

# Modelling tools & applications

ICAS-MoT

Rafiqul Gani

PSE for SPEED

Skyttemosen 6, DK-3450 Allerød, Denmark

rgani2018@gmail.com

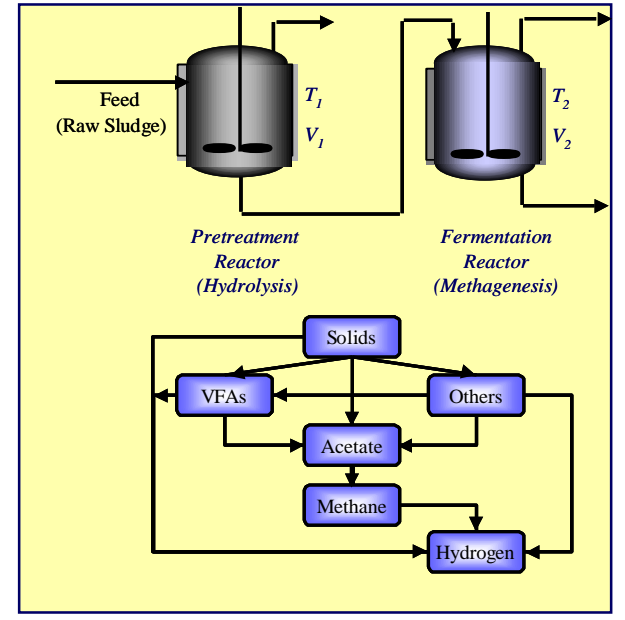
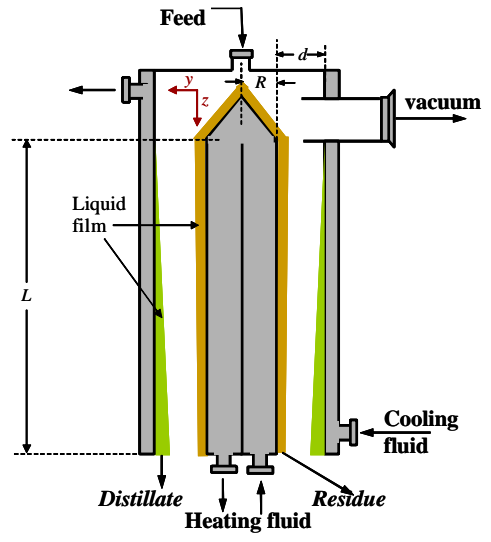
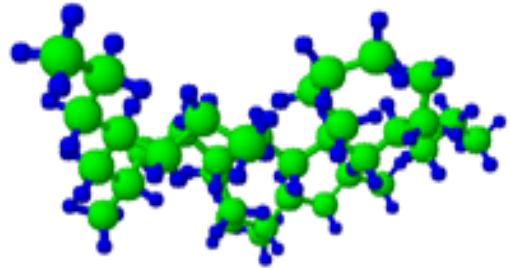
[www.pseforspeed.com](http://www.pseforspeed.com)

# Overview

- ❖ Modelling concept (summary)
- ❖ Tool for model analysis & solution (MoT)
- ❖ Examples with MoT

