



CiT

Dr. M. Wulkow
Computing in Technology

Presto-Kinetics 11 – Overview

Dr. Michael Wulkow

- ▶ Founded in 1992, M. Wulkow is a mathematician
- ▶ Software for modeling and simulation in chemistry, mainly
 - ▶ PREDICI® (polymer kinetics)
 - ▶ PARSIVAL® (crystallization, particle systems)
 - ▶ PRESTO-KINETICS® (chemical and bio kinetics, spatial distributions)
- ▶ Clients
 - ▶ Many companies and universities world-wide

- ▶ Basic chemical kinetics
- ▶ Bio kinetics and systems biology
- ▶ Reactor models → batch, semi-batch, continuous, plug-flow, cascades
- ▶ Modules for
 - ▶ kinetics
 - ▶ phase changes
 - ▶ mass transfers
 - ▶ particle growth
 - ▶ reactor flows
 - ▶ spatial profiles
- ▶ Solver for partial differential equations
- ▶ Framework for multiple-phase systems
- ▶ Sophisticated parameter estimation based on unique algorithm

Further important modules

- ▶ **PDE solver** for treatment of spatial profiles in catalysts or tube reactors
- ▶ **Sensitivity analysis** for parameters
- ▶ **Recipe variation**
- ▶ **Optimal control** module for stream and temperature control
- ▶ **Cape-Open** interface for access of thermodynamic property packages
- ▶ User **database** for handling of substance and kinetic data

Presto-Kinetics 11 by CiT

File Project Simulation Tools Navigation

Quick simulation Control Additional Structural windows Additional info PE options

Debug mode Create Replay Show Start 1_Step Stop Restart End time 3600 Use condition Replay Close Workshop MC results Reaction system Variation results Simulation Deriv results List of open windows Show mass balance Show scale in graphs Actualize graphs during PE

Graphs (Model: Oregonator.xml, Recipe: Rec2)

Conc [mol/l] vs Time [s]

Model/Simulation PE + Optimization + Sensitivity analysis

Components Library Settings & Numerics Script Report PSD data

Index	Name	Active	Group	Educt[1]	Educt[2]	Act	Product[1]	Product[2]	Product[3]	Coefficient
1	Kinetic	yes	Br	+ BrO3	<->		HB(O)2	+ HOBr		k1
2	Kinetic	yes	Br	+ HB(O)2	<->		2H(O)Br			k2
3	Kinetic	yes	BrO3				2Ce	+ 2HB(O)2		k3
4	Kinetic	yes	HB(O)2^2				BrO3	+ HOBr		k4
5	Kinetic	yes	Ce				Br			k5

Reaction step of Oregonator.xml

Name: General kinetic step for elemental species
 Pattern: $n1A1 + \dots + n5A5 \leftrightarrow m1B1 + \dots + m5B5$
 Result: $Br + HB(O)2 \leftrightarrow 2H(O)Br, k2, 0$

No.	n _{i,j}	A _i	order _i	m _j	B _j	order _j
1		Br		2	H(O)Br	
2		HB(O)2	k2			
3			<->			
4			0			
5			k _{back} (script)			

Script (+Enthapy):
 Reaction law ip:
 Equilibrium

The stoichiometric factors are wrong
 Sum of mol weights:
 left side: 1.530e-01
 right side: 1.560e-01
 difference: 1.00000e-03

Messages

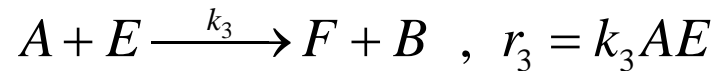
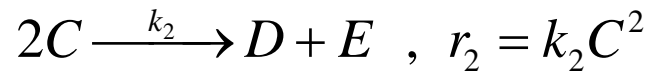
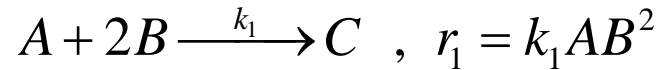
Time [s] Exit [kg]

Messages Integration data Msg + Error+Warnings

36000e+03 36000e+03 00:00:02

Close

- ▶ Typical scheme



- ▶ Differential equations (ODEs)

$$A' = -r_1 - r_3 = -k_1 AB^2 - k_3 AE$$

$$B' = -2r_1 + r_3 = -2k_1 AB^2 + k_3 AE$$

$$C' = r_1 - 2r_2 = k_1 AB^2 - 2k_2 C^2$$

$$D' = r_2 = k_2 C^2$$

$$E' = r_2 - r_3 = k_2 C^2 - k_3 AE$$

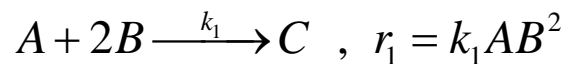
$$F' = r_3 = k_3 AE$$

- ▶ Often formally written in terms of matrix-vector product

$$\begin{bmatrix} A' \\ B' \\ C' \\ D' \\ E' \\ F' \end{bmatrix} = \begin{bmatrix} -1 & 0 & -1 \\ -2 & 0 & 1 \\ 1 & -2 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} k_1 AB^2 \\ k_2 C^2 \\ k_3 AE \end{bmatrix} = S \cdot r$$

- ▶ Implementation in plain code requires proper numbering
- ▶ In large systems single reactions are difficult to see

