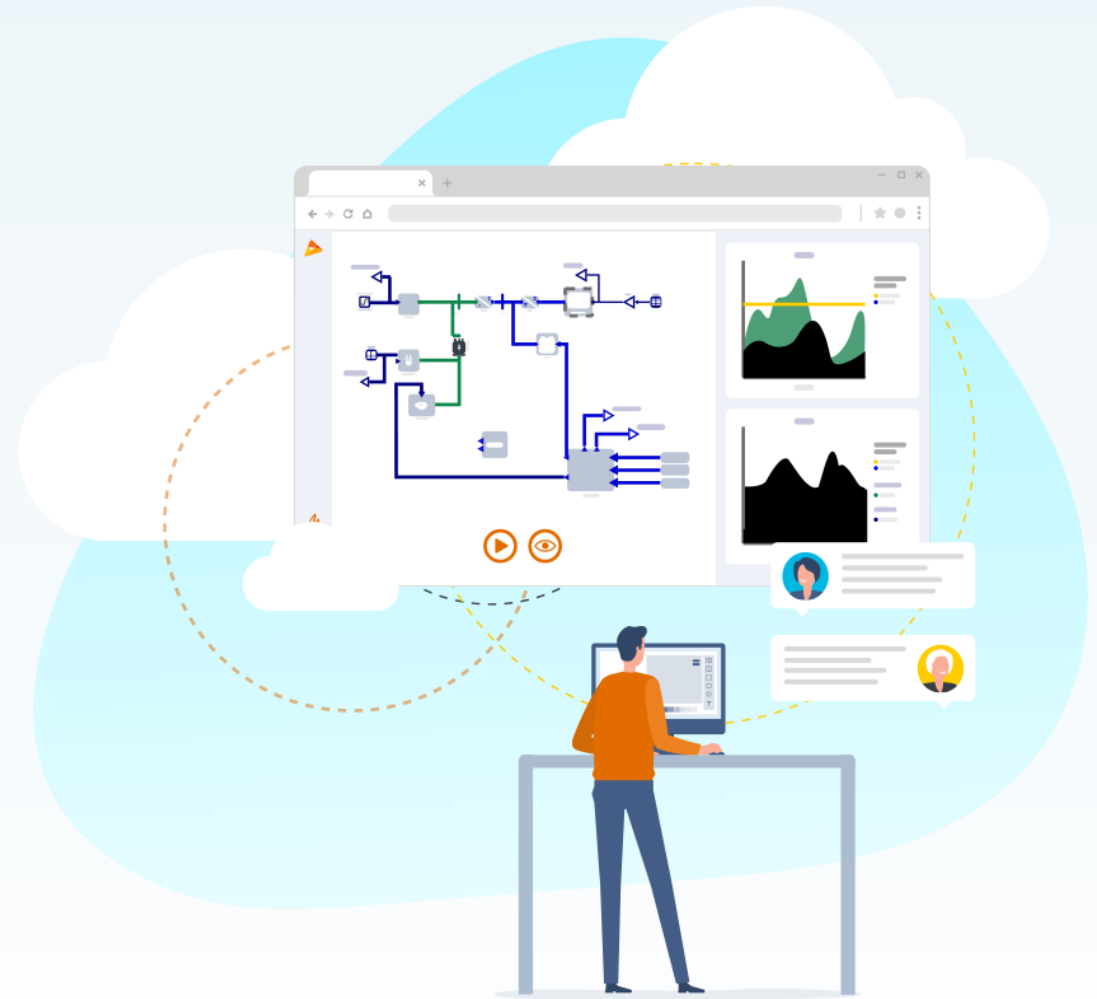


Model Initialization and Debugging

Presented by Modelon

