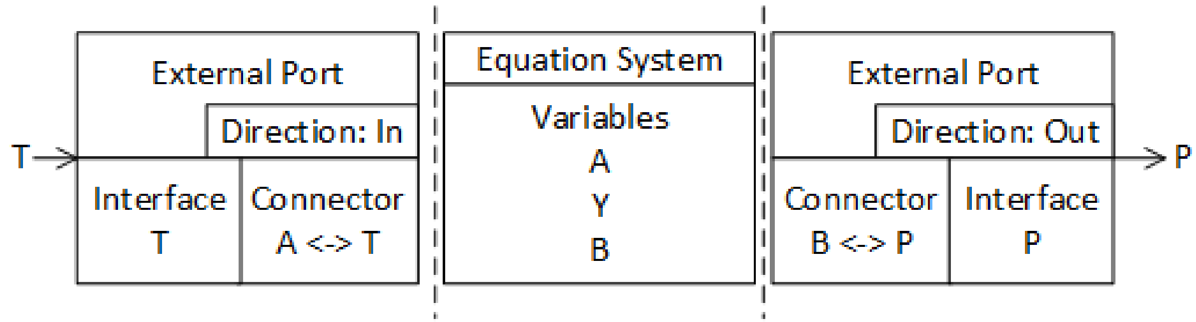
MOSAIC external Ports:



Y – internal variable
A – variable connected to input T
B – variable connected to output P

MOSAIC Ports – let‘s get connected

MOSAIC external Ports:



- Direction:
  In or Out

- Interface:
  Which variables will be presented?
  -> naming, dimension, engineering unit, direction
  e.g. p, scalar, bar, out

- Connector:
  How are internal variables and external interface variables connected?
  -> e.g. A <-> T , B <-> P

d|b|t|a
www.dbta.tu-berlin.de

CAPE-OPEN and MOSAIC I – Physical Properties

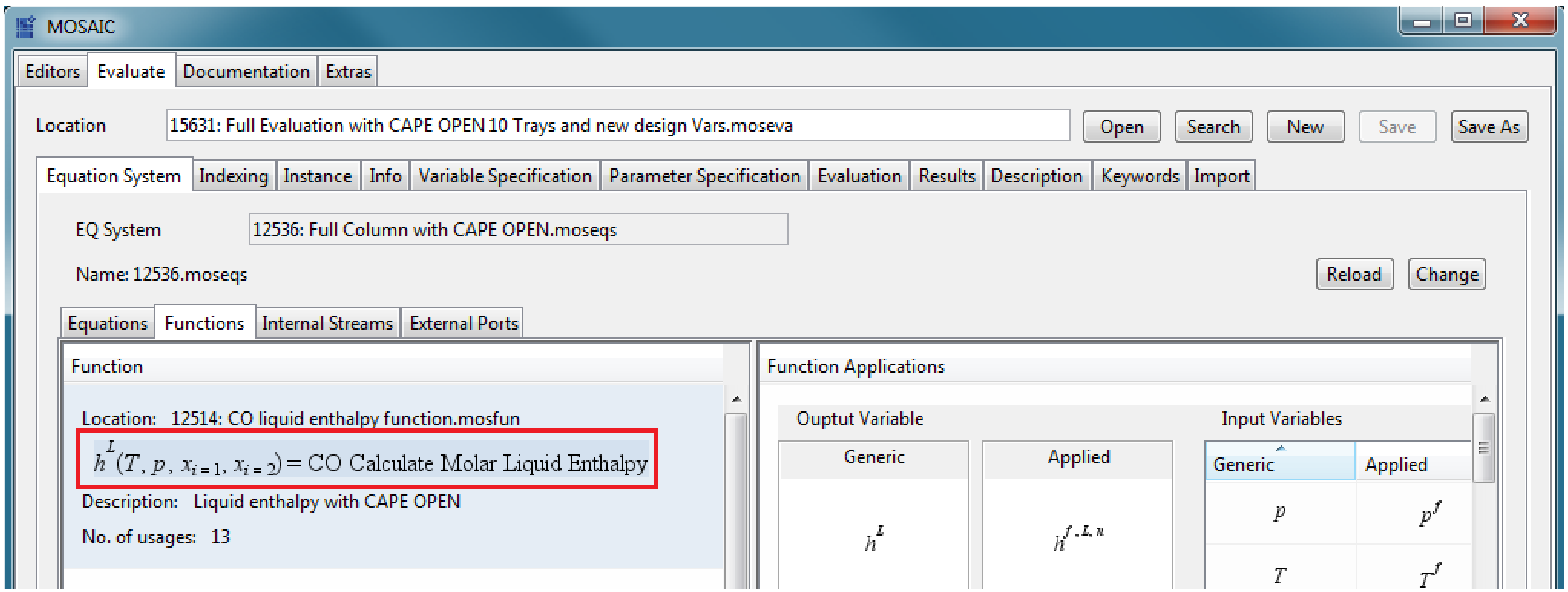
Physical properties in MOSAIC:
Variables to be calculated by external functions, e.g. $p_i^{\mathrm{LV}}(T)$