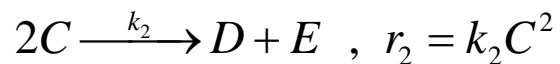
- ▶ Single reactions derive their “own” ODEs internally
- ▶ All terms are superposed, even kinetic steps and abstract ODEs
- ▶ All further terms are added automatically
- ▶ Once a kinetic step pattern is implemented and validated, it can be used again and again.
- ▶ Reaction rates can be arbitrarily complex
- ▶ Equilibrium reactions supported (requires extension of S and r otherwise)



$$A' = -k_1 r AB^2$$

$$B' = -2k_1 AB^2$$

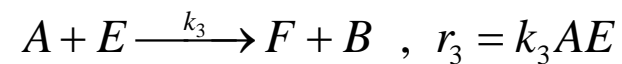
$$C' = k_1 AB^2$$



$$C' = -2k_2 C^2$$

$$D' = k_2 C^2$$

$$E' = k_2 C^2$$



$$A' = -k_3 AE$$

$$E' = -k_3 AE$$

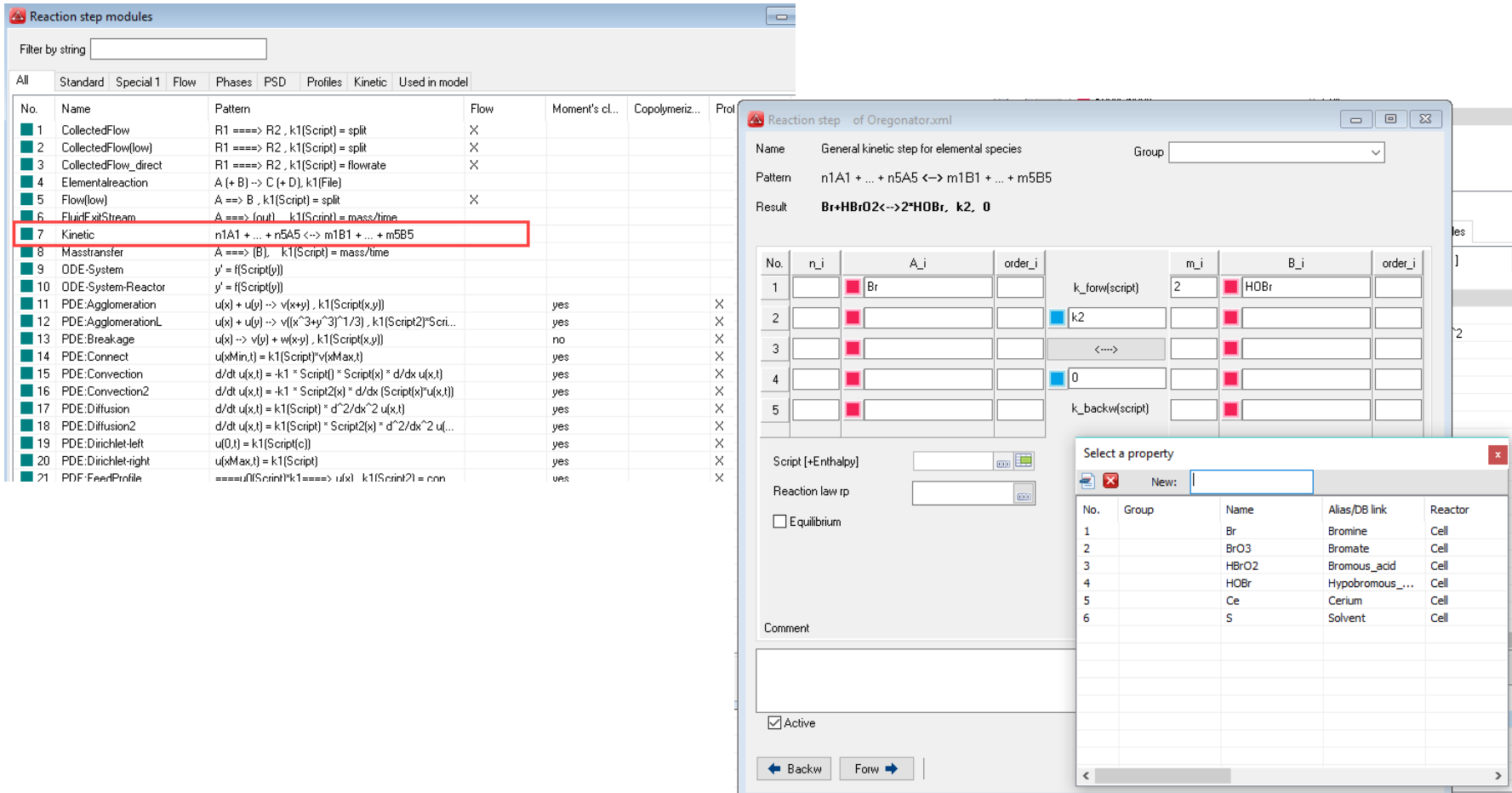
$$B' = k_3 AE$$

$$F' = k_3 AE$$

- ▶ Internal treatment of species based on mol/volume
- ▶ Volume treatment important in many applications
- ▶ Recipe input allows all kind of inputs that is transformed to basic units
- ▶ Reaction rates have to be returned in terms of mol/volume/time
- ▶ Volume influenced by streams and density changes

$$\frac{dA}{dt} = r_A^{total} - \frac{V_R'}{V} A - \frac{V_\rho'}{V} A$$

- ▶ Select reaction step pattern from list using, e.g. “General kinetic”
- ▶ Assign species w.r.t modeling context



The image shows a software interface for configuring reaction step modules. On the left, a table lists various modules, with the 'Kinetic' module (No. 7) highlighted in red. On the right, a configuration window for the 'General kinetic step for elemental species' is shown, with a 'Result' field containing the chemical equation $\text{Br} + \text{HBrO}_2 \leftrightarrow 2\text{HBr}$ and $k_2 = 0$. Below this, a table maps species to reaction orders, and a 'Select a property' dialog is open, showing a list of species and their associated reactors.