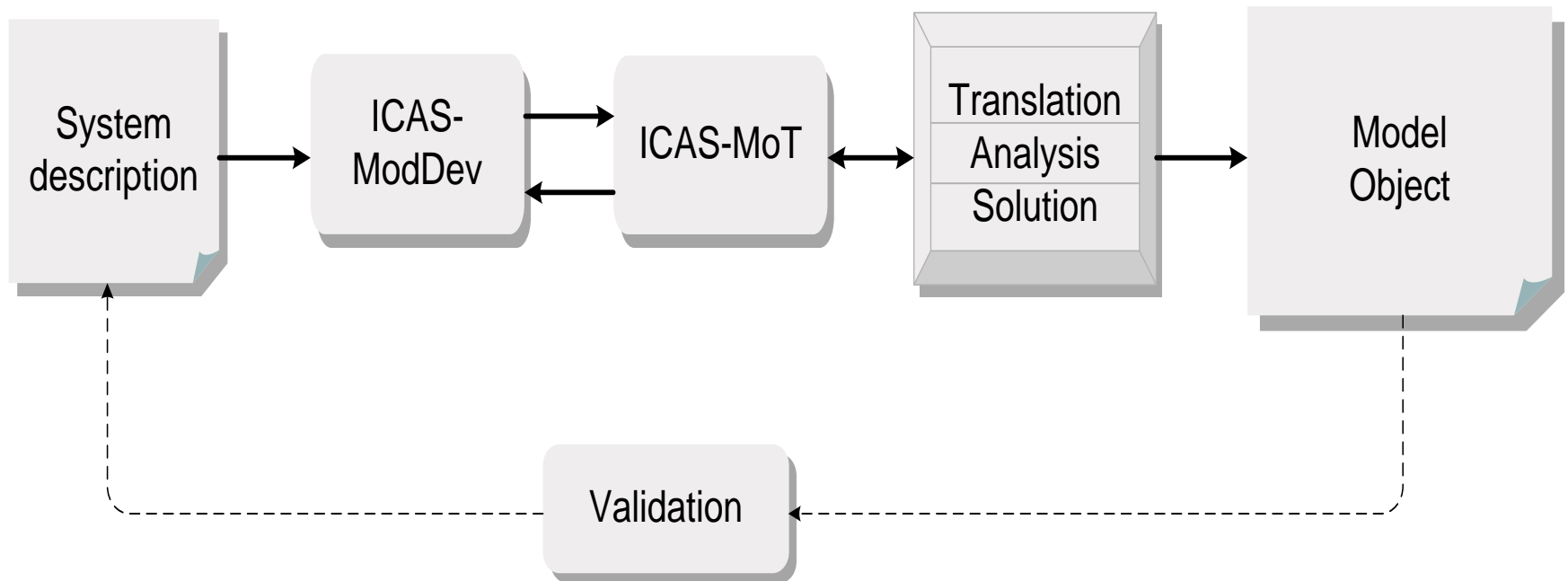
# Modelling concept - 1

For a given modelling task, generate (create) the mathematical model; analyze the model; solve the model; create a model object & finally use

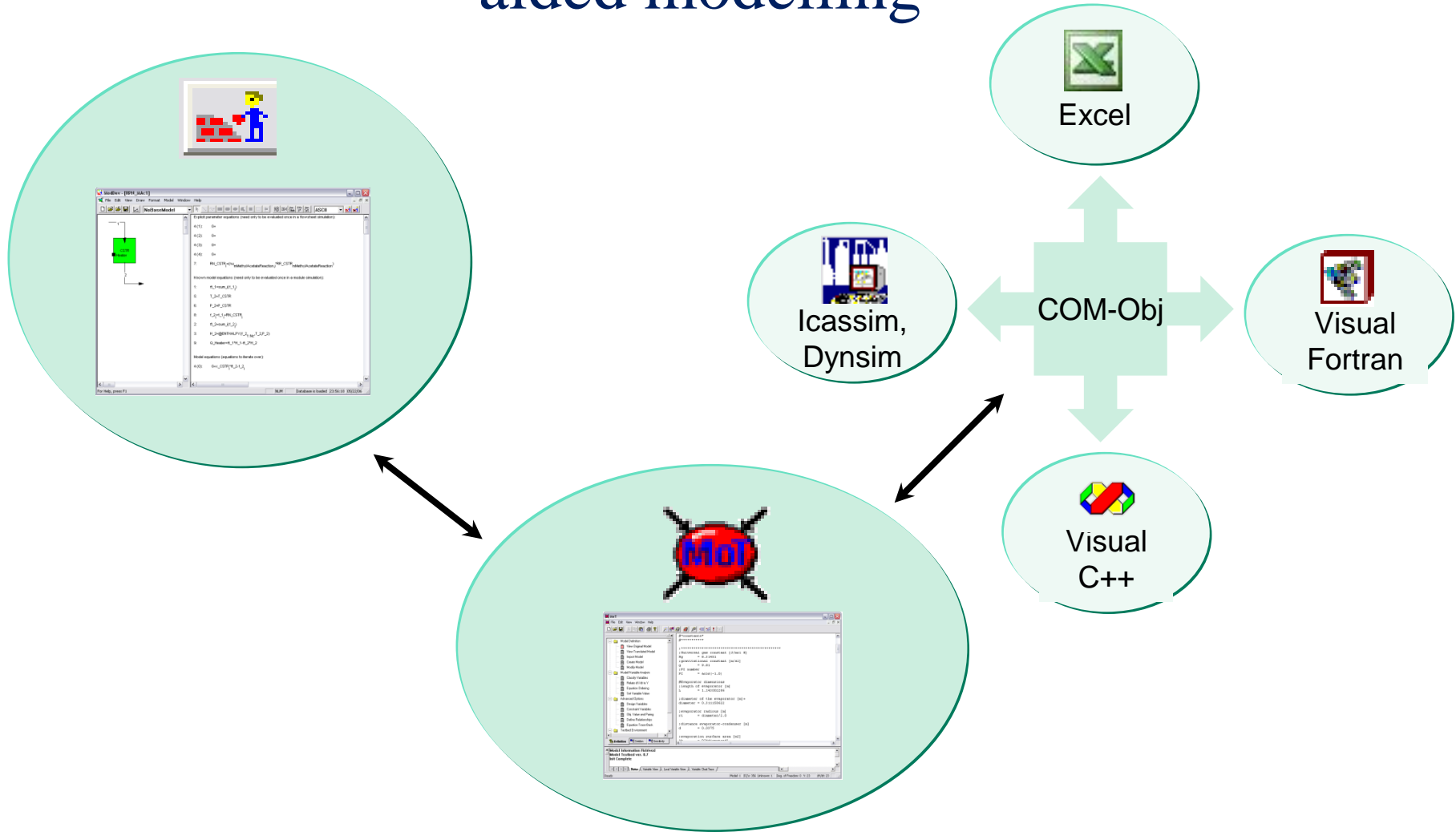


# Modelling concept - 2

**For a given modelling task, generate (create) the mathematical model; analyze the model; solve the model; create a model object & finally use**