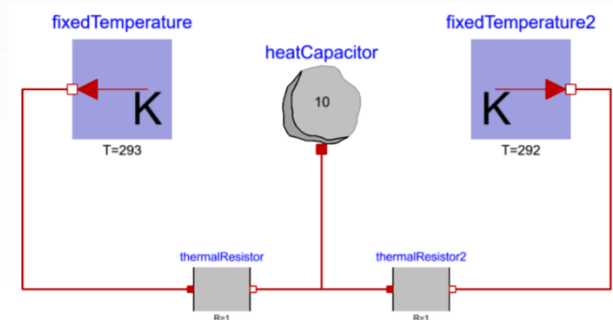


Equation based modeling

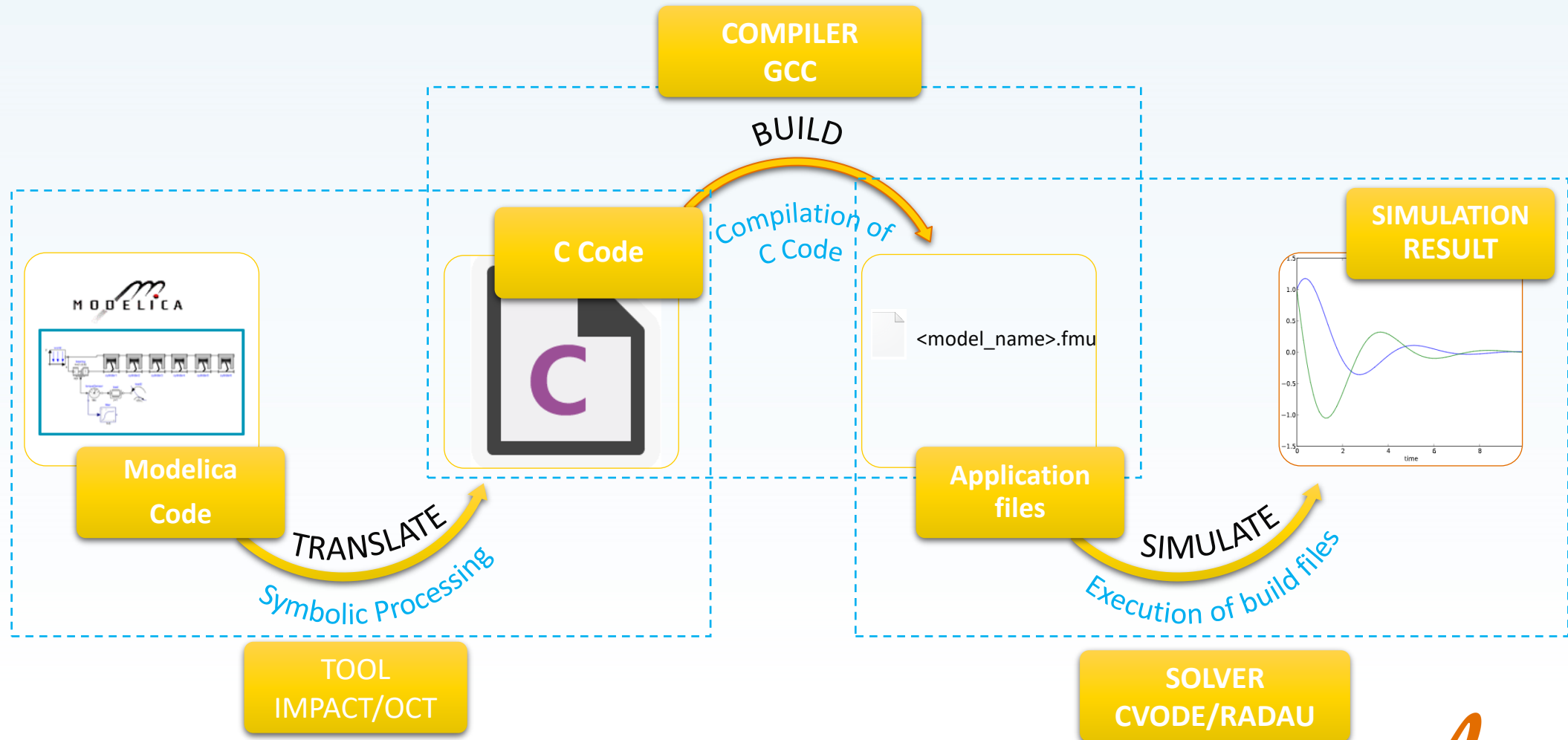
Where does it all come from?