No.	Name	Pattern	Flow	Moment's cl..	Copolymeriz...	Pro
1	CollectedFlow	R1 ==> R2 , k1[Script] = split	X			
2	CollectedFlow(low)	R1 ==> R2 , k1[Script] = split	X			
3	CollectedFlow_direct	R1 ==> R2 , k1[Script] = flowrate	X			
4	Elementalreaction	A (+ B) -> C (+ D), k1[File]				
5	Flow(low)	A ==> B , k1[Script] = split	X			
6	FluidExitStream	A ==> [out] ... k1[Script] = mass/time				
7	Kinetic	n1A1 + ... + n5A5 <-> m1B1 + ... + m5B5				
8	Masstransfer	A ==> [B] , k1[Script] = mass/time				
9	ODE-System	y' = f[Script(y)]				
10	ODE-System-Reactor	y' = f[Script(y)]				
11	PDE:Agglomeration	u(x) + u(y) -> v(x+y) , k1[Script(x,y)]		yes		X
12	PDE:AgglomerationL	u(x) + u(y) -> v((x^3+y^3)^(1/3)) , k1[Script2]*Sci...		yes		X
13	PDE:Breakage	u(x) -> v(y) + w(x-y) , k1[Script(x,y)]		no		X
14	PDE:Connect	u(xMin,t) = k1[Script]*v(xMax,t)		yes		X
15	PDE:Convection	d/dt u(x,t) = -k1 * Script(x) * Script(x) * d/dx u(x,t)		yes		X
16	PDE:Convection2	d/dt u(x,t) = -k1 * Script2(x) * d/dx (Script(x)*u(x,t))		yes		X
17	PDE:Diffusion	d/dt u(x,t) = k1[Script] * d^2/dx^2 u(x,t)		yes		X
18	PDE:Diffusion2	d/dt u(x,t) = k1[Script] * Script2(x) * d^2/dx^2 u(...		yes		X
19	PDE:Dirichlet-left	u(0,t) = k1[Script(c)]		yes		X
20	PDE:Dirichlet-right	u(xMax,t) = k1[Script]		yes		X
21	PDF:FeedProfile	====u[Script]*k1====> u(x) k1[Script2] = nnn		yes		X

Reaction step configuration details:

- Name: General kinetic step for elemental species
- Pattern: $n1A1 + \dots + n5A5 \leftrightarrow m1B1 + \dots + m5B5$
- Result: $\text{Br} + \text{HBrO}_2 \leftrightarrow 2\text{HBr}$, $k_2 = 0$

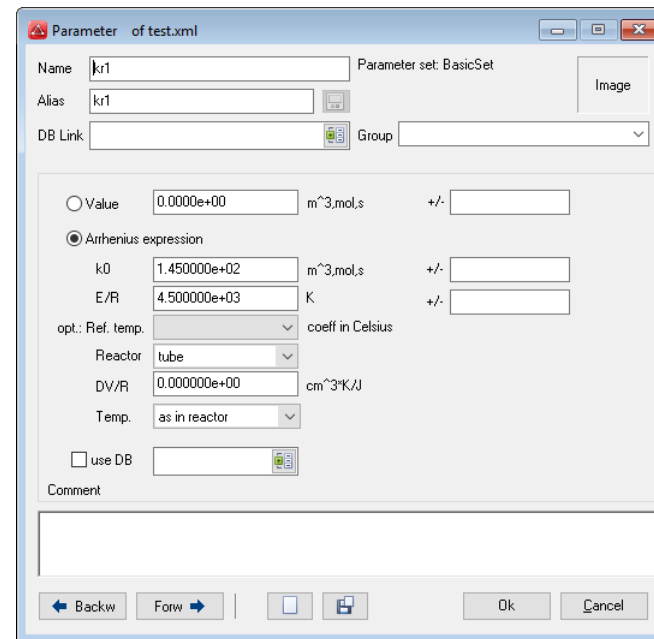
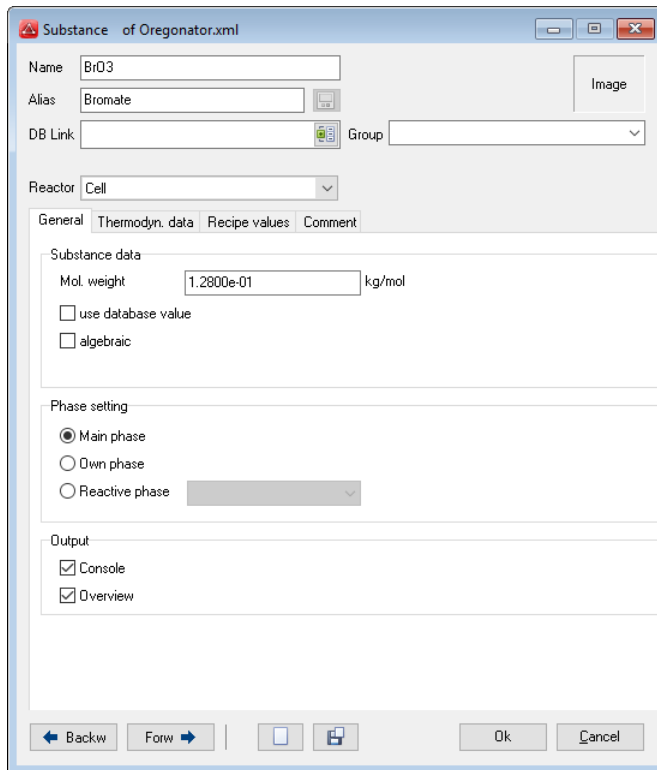
No.	n_i	A_i	order_i	k_for(script)	m_i	B_i	order_i
1		Br			2	HBr	
2				k2			
3				<---->			
4				0			
5				k_backw(script)			

Select a property dialog:

No.	Group	Name	Alias/DB link	Reactor
1		Br	Bromine	Cell
2		BrO3	Bromate	Cell
3		HBrO2	Bromous_acid	Cell
4		HOB	Hypobromous_...	Cell
5		Ce	Cerium	Cell
6		S	Solvent	Cell