# Modelling concept – 3: Integrated computer aided modelling

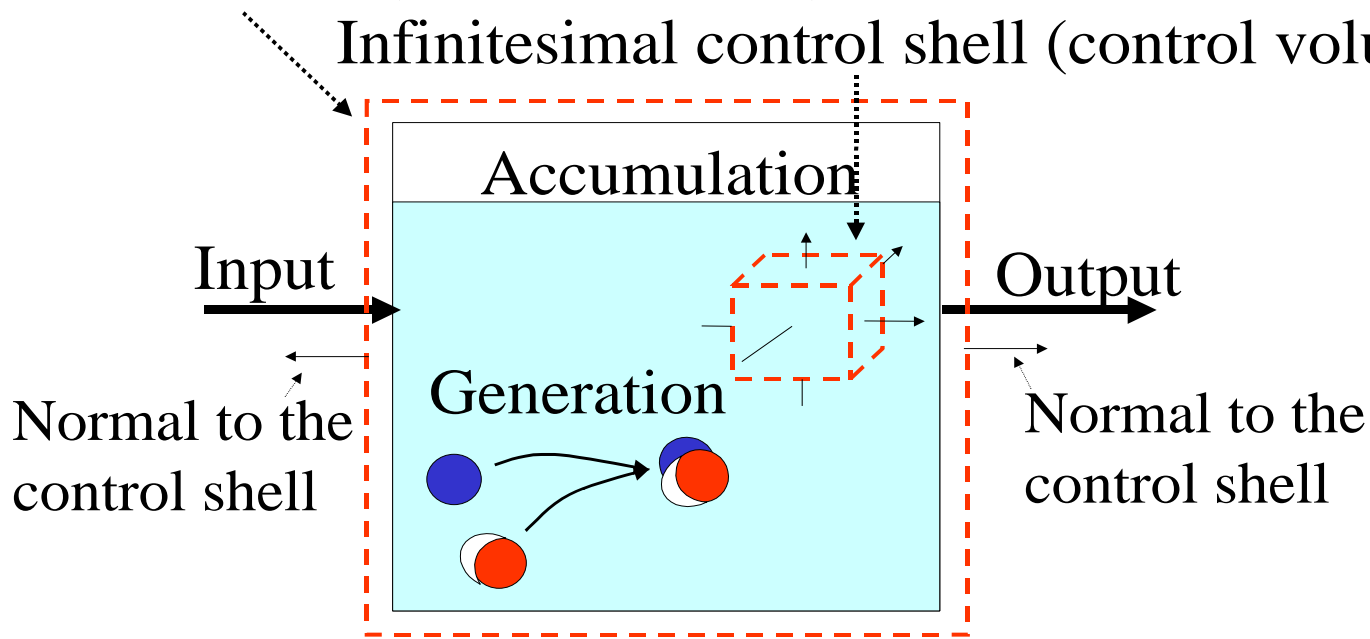


# Tool for model generation: ModDev -1

**Describe balance volume (control shell) & connection to surroundings; retrieve equations from model library; export model to MoT**

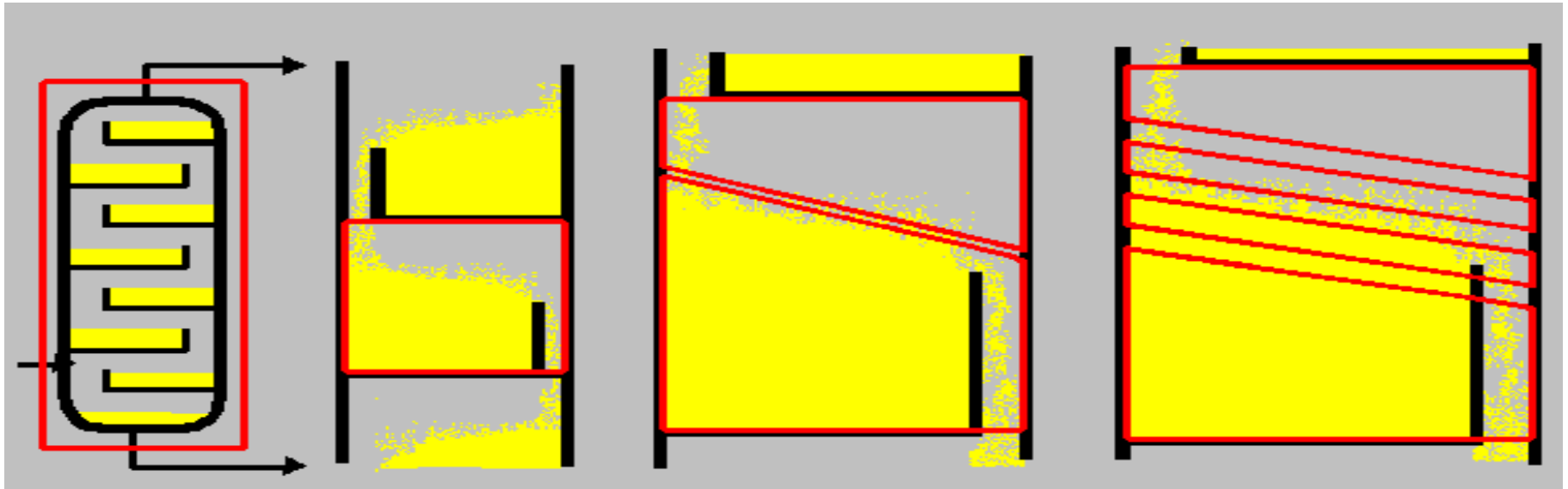
Control shell (control volume)

Infinitesimal control shell (control volume)



