TECHNISCHE UNIVERSITÄT DRESDEN

Department of Architecture, Institute for Building Climate Control

Modelica Code Generators

Improving workflow efficiency and handling PDE submodels

Dr. Andreas Nicolai
andreas.nicolai@tu-dresden.de

Dresden, 23.04.2014

Modelica Code Generator

Modelica source code is generated by a tool based on input parameters/data selection such, that the generated model can be used afterwards *as if written manually*.

Advantage over manual Modelica code writing:

- faster
- less error prone
- standardized result / model code looks always the same

TECHNISCHE UNIVERSITÄT DRESDEN

Modelica Code Generator

Code generators should be considered when:

- many models should be generated, and
- effort in code generator development is amortized over effort on manual code writing (and debugging)
- large models (consisting of many sub-models) need to be created based on structured input data (e.g. BIM model data),
- documented equations and parameters are needed as compared to black-box component models (e.g. FMU)

Motivation – Work flow efficiency

Code generators may be useful for:

- Models with *large* parameter sets
- Models with *many* different parameter sets (Databases)

Examples:

- Material database, parameters for material behavior
- Plant component model, use "Manufacturers specs"

TECHNISCHE UNIVERSITÄT DRESDEN

Motivation – Work flow efficiency

Native Modelica work flow:

- drag & drop Modelica model
- edit (long) list of parameters

with scripts:

- parse Modelica model, and
- substitute parameters via substitution table, but
- *writing/managing scripts not efficient from modeling engineers point of view*

TECHNISCHE UNIVERSITÄT DRESDEN

Motivation – Work flow efficiency

Using Tool-specific functionality:

- drag & drop Modelica model
- choose model parameterization from parameter list table
- *which tool can do this currently?*
- *which tool provides enough flexibility to support engineers demands on data representation (for the purpose of selection/choosing data)?*

TECHNISCHE UNIVERSITÄT DRESDEN

Motivation – Partial Differential Equations

PDE transformation to ODE in Modelica:

- possible by manually entering discretized (FEM, FVM) equations in Modelica
- time consuming since discretization/grid needs to be incorporated
- equations are always the same, yet parameters for *each element/node differ*

Use code generation for specific PDE types and known parameterization!

Motivation – Partial Differential Equations

```
model Wall3 "A 1D Wall Model generated from Therakles"
  type HeatConductionCoefficient = Real(unit = "W/(m.K)", min = 0) "Effective heat cunduction coefficient";
  type ThermalConductivity = Real(unit = "W/(m2.K)", min = 0) "Thermal conductivity";
  parameter Integer nLayer = 2 "Number of material layers";
  parameter Modelica.SIunits.Temperature T0 = 293.15 "Initial temperature";
  parameter Modelica.SIunits.Density rho[nLayer] = {2100.0, 1100} "Material density";
  parameter Modelica.SIunits.SpecificHeatCapacity cp[nLayer] = {780.0, 820} "Material specific heat capacity";
  parameter ThermalConductivity lambda[nLayer] = {1.8, 0.04} "Material thermal conductivity";
  parameter Integer n = 3 "Number of discretization elements";
  parameter Modelica.SIunits.Length dx[n] = {0.1, 0.2, 0.08} "Element discretization";
  parameter Modelica.SIunits.Temperature T_left = 273.15 "Left temperature";
  parameter Modelica.SIunits.Temperature T_right = 293.15 "Right temperature";
  parameter HeatConductionCoefficient alpha_left = 8 "Left heat conduction coefficient";
  parameter HeatConductionCoefficient alpha_right = 25 "Right heat conduction coefficient";
  Modelica.SIunits.EnergyDensity u[n] "Energy density of the wall";
  Modelica.SIunits.Temperature T[n] "Wall temperature";
  Modelica.SIunits.HeatFlux q[n - 1] "Internal heat conduction flux";
  Modelica.SIunits.HeatFlux q_left "Left side heat conduction flux";
  Modelica.SIunits.HeatFlux q_right "Right side heat conduction flux";
initial equation
  // discretization elements 1 and 2 have material #1, element 3 uses material #2
  u[1] = rho[1] * cp[1] * T0;
  u[2] = rho[1] * cp[1] * T0;
  u[3] = rho[2] * cp[2] * T0;
equation
  u[1] = rho[1] * cp[1] * T[1] "Energy storage relation, element 1";
  u[2] = rho[1] * cp[1] * T[2] "Energy storage relation, element 2";
  u[3] = rho[2] * cp[2] * T[3] "Energy storage relation, element 3";
  q_left = alpha_left * (T_left - T[1]) "Left heat conduction boundary condition";
  q_right = alpha_right * (T[3] - T_right) "Right heat conduction boundary condition";
  q[1] = (dx[1] + dx[2]) / (dx[1] / lambda[1] + dx[2] / lambda[1]) * (T[1] - T[2]);
  q[2] = (dx[2] + dx[3]) / (dx[2] / lambda[1] + dx[3] / lambda[2]) * (T[2] - T[3]);
  der(u[1]) = (q_left - q[1]) / dx[1] "Energy conservation equation, element 1";
  der(u[2]) = (q[1] - q[2]) / dx[2] "Energy conservation equation, element 2";
  der(u[3]) = (q[2] - q_right) / dx[3] "Energy conservation equation, element 3";
end Wall3;
```