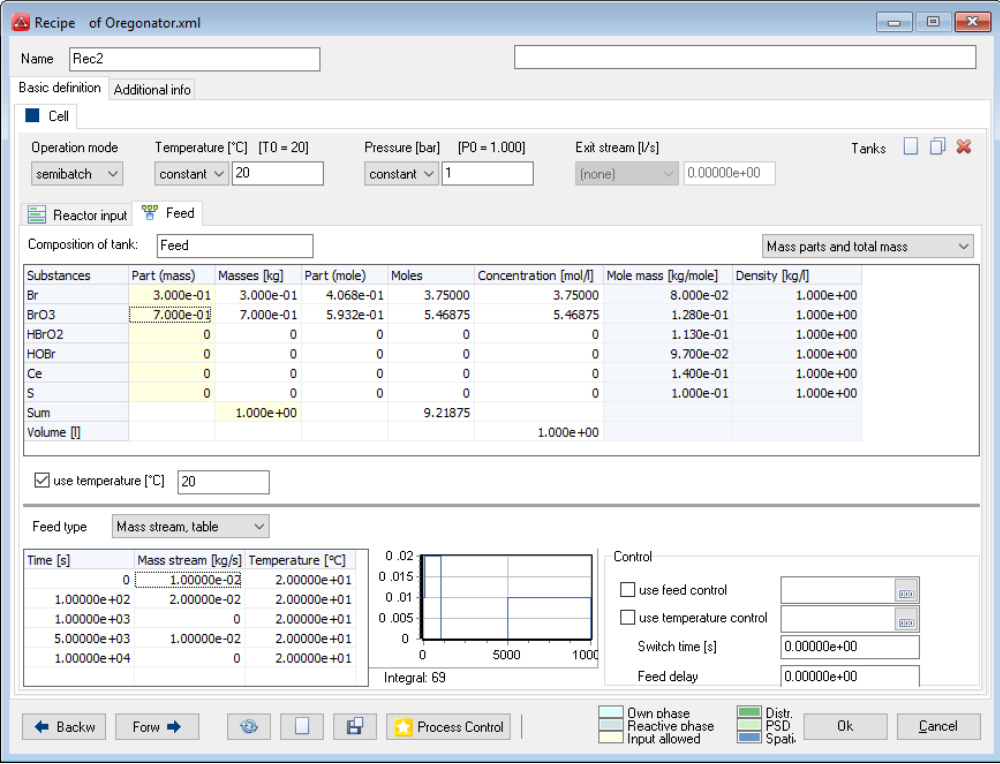
Model administration – components

- ▶ Various properties and settings
- ▶ Comfortable editing
- ▶ Access to user-database
- ▶ Extension by user-scripts possible



Recipes

- ▶ All input of all species entered in terms of recipes
- ▶ Various input types
- ▶ Multi-reactor treatment
- ▶ Temperature and pressure control
- ▶ Project contains list of recipes, one active
- ▶ Various feed and control strategies



Recipe of Oregonator.xml

Name: Rec2

Basic definition: Additional info

Cell

Operation mode: semibatch

Temperature [°C]: [T0 = 20] constant 20

Pressure [bar]: [P0 = 1.000] constant 1

Exit stream [l/s]: (none) 0.00000e+00

Tanks: [] [] [X]

Reactor input: Feed

Composition of tank: Feed

Substances	Part (mass)	Masses [kg]	Part (mole)	Moles	Concentration [mol/l]	Mole mass [kg/mole]	Density [kg/l]
Br	3.000e-01	3.000e-01	4.068e-01	3.75000	3.75000	8.000e-02	1.000e+00
BrO3	7.000e-03	7.000e-01	5.932e-01	5.46875	5.46875	1.280e-01	1.000e+00
HBrO2	0	0	0	0	0	1.130e-01	1.000e+00
HOBr	0	0	0	0	0	9.700e-02	1.000e+00
Ce	0	0	0	0	0	1.400e-01	1.000e+00
S	0	0	0	0	0	1.000e-01	1.000e+00
Sum		1.000e+00		9.21875			
Volume [l]					1.000e+00		

use temperature [°C] 20

Feed type: Mass stream, table

Time [s]	Mass stream [kg/s]	Temperature [°C]
0	1.00000e-02	2.00000e+01
1.00000e+02	2.00000e-02	2.00000e+01
1.00000e+03	0	2.00000e+01
5.00000e+03	1.00000e-02	2.00000e+01
1.00000e+04	0	2.00000e+01

Integral: 69

Control:

use feed control

use temperature control

Switch time [s]: 0.00000e+00

Feed delay: 0.00000e+00

Own phase: Dist. PSD:

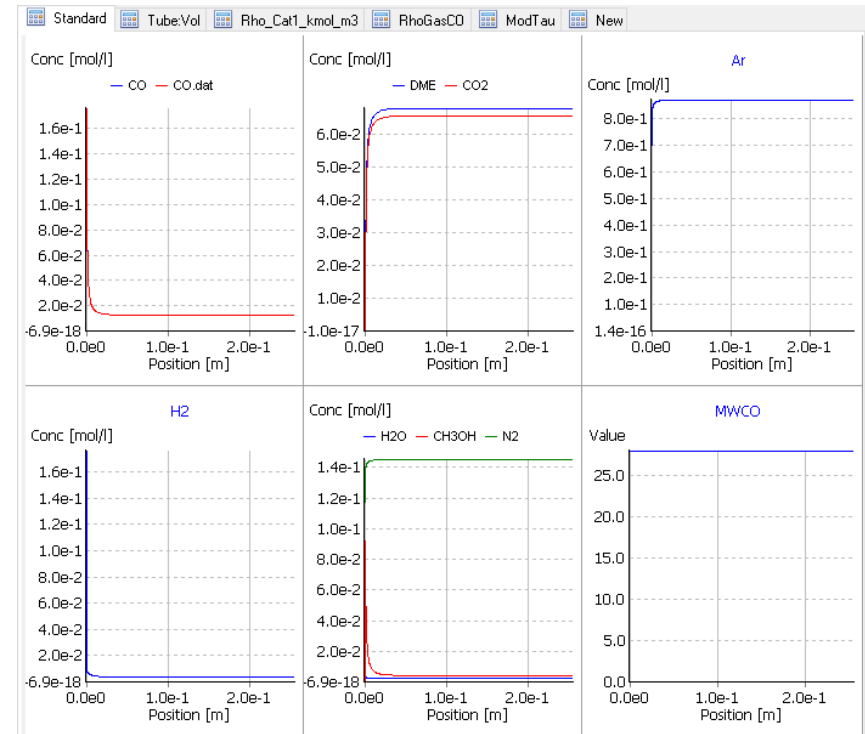
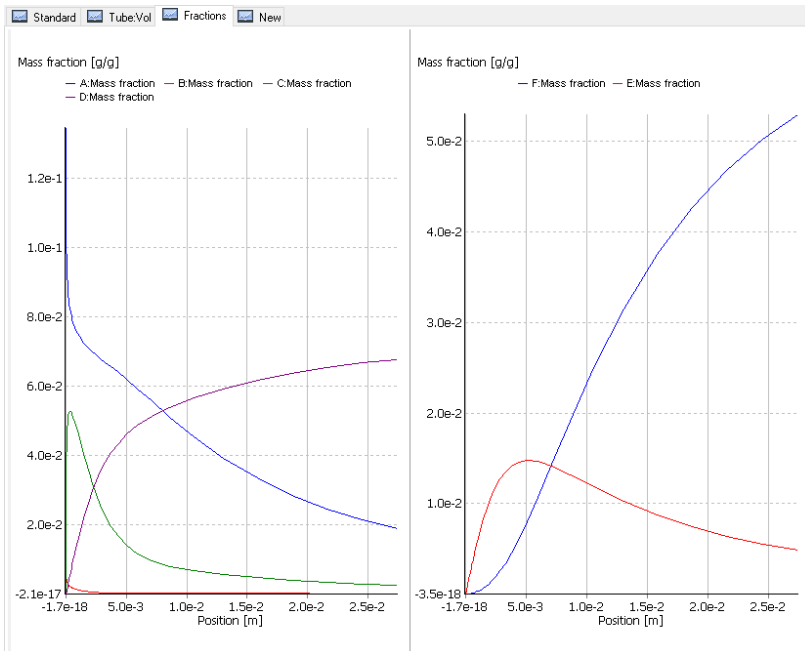
Reactive phase: Spati.:

Input allowed:

Ok Cancel

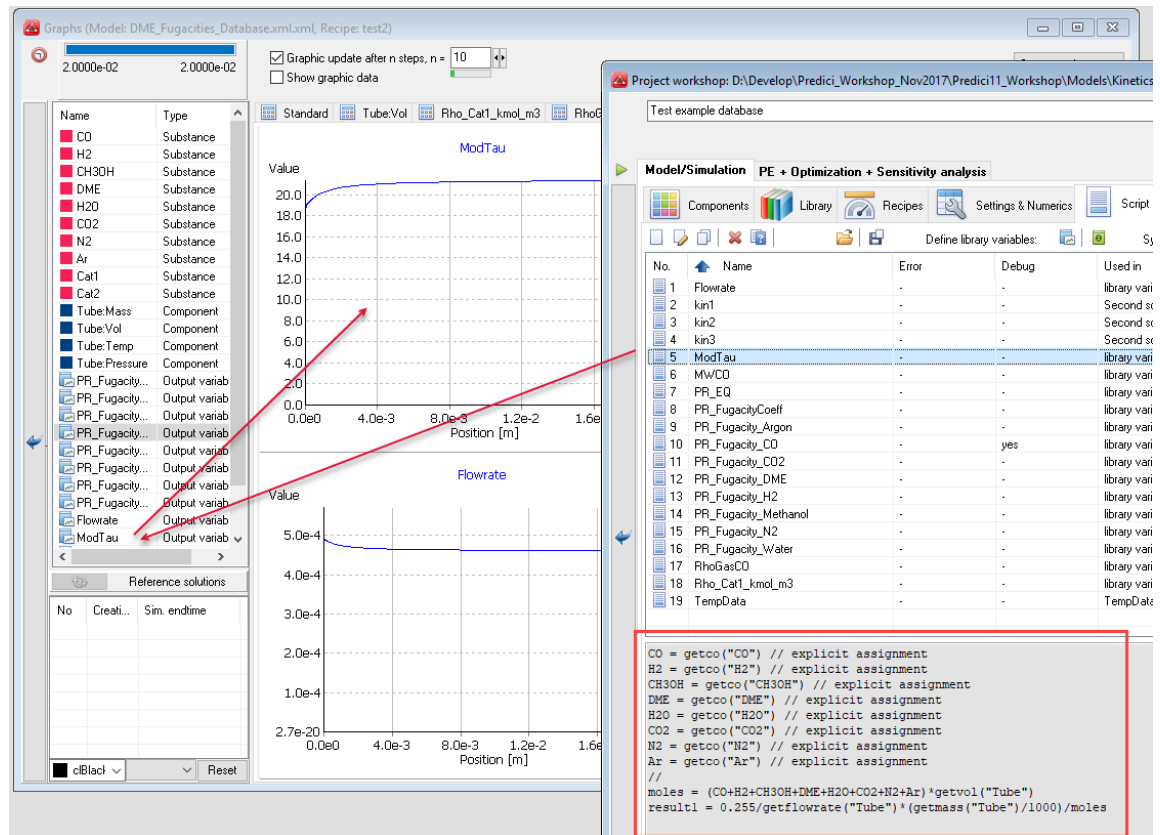
Powerful online chart administration

- ▶ Create own graphic tabs, even during a simulation
- ▶ Drag & drop curves to graphic tab
- ▶ Combine graphics in one chart
- ▶ Loaded reference data are stored within the project