Supported target platforms for „CO physical properties“ code generation:

- Matlab
- gPROMS

CAPE-OPEN and MOSAIC I – Physical Properties

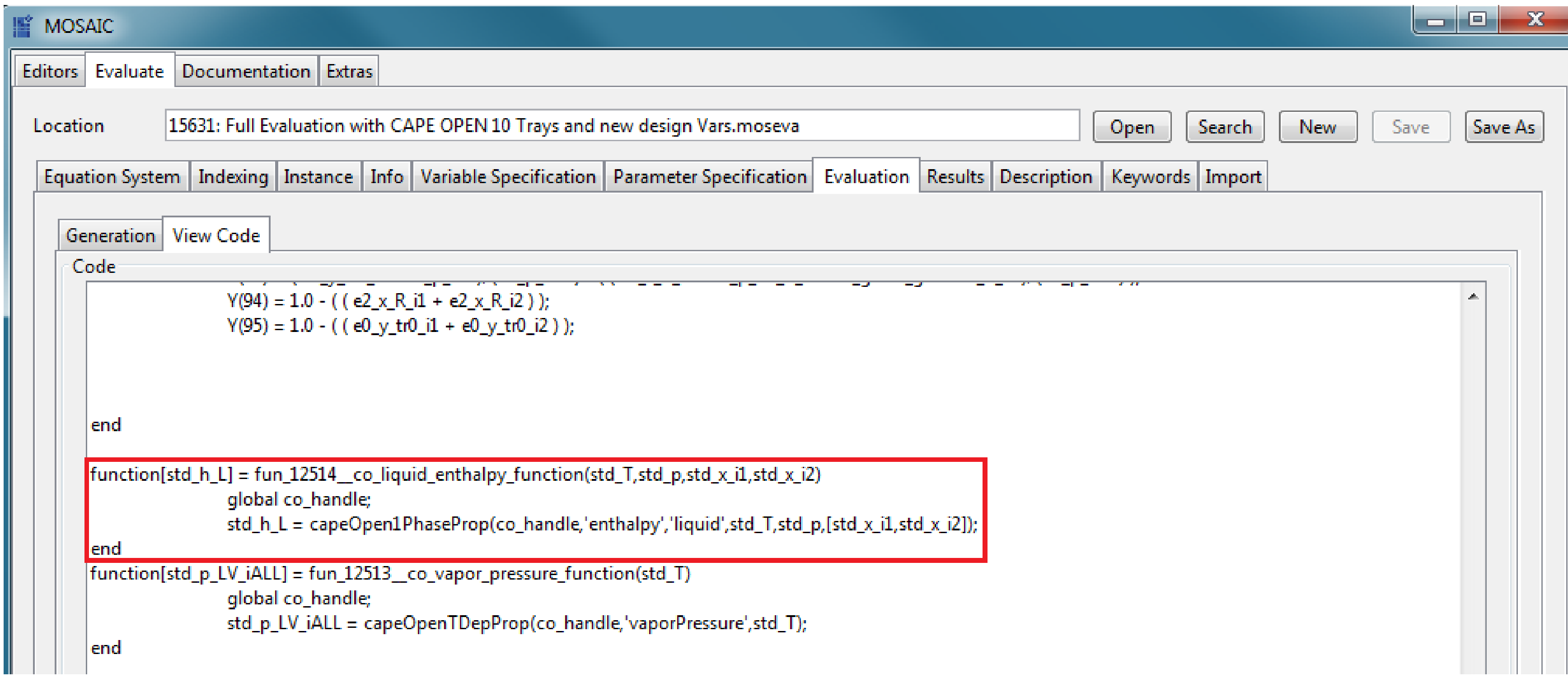
MOSAIC Example – CO function:



d|b|t|a
www.dbta.tu-berlin.de

CAPE-OPEN and MOSAIC I – Physical Properties

MOSAIC Example – Matlab code:



d|b|t|a
www.dbta.tu-berlin.de

Summary

MOSAIC

- A **modular** equation based modeling tool
- Implemented in **Java**, using **XML/MathML**
- Provides **automatic code generation** for specific platforms (e.g. Matlab, C++)
- Can use the concept of **ports**
- Supports **CO physical properties** in code generation (Matlab, gPROMS)

MOSAIC is not

- Designed to be a full solver / process simulator
- A programming language
- A computer algebra system (CAS)

Technische Universität Berlin

Thank you very much for your kind attention.

MOSAIC

www.mosaic-modeling.de

Acknowledgement:

This project is

- supported by the Cluster of Excellence 'Unifying Concepts in Catalysis'
- coordinated by the Technical University of Berlin and
- funded by the German Research Foundation.