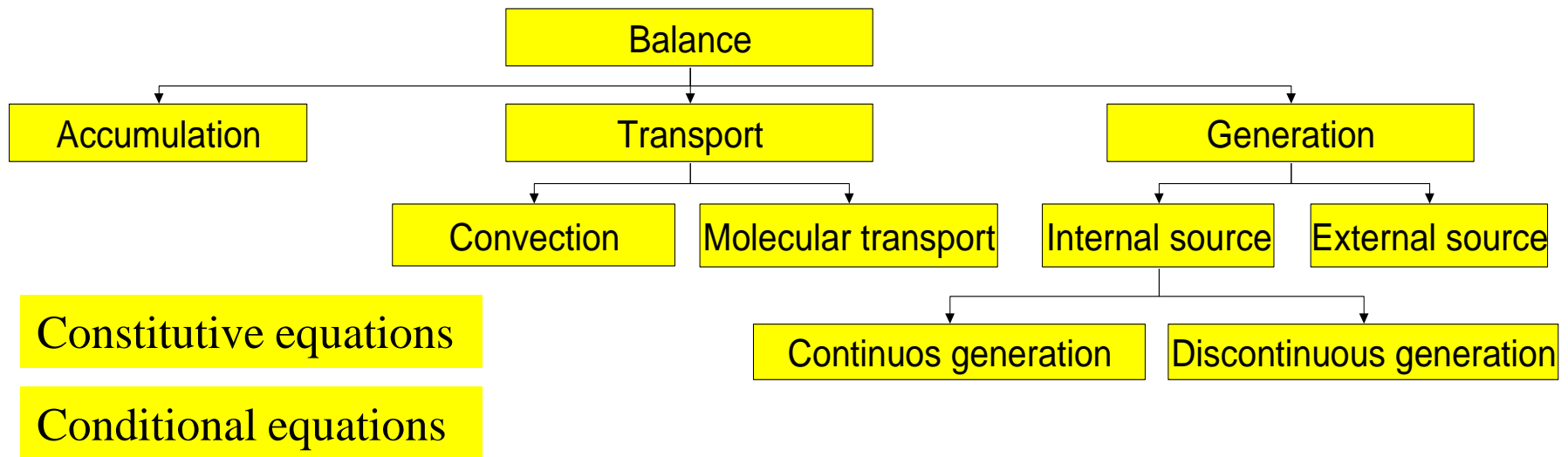
# Tool for model generation: ModDev -2

**Describe balance volume (control shell) & connection to surroundings; retrieve equations from model library; export model to MoT**