- State variables
- Linear systems
- Nonlinear systems

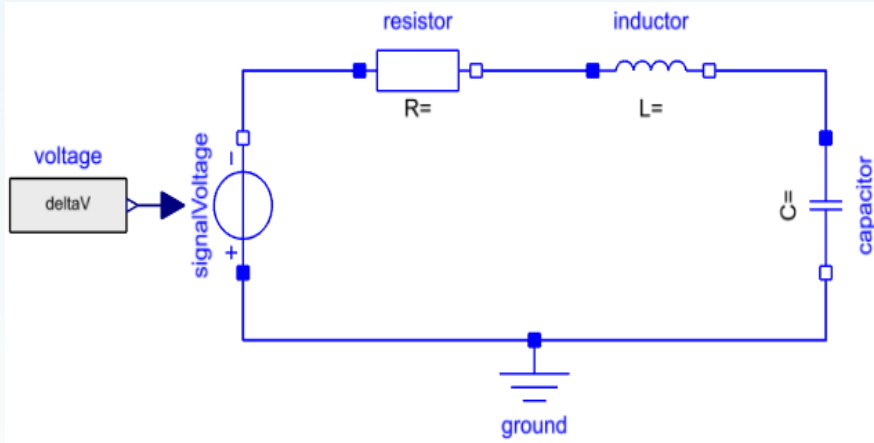
```
1 model HeatCapacitor
2   parameter Modelica.Units.SI.HeatCapacity C "Heat capacity of element (=cp*m)";
3
4   Modelica.Units.SI.Temperature T(start=293.15) "Temperature of element";
5   Modelica.Units.SI.TemperatureSlope der_T(start=0) "Time derivation of temperature (=der(T))";
6
7   .Modelica.Thermal.HeatTransfer.Interfaces.HeatPort_a port annotation(...);
8
9   equation
10    T = port.T;
11    der_T=der(T);
12    C*der(T)=port.Q_flow;
13    annotation(...);
14 end HeatCapacitor;
```



Compilation and Simulation



Causalization and sequencing



Reduce to minimum number of equations and select states

$$\begin{cases} u_0 = \Delta V & (1) \\ u_R = R i_R & (2) \\ u_L = L \frac{di_L}{dt} & (3) \\ i_C = C \frac{du_C}{dt} & (4) \\ i_R = i_0 & (5) \\ i_L = i_0 & (6) \\ i_C = i_0 & (7) \\ \Delta V = u_R + u_L + u_C & (8) \end{cases}$$

1 means: variable is in equation

Structure incidence matrix

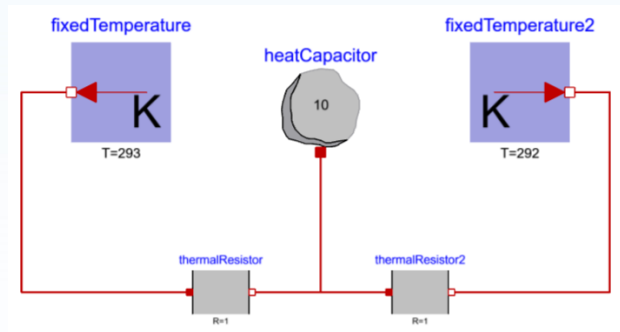
	u_0	i_0	u_R	i_R	u_L	$\frac{di_L}{dt}$	$\frac{du_C}{dt}$	i_C
(1)	1	0	0	0	0	0	0	0
(2)	0	0	1	1	0	0	0	0
(3)	0	0	0	0	1	1	0	0
(4)	0	0	0	0	0	0	1	1
(5)	0	1	0	1	0	0	0	0
(6)	0	1	0	0	0	0	0	0
(7)	0	1	0	0	0	0	0	1
(8)	1	0	1	0	1	0	0	0