Scripts

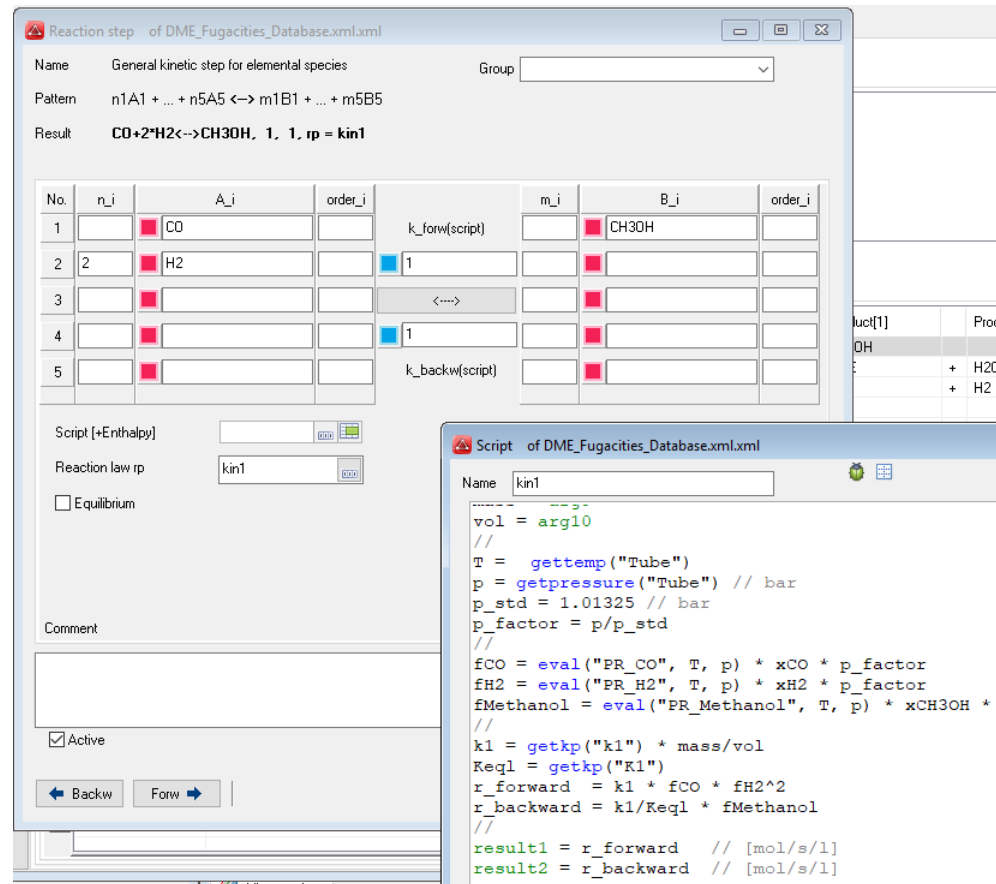
- ▶ Direct output of user-defined expressions of any kind
- ▶ User-defined equations and rates
- ▶ Access to all system variables
- ▶ Export of results



The screenshot displays a software interface for a simulation. The main window shows two plots: 'ModTau' and 'Flowrate'. The 'ModTau' plot shows a value increasing from 0 to approximately 20.0 over a position range of 0.0e0 to 1.6e-2. The 'Flowrate' plot shows a value decreasing from approximately 5.0e-4 to 2.7e-20 over the same position range. A table on the right lists simulation variables, including 'ModTau' which is highlighted. Below the table, a script editor shows the following code:

```
CO = getco("CO") // explicit assignment
H2 = getco("H2") // explicit assignment
CH3OH = getco("CH3OH") // explicit assignment
DME = getco("DME") // explicit assignment
H2O = getco("H2O") // explicit assignment
CO2 = getco("CO2") // explicit assignment
N2 = getco("N2") // explicit assignment
Ar = getco("Ar") // explicit assignment
//
moles = (CO+H2+CH3OH+DME+H2O+CO2+N2+Ar)*getvol("Tube")
result1 = 0.255/getflowrate("Tube")*(getmass("Tube")/1000)/moles
```

- ▶ Reaction rates
 - ▶ Presto-Kinetics combines stoichiometry, rate parameters and reaction law
 - ▶ Scripts may re-define any part of a kinetics step



The screenshot displays two windows from the Presto-Kinetics software. The top window, titled "Reaction step of DME_Fugacities_Database.xml.xml", shows the configuration for a reaction step. The name is "General kinetic step for elemental species" and the pattern is $n_1A_1 + \dots + n_5A_5 \leftrightarrow m_1B_1 + \dots + m_5B_5$. The result is $CO+2\cdot H_2 \leftrightarrow CH_3OH, 1, 1, rp = kin1$.