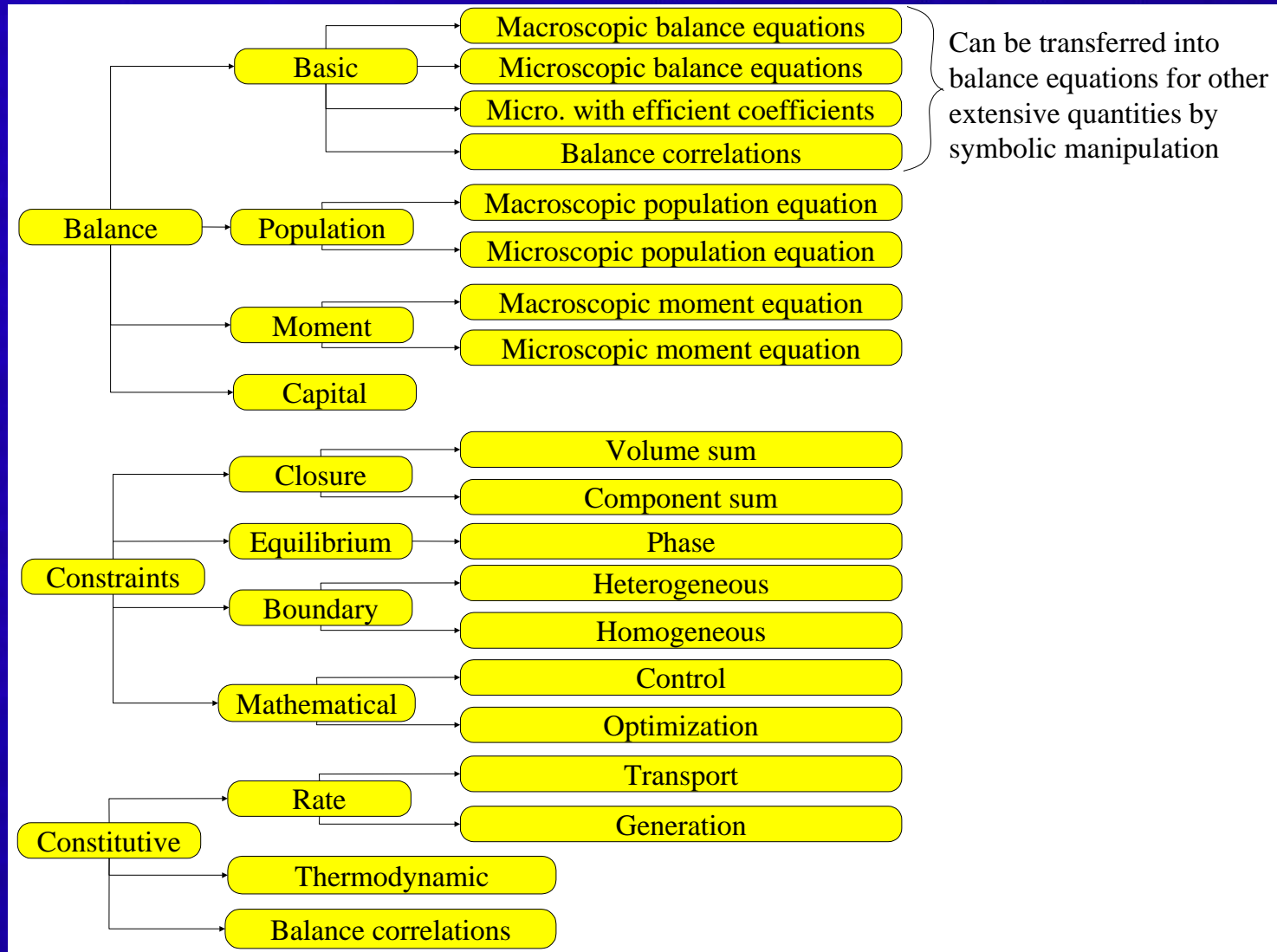
# Tool for model generation: ModDev -3

**Describe balance volume (control shell) & connection to surroundings; retrieve equations from model library; export model to MoT**



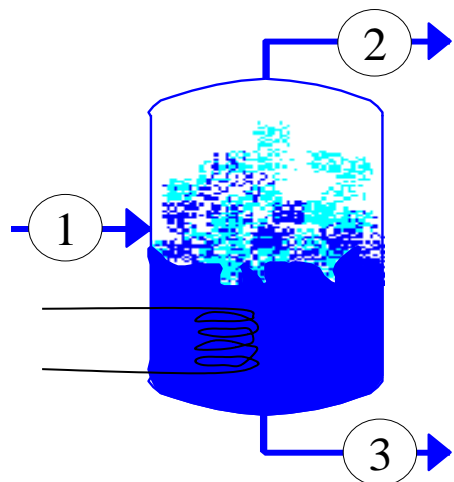


# Match problem description with Reference Models



Retrieve matched models otherwise build new models

# Tool for model generation: ModDev - example



## STREAM CONNECTION OBJECT

Name: 3

Models for quantities:

Energy (enthalpy):  $H_3 = @FUNC\_E(2, f_{3[]}, T_3, P_3)$

Models for the “from”-connection: (*equilibrium*)

Energy connection:  $T_3 = T_{flash}$

Momentum connection:  $P_3 = P_{flash}$

## SHELL OBJECT

Name: *flash*

Assumed phase condition: *Calculate (VL)*

Equilibrium model:  $0 = f_{2i}/ft_2 - K_{flash} * f_{3i}/ft_3$ ,  $@KEQ(T_{flash}, P_{flash}, f_{2[]}, f_{3[]}, \#K_{flash})$ , **no accumulation**, include **mass & energy balance**

## SHELL CONNECTION OBJECT

Name: *heater*

Connection models:

Energy connection:  $Q_{heater} = Q_{flash}$

# Tool for model generation: ModDev - example

## Model equations written in ASCII-text is exported to MoT

```
T1 = Tout
P1 = Pout
Ttank=Tout
Tv = T1
Pv = P1
ntot = sum_i(n[i])
zTank[i] = n[i]/ntot
```

```
#find k values
Psat[i] = (10^(DB_AntoineA[i] - DB_AntoineB[i]/(Tout-273.15+DB_AntoineC[i])))/760
K[i] = Psat[i]/Pout
x[i] = zTank[i]/(1+phi*(K[i]-1))
y[i] = x[i]*K[i]
```

```
#get the enthalpies
#hVap[i] = (Avap[i]*(1-(Tout/...
#hV[i]=((((E[i]*0.2*Tout+D[i]...
#hVr[i]=((((E[i]*0.2*Tref+D[...
#hL[i] = (((((E[i]*0.2*Tout+D...
#hLr[i] = (((((E[i]*0.2*Tref+D...
```

```
#get the densities
#dL[i] = A105[i]/B105[i]^(1+(1-Ttank/C105[i])^D105[i])
dL[i] = ADippr101[i]/BDippr101[i]^(1+(1-Ttank/CDippr101[i])^DDippr101[i])

DenL = 1 / sum_i( x[i]/dL[i] )
DenV = Pout/(0.08314*Tout)

|rachford rice
O = ntot*(1-phi)/DenL + ntot*phi/DenV - Vol
O = sum_i( zTank[i]*(1-K[i]) / (1 + phi*(K[i]-1)) )
Level = ntot*(1-phi)/(Area*DenL)
L = ValveL*Level
V = ValveV*(Pout-Pmin)
Hv = sum_i( (hV[i]-hVr[i]+hVap[i])*y[i] )
Hl = sum_i( (hL[i]-hLr[i])*x[i] )
O = (Hl*ntot*(1-phi) + Hv*ntot*phi - Htank)/1000

#update holdups
dndt[i] = F*z[i]-L*x[i]-V*y[i]
dHtankdt = Hf*F - Hl*L - Hv*V + q
```