Triangularization
(Block) Lower Triangle

	u_0	i_0	i_R	i_C	u_R	$\frac{du_C}{dt}$	u_L	$\frac{di_L}{dt}$
(1)	1	0	0	0	0	0	0	0
(6)	0	1	0	0	0	0	0	0
(5)	0	1	1	0	0	0	0	0
(7)	0	1	0	1	0	0	0	0
(2)	0	0	1	0	1	0	0	0
(4)	0	0	0	1	0	1	0	0
(8)	1	0	0	0	1	0	1	0
(3)	0	0	0	0	0	0	1	1

Modelon Impact diagnostics

- generate_html_diagnostics



Application Execution Export Units Storage Workspace

Dynamic Custom Linearize

SIMULATION OPTIONS

ncp 500

dynamic_diagnostics ☐

+ Add new

COMPILER OPTIONS

generate_html_diagnostics ☒

include_protected_variables ☐

c_compiler gcc

+ Add new

SOLVER OPTIONS

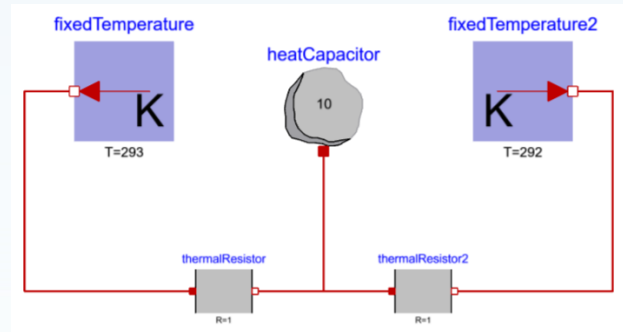
+ Add new

RUNTIME OPTIONS

+ Add new

CANCEL SAVE

Modelon Impact diagnostics



SIMULATIONS

- Result 3
- Result 2
- Result 1

☐ Only s

CALCULATED VALUES

Filter...

- ThermalWithState
 - fixedTemperature

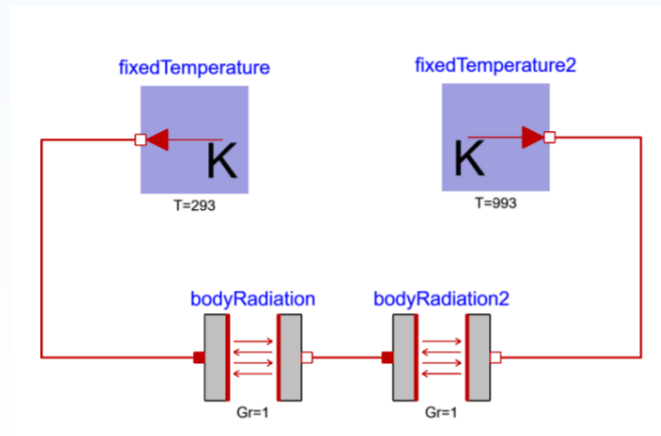
Context menu for Result 3:

- Show simulation log
- Show compilation log
- Download result (.mat)
- Download result (.csv)
- View diagnostics
- Rename
- Delete

	heatCapacitor.T	thermalResistor.dT	thermalResistor.Q_flow	thermalResistor2.dT	thermalResistor2.Q_flow	heatCapacitor.port.Q_flow	der(heatCapacitor.T)	heatCapacitor.der_T
heatCapacitor.T = 293.15	0							
thermalResistor.dT = fixedTemperature.port.T - heatCapacitor.T	0	0						
thermalResistor.dT = thermalResistor.R * thermalResistor.Q_flow		0	0					
thermalResistor2.dT = heatCapacitor.T - fixedTemperature2.T		0		0				
thermalResistor2.dT = thermalResistor2.R * thermalResistor2.Q_flow			0	0	0			
heatCapacitor.port.Q_flow + (- thermalResistor.Q_flow) = 0			0		0	0		
heatCapacitor.C * der(heatCapacitor.T) = heatCapacitor.port.Q_flow						0	0	0
heatCapacitor.der_T = der(heatCapacitor.T)							0	0