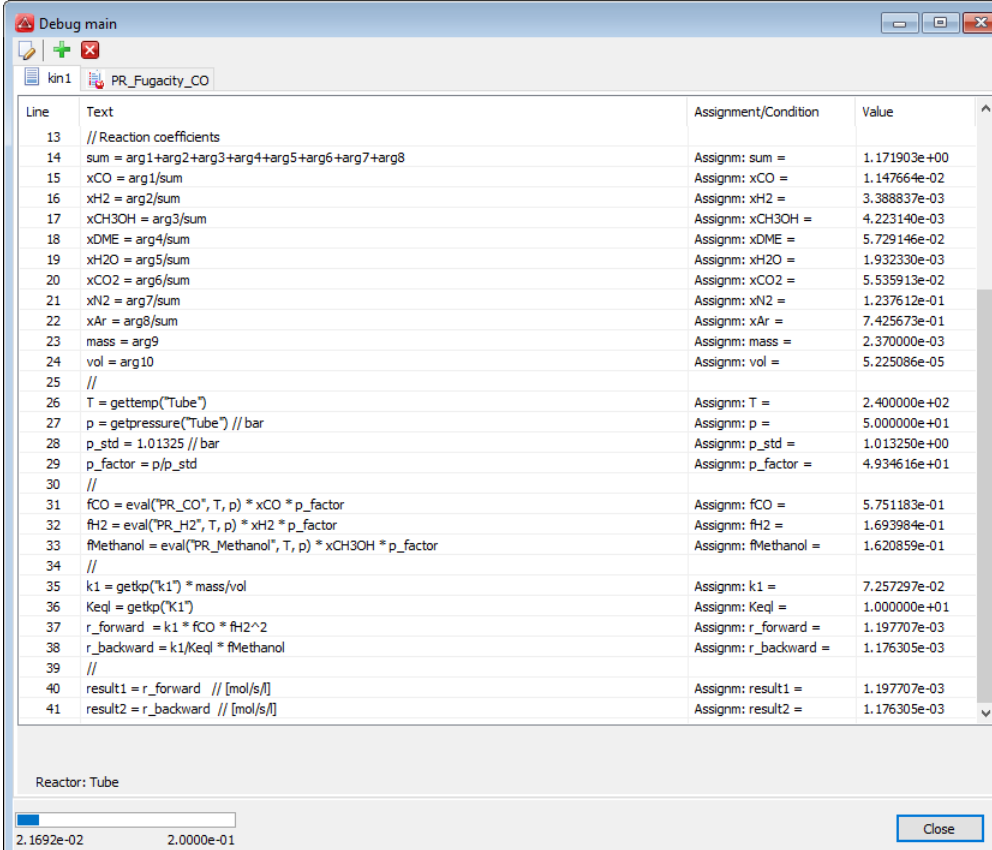
No.	n _i	A _i	order _i	m _i	B _i	order _i
1		CO			CH ₃ OH	
2	2	H ₂		1		
3						
4				1		
5						

The bottom window, titled "Script of DME_Fugacities_Database.xml.xml", shows a script for the reaction step named "kin1". The script defines variables for volume, temperature, pressure, and fugacity coefficients, and calculates the forward and backward reaction rates.

```

Name kin1
-----
vol = arg10
//
T = gettemp("Tube")
p = getpressure("Tube") // bar
p_std = 1.01325 // bar
p_factor = p/p_std
//
fCO = eval("PR_CO", T, p) * xCO * p_factor
fH2 = eval("PR_H2", T, p) * xH2 * p_factor
fMethanol = eval("PR_Methanol", T, p) * xCH3OH *
//
k1 = getkp("k1") * mass/vol
Reql = getkp("K1")
r_forward = k1 * fCO * fH2^2
r_backward = k1/Reql * fMethanol
//
result1 = r_forward // [mol/s/l]
result2 = r_backward // [mol/s/l]
    
```

- ▶ Script debugger
 - ▶ line-by-line check of code
 - ▶ check at any time during simulation possible (step mode)



Line	Text	Assignment/Condition	Value
13	// Reaction coefficients		
14	sum = arg1+arg2+arg3+arg4+arg5+arg6+arg7+arg8	Assignm: sum =	1.171903e+00
15	xCO = arg1/sum	Assignm: xCO =	1.147664e-02
16	xH2 = arg2/sum	Assignm: xH2 =	3.388837e-03
17	xCH3OH = arg3/sum	Assignm: xCH3OH =	4.223140e-03
18	xDME = arg4/sum	Assignm: xDME =	5.729146e-02
19	xH2O = arg5/sum	Assignm: xH2O =	1.932330e-03
20	xCO2 = arg6/sum	Assignm: xCO2 =	5.535913e-02
21	xN2 = arg7/sum	Assignm: xN2 =	1.237612e-01
22	xAr = arg8/sum	Assignm: xAr =	7.425673e-01
23	mass = arg9	Assignm: mass =	2.370000e-03
24	vol = arg10	Assignm: vol =	5.225086e-05
25	//		
26	T = gettemp("Tube")	Assignm: T =	2.400000e+02
27	p = getpressure("Tube") // bar	Assignm: p =	5.000000e+01
28	p_std = 1.01325 // bar	Assignm: p_std =	1.013250e+00
29	p_factor = p/p_std	Assignm: p_factor =	4.934616e+01
30	//		
31	fCO = eval("PR_CO", T, p) * xCO * p_factor	Assignm: fCO =	5.751183e-01
32	fH2 = eval("PR_H2", T, p) * xH2 * p_factor	Assignm: fH2 =	1.693984e-01
33	fMethanol = eval("PR_Methanol", T, p) * xCH3OH * p_factor	Assignm: fMethanol =	1.620859e-01
34	//		
35	k1 = getkp("k1") * mass/vol	Assignm: k1 =	7.257297e-02
36	Keq1 = getkp("K1")	Assignm: Keq1 =	1.000000e+01
37	r_forward = k1 * fCO * fH2^2	Assignm: r_forward =	1.197707e-03
38	r_backward = k1/Keq1 * fMethanol	Assignm: r_backward =	1.176305e-03
39	//		
40	result1 = r_forward // [mol/s/l]	Assignm: result1 =	1.197707e-03
41	result2 = r_backward // [mol/s/l]	Assignm: result2 =	1.176305e-03