**Note: all the model equations are not shown here**

# Tool for model analysis & solution: MoT

**Translated form of the model**

```

hLFeed[i] = (((E[i]*0.2*Tfeed+D[i]*0.25)*Tfeed+C[i]/3.0)*Tfeed+B[i]*0.5)*Tfeed+A[i]*Tfeed)/1
Hv = sum_i( hV[i]-hVr[i])*y[i]
HI = sum_i( hL[i]-hLr[i])*x[i]
Hf = sum_i( hLFeed[i]-hLr[i])*z[i]
ntot = sum_i(n[i])
zTank[i] = n[i]/ntot
dL[i] = A105[i]/B105[i]^(1+(1-Ttank/C105[i])^D105[i])
DenL = sum_i( x[i]*dL[i] )
DenV = Pout/(0.08314*Tout)
0 = ntot*(1-phi)/DenL + ntot*phi/DenV - Vol
0 = sum_i( zTank[i]*(1-K[i]) / (1 + phi*(K[i]-1)) )
0 = HI*ntot*(1-phi) + Hv*ntot*phi - Htank
Level = ntot*(1-phi)/(Area*DenL)
L = ValveL*Level
V = ValveV*(Pout-Pmin)
dndt[i] = F*z[i]-L*x[i]-V*y[i]
dHtankdt = Hf*F - HI*L - Hv*V + q
    
```

**Library**

**Explicit**

**Implicit**

**ODE**

==> Translator says: Model imported successfully.  
 Modelling Testbed ver. 0.1  
 Init Complete

Ready Model: 1 EQ's: 18 Unknowns: 0 Deg. of Freedom: 3 Y: 0 dY/dt: 0

Start Exploring - C:\rafique Microsoft PowerPoint - [rg... ICAS - ICAS1  
 Microsoft Visual C++ Runtime ... MoT Model Testbed [Untitled]

# Tool for model analysis & solution: MoT

## Model analysis & incidence matrix

**Model Definition**

- View Original Model
- View Translated Model
- Import Model
- Create Model
- Modify Model

**Model/Variable Analysis**

- Classify Variables
- Relate dY/dt to Y
- Equation Ordering
- Set Variable Value

**Advanced Options**

- Define Relationships
- Equation Trace Back

**Testbed Environment**

- Define Compounds
- Define Stream Composition

Classify Variables	Parameter	Unknown	Known	Dependent
dQEr	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
qO2max	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Fin	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Fout	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Gin	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Ein	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
VG	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
VE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Explicit Variables (Value for Last Solution)	Value
dVG	-0.4242029069
dVE	8.72540043602
dVX	1.18943182814
dV	0
OUR	0.07215741752
CPR	0.32633224144
EPR	0.01595478145
P	13.6168763066

Incidence Matrix	G	WG	V	E	VE	X	VX	dV	option2	qO2lim	Gm	Em	qGoxlim	EPR	option1	qEmpot	option4	qGm	option5	qG	option3	dVG	qGr	qGf	option7	qfOxlim	option6	qEr	qEm	qCO2	qO2
G=VG/V	1																														
E=VE/V			1																												
X=VX/V				1																											
dV=Fin-Fout								1																							
option2=qGmax									1																						
qO2lim=qO2max										1																					
Gm=G/MwG											1																				
Em=E/MwE												1																			
qGoxlim=qO2lim/kog													1																		
EPR=Em*kla*air														1																	
option1=qG0m+alph															1																
qEmpot=gamma*Em																1															
option4=qGoxlim																	1														
qGm=min(option1,op																		1													

Color Coding

- Use Color Coding
- Show Parameters
- Show Known Variables

Singularity Matrix - Only the Appropriate Equations and Variables will be shown

Variable Singularity Test

Click the 'Perform Test' to test all Perform Test

**Model Information Retrieved**

Status Variable View Local Variable View Variable Chart Trace

Ready Model: 1 EQ's: 36 Unknown: 0 Deg. of Freedom: 0 Y: 4 dY/dt: 4 Res

Eq. No.	Equation	Number of equations
1	$\ln q_i = b_n (Z - 1) - \ln(Z - B) - A/B(2\alpha_n^{0.5} - b_n) \ln(1 + B/Z)$ $i=1, NC$	$NC$
2	$a_n = (\alpha_n^{0.5} T_n / P_n^{0.5}) / \sum_j (x_j \alpha_j^{0.5} T_j / P_j^{0.5})$ $i=1, NC$	$NC$
3	$b_n = (T_n / P_n) / \sum_j (x_j T_j / P_j)$ $i=1, NC$	$NC$
4	$Z^2 - Z^2 + Z(A - B - B^2) - AB = 0$	1
5	$A = aP(RT)^2$	1
6	$B = bP(RT)$	1
7	$a = \sum_i x_i \sum_j x_j a_{ij}$ $i=1, NC; j=1, NC$	1
8	$b = \sum_i x_i \sum_j x_j b_{ij}$ $i=1, NC; j=1, NC$	1
9	$a_{ij} = (a_i a_j)^k (1 - k_{ij})$ $i=1, NC; j=1, NC$	$NC * NC$
10	$b_{ij} = (b_i + b_j) / 2$ $i=1, NC; j=1, NC$	$NC * NC$
11	$a_i = \psi_A \alpha_i (T)(R^2 T_n^2 / P_n)$ $i=1, NC$	$NC$
12	$b_i = \psi_B (RT_n / P_n)$ $i=1, NC$	$NC$
13	$\alpha_i(T) = [1 + m_i (1 - T_n^{0.5})]^2$ $i=1, NC$	$NC$
14	$m_i = 0.48 + 1.574 \omega_i - 0.176 \omega_i^2$ $i=1, NC$	$NC$
15	$T_n = T / T_n$ $i=1, NC$	$NC$
Total number of equations = $5 + 8NC + 2NC * NC$		
Total number of variables = $10 [T, P, R, \psi_A, \psi_B, a, b, A, B, Z] + 12NC [x, T, P, \alpha, T, \alpha]$ $m, a, b, a, b, \psi] + 3NC * NC [a, b, k] = 10 + 12NC + 3NC * NC$		

Table 2d: Incidence matrix of SRK EOS property model equations.