Linear and non-linear equations

- Some models are not possible to sort completely
- Blocks with interdependency along diagonal
 - Linear or non-linear system



	bodyRadiation.Q_flow	bodyRadiation.port_b.T	bodyRadiation.dT	bodyRadiation2.dT
bodyRadiation.Q_flow = bodyRadiation.Gr * \$	0	X		
bodyRadiation.Q_flow = bodyRadiation2.Gr *	0	X		
bodyRadiation.dT = bodyRadiation.port_a.T -		0	0	
bodyRadiation2.dT = bodyRadiation.port_b.T -		0		0

Separate Initial and Dynamic equation systems

- Modelica allows for separate equations for initial conditions

```
parameter Modelica.Units.SI.Temperature T_start "initial temperature";  
.Modelica.Units.SI.Temperature T "Temperature";  
.Modelica.Units.SI.HeatFlowRate q_flow(nominal=10000) "heat flow rate";  
.Modelica.Thermal.HeatTransfer.Interfaces.HeatPort_a port annotation(...);  
equation  
  der(T)*m*c = q_flow;  
  q_flow = port.Q_flow;  
  T = port.T;  
  
initial equation  
  T = T_start;
```

- This effectively means that the Initial problem and Dynamic problem, will generate separate code to be solved for

Best practice

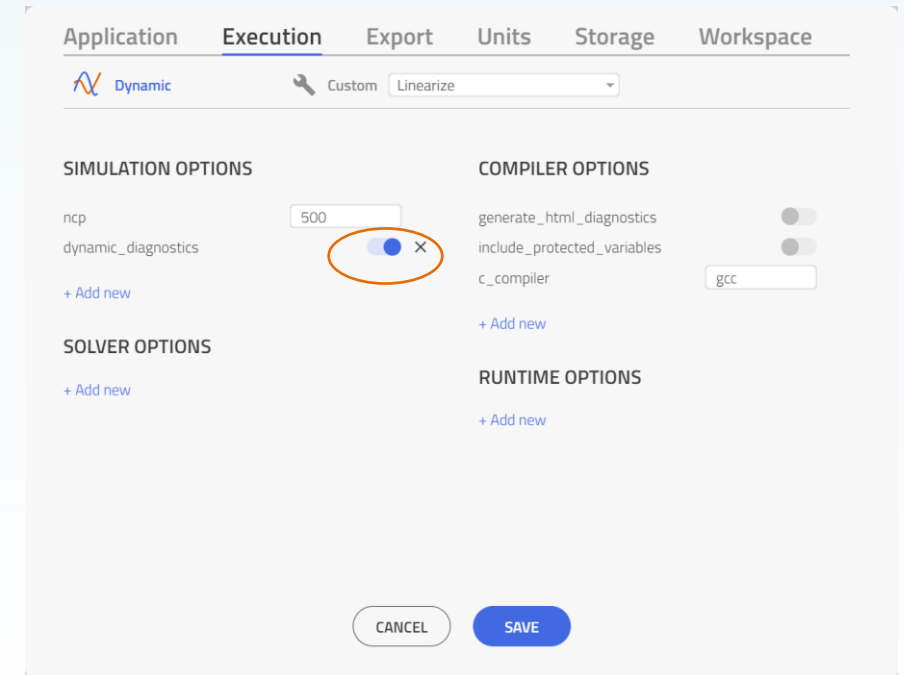
- Use diagnostics to get better understanding of your system
- Identify:
 - States
 - Nonlinear iteration variables
- Verify that they are initialized properly