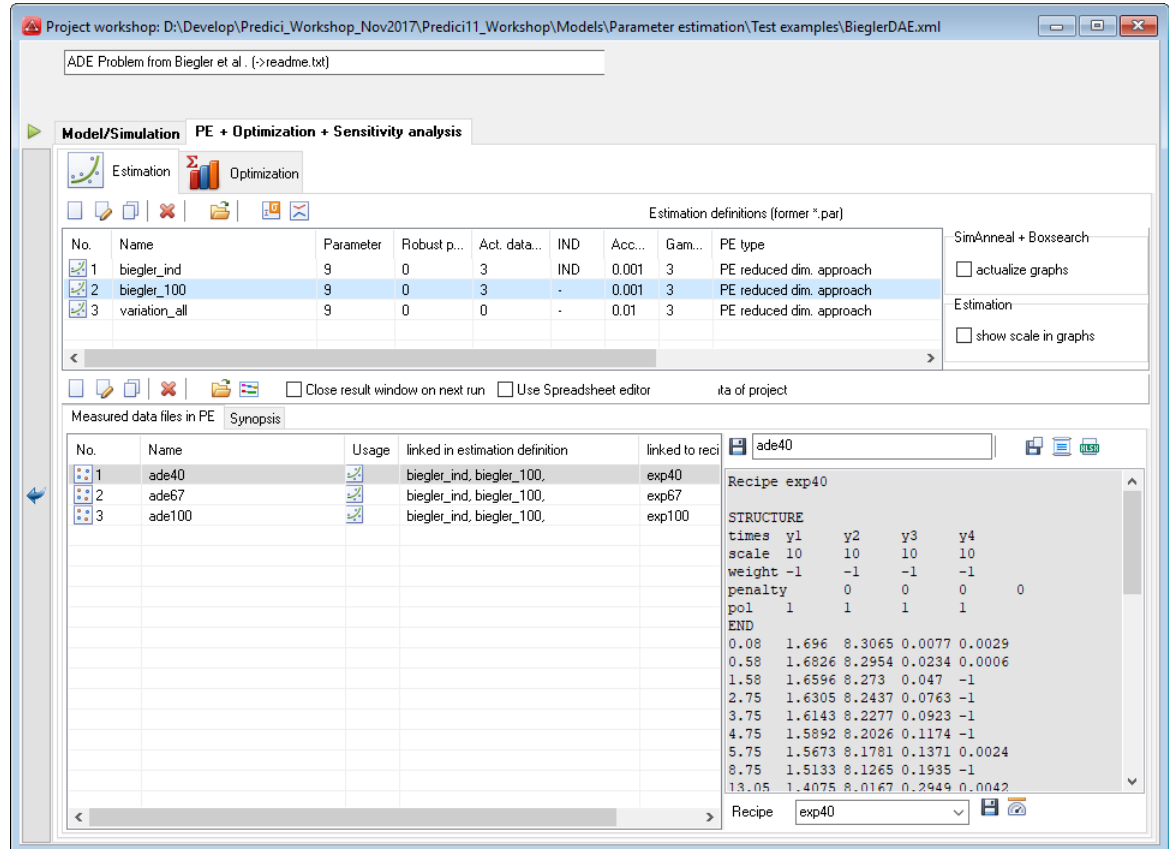
Reactor: Tube

2.1692e-02 2.0000e-01

Close

- ▶ Task: fit any set of model parameters vs. set of experimental data of all kind
 - ▶ concentrations
 - ▶ temperature
 - ▶ user-defined script expressions
- ▶ Important aspects
 - ▶ sensitivities
 - ▶ scaling
 - ▶ dependencies
- ▶ Algorithms
 - ▶ box search
 - ▶ simulated annealing
 - ▶ sophisticated Gauss-Newton algorithm with damping strategy
 - ▶ extension: algorithm to detect dependencies among parameters
 - ▶ **New:** Bayesian analysis and uncertainty quantification

- ▶ Definition includes all necessary information
 - ▶ parameter definitions
 - ▶ experimental data files and recipes
 - ▶ various settings



The screenshot shows a software window titled "Project workshop: D:\Develop\Predici_Workshop_Nov2017\Predici11_Workshop\Models\Parameter estimation\Test examples\BieglerDAE.xml". The main area displays "Estimation definitions (former *.par)" with the following table:

No.	Name	Parameter	Robust p...	Act. data...	IND	Acc...	Gam...	PE type
1	biegler_ind	9	0	3	IND	0.001	3	PE reduced dim. approach
2	biegler_100	9	0	3	.	0.001	3	PE reduced dim. approach
3	variation_all	9	0	0	.	0.01	3	PE reduced dim. approach

Below this table is a section for "Measured data files in PE" with a "Synopsis" view:

No.	Name	Usage	linked in estimation definition	linked to reci
1	ade40		biegler_ind, biegler_100,	exp40
2	ade67		biegler_ind, biegler_100,	exp67
3	ade100		biegler_ind, biegler_100,	exp100

On the right, the "Recipe exp40" editor shows the following structure:

```
STRUCTURE
times y1 y2 y3 y4
scale 10 10 10 10
weight -1 -1 -1 -1
penalty 0 0 0 0
pol 1 1 1 1
END
0.08 1.696 8.3065 0.0077 0.0029
0.58 1.6826 8.2954 0.0234 0.0006
1.58 1.6596 8.273 0.047 -1
2.75 1.6305 8.2437 0.0763 -1
3.75 1.6143 8.2277 0.0923 -1
4.75 1.5892 8.2026 0.1174 -1
5.75 1.5673 8.1781 0.1371 0.0024
8.75 1.5133 8.1265 0.1935 -1
13.05 1.4075 8.0167 0.2949 0.0042
```

Parameter estimation

- ▶ Values of fitted parameters including statistic values
- ▶ Estimation of essential degrees of freedom
- ▶ Detailed presentation of residuals and parameters