Eq.	Unknown Variables														
	$T_r$	$m$	$\alpha$	$\underline{b}$	$\underline{a}$	$\underline{b}$	$\underline{a}$	$a$	$b$	$B$	$A$	$Z$	$\underline{b}_r$	$\underline{a}_r$	$\underline{Q}$
15	*														
14		*													
13	*	*	*												
12				*											
11			*		*										
10				*		*									
9					*		*								
7							*	*							
8						*			*						
6									*	*					
5								*			*				
4 <sup>1</sup>										*	*	*			
3													*		
2			*											*	
1										*	*	*	*	*	*

<sup>1</sup>: requires the solution of a cubic equation in  $Z$

# Tool for model analysis & solution: MoT

**Calculator (debug) mode of MoT**

	Break	Equation	Result
1	<input type="checkbox"/>	$Q=VG/V$	0
2	<input type="checkbox"/>	$E=VE/V$	0
3	<input type="checkbox"/>	$X=VX/V$	2.5
4	<input type="checkbox"/>	$dV=Fin-Fout$	0
5	<input type="checkbox"/>	$option2=qGmax$	0.02
6	<input type="checkbox"/>	$qO2lim=qO2max$	0.00657
7	<input type="checkbox"/>	$Gm=G/MwG$	0
8	<input type="checkbox"/>	$Em=EMwE$	0
9	<input type="checkbox"/>	$qGoxlim=qO2lim/kog$	0.00285652
10	<input type="checkbox"/>	$EPR=Em*kla*air$	0
11	<input type="checkbox"/>	$option1=qG0m+alpha*(Gm-G0m)$	0.001325
12	<input type="checkbox"/>	$qEmpot=gamma*Em$	0
13	<input type="checkbox"/>	$option4=qGoxlim$	0.00285652
14	<input type="checkbox"/>	$qGm=min(option1,option2)$	0.001325
15	<input type="checkbox"/>	$option5=qEmpot$	0
16	<input type="checkbox"/>	$qG=MwG*qGm$	0.2385
17	<input type="checkbox"/>	$option3=qGm$	0.001325
18	<input type="checkbox"/>	$dVG=-qG*VX+Gin*Fin-G*Fout$	27.615
19	<input checked="" type="checkbox"/>	$qGr=min(option3,option4)$	
20	<input type="checkbox"/>	$qGf=max(qGm-qGr,0)$	
21	<input type="checkbox"/>	$option7=(qO2lim-kog*qGr)/koe$	
22	<input type="checkbox"/>	$qEoxlim=max(option7,0)$	
23	<input type="checkbox"/>	$option6=qEoxlim$	
24	<input type="checkbox"/>	$qEr=min(option5,option6)$	
25	<input type="checkbox"/>	$qEm=-qEr+keg*qGf$	
26	<input type="checkbox"/>	$qCO2=kc1*qGr+kc2*qGf+kc3*qEr$	
27	<input type="checkbox"/>	$qO2=kog*qGr+koe*qEr$	
28	<input type="checkbox"/>	$mum=YGr*qGr+YGf*qGf+YEr*qEr$	
29	<input type="checkbox"/>	$dQspec=(dQGGr*qGr+dQGf*qGf+dQER*qEr)/3600$	
30	<input type="checkbox"/>	$OUR=VX*alphaO2$	

	Current Value
qGr	0.00285652173913
option3	0.001325
option4	0.00285652173913

Ready Model: 1 EQ's: 36 Unknown: 0 Deg. of Freedom: 0 Y: 4 dY/dt: 4 Res



# Tool for model analysis & solution: MoT

## Select the appropriate solver options

**Verify Model Solution Options...**

Overall Solution Selection

- Algebraic Solution
- Function Evaluation Only
- Dynamic Solution**

Model Ordering Options

- Original Order
- Ordered
- Partitioned and Ordered**

Specify multiple runs

Algebraic Options | **Dynamic Options** | Optimizer Options

Dynamic Solver

- BDF**
- Runge-Kutta (5th order)