Best practice

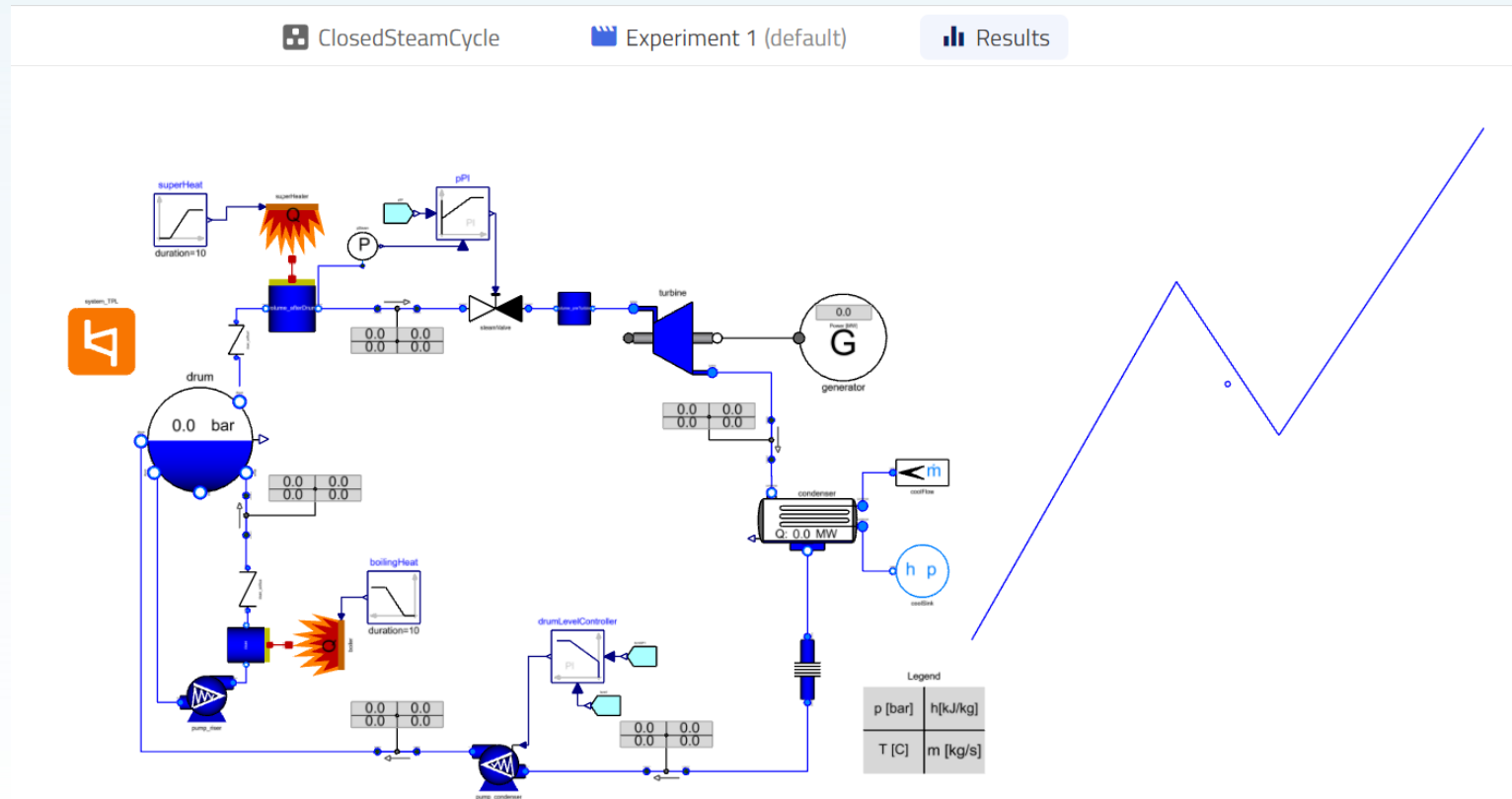
- Create subsystem tests for standard operating conditions
- Once tested and verified, integrate into system model
- Aggregate initial conditions using records or parameters that can be propagated

Dynamic diagnostics

- If your model is initializing but not running as expected
- Enable dynamic_diagnostics
- Simulation results will include solver diagnostics



Example: Closed Steam Cycle





Accurate Simulations. Better Decisions.