Integration Options

- Forward**
- Backward
- Forward then backward
- Backward then forward

Number of time steps:   Generate eigenvalue repo

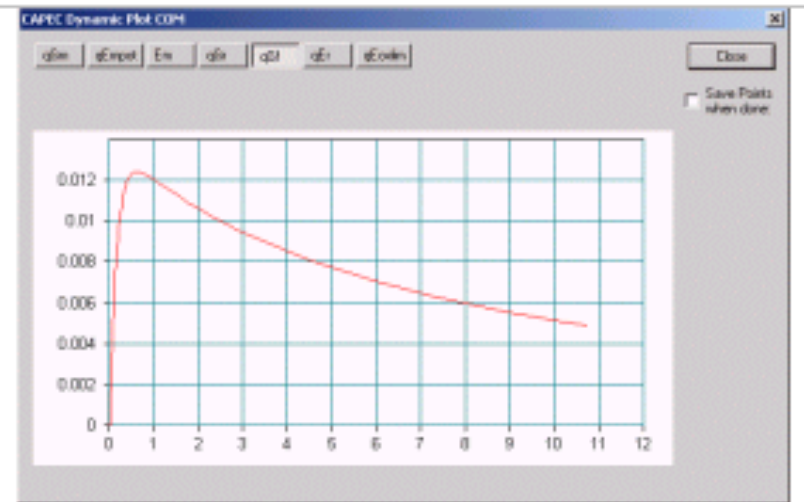
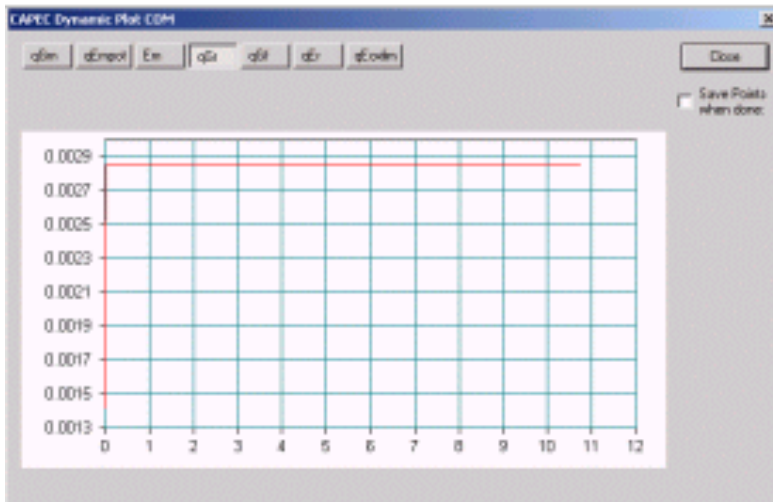
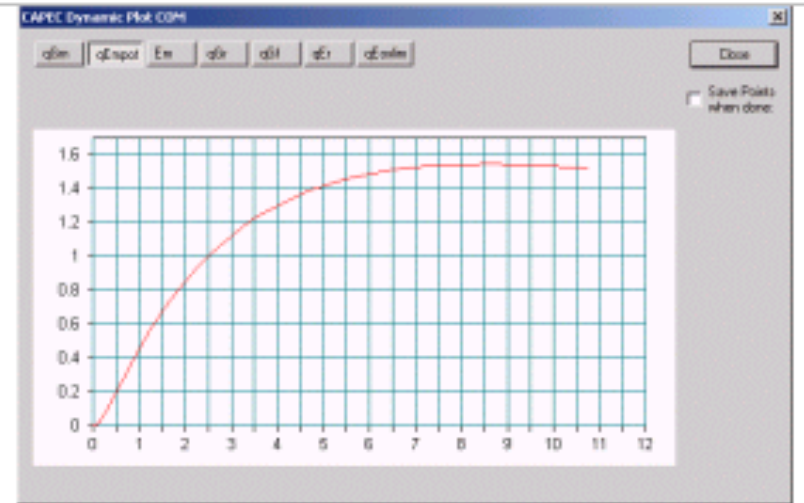
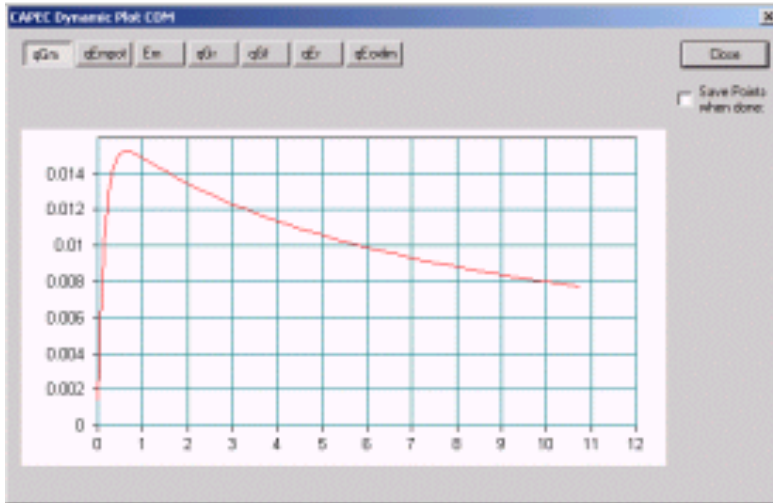
End time:   Run to steady state

Model scaling (p/w):    Steady state solution

Global error control (eps):  Criteria:

# Tool for model analysis & solution: MoT

## Visualization of simulation results



# Use of MoT model-objects

- Run MoT models on a stand-alone basis
- Run MoT models from a simulator (export MoT models to ICAS or other simulation engines)
- Run Mot models from external software environments (for example, EXCEL)
- Create customized simulator!

# Modelling exercise – MoT

Run MoT model objects from EXCEL

Note: It is necessary to first execute ComMOT.exe and then open the supplied EXCEL-macro (MoT\_Model\_Interface.xls)