Writing Code Generators – A Tutorial

- Prototype code
- Substitution method: in-code / template file
- Performance/scalability test
- **Milestone: working submodels**
- combining models
- adding graphical annotations
- **Milestone: fully working (isolated) model**
- adding connections to the outside
- **Milestone: final model**

Start with manually written *Prototype* Model:

```
model Climate "Climate data"
  // Modelica standard library has no type for relative humidity, so we create one
  type RelativeHumidity = Real(unit = "1", min = 0, max = 1) "Relative humidity";
  output Modelica.SIunits.Temperature AmbientTemperature "Ambient temperature";
  output RelativeHumidity AmbientRelativeHumidity "Ambient relative humidity";
  Modelica.Blocks.Tables.CombiTable1D temperatureCombiTable(tableOnFile = false,
      table = [1,-2.6; 2,-3.9; 8759,0.7; 8760,-0.8]);
  Modelica.Blocks.Tables.CombiTable1D relhumCombiTable(tableOnFile = false,
      table = [0,0.8; 8760, 0.8]);
equation
  temperatureCombiTable.u[1] = time / 3600;
  AmbientTemperature = temperatureCombiTable.y[1] + 273.15;
  relhumCombiTable.u[1] = time / 3600;
  AmbientRelativeHumidity = relhumCombiTable.y[1] / 100.0;
end Climate;
```

TECHNISCHE UNIVERSITÄT DRESDEN

Replace parameterized code with placeholders:

```
model Climate "Climate data"
  // Modelica standard library has no type for relative humidity, so we create one
  type RelativeHumidity = Real(unit = "1", min = 0, max = 1) "Relative humidity";
  output Modelica.SIunits.Temperature AmbientTemperature "Ambient temperature";
  output RelativeHumidity AmbientRelativeHumidity "Ambient relative humidity";
  Modelica.Blocks.Tables.CombiTable1D temperatureCombiTable(tableOnFile = false,
       table = ${TEMPERATURE_DATA});
  Modelica.Blocks.Tables.CombiTable1D relhumCombiTable(tableOnFile = false,
       table = ${RELHUM_DATA});
equation
  temperatureCombiTable.u[1] = time / 3600;
  AmbientTemperature = temperatureCombiTable.y[1] + 273.15;
  relhumCombiTable.u[1] = time / 3600;
  AmbientRelativeHumidity = relhumCombiTable.y[1] / 100.0;
end Climate;
```

Substitution options

Parameterized code in template file:

- read template file (needs resource file handling)
- substitution placeholders
- write file

Parameterized file in-code:

- take compiled string
- substitute placeholders and write file
  (changes to template require recompile)

TECHNISCHE UNIVERSITÄT DRESDEN

Substitution option: in-code

```
model Climate
  type RelativeHumidity = Real(unit = "1", min = 0, max = 1) "Relative humidity";
  output Modelica.SIunits.Temperature AmbientTemperature "Ambient temperature";
  output RelativeHumidity AmbientRelativeHumidity "Ambient relative humidity";
…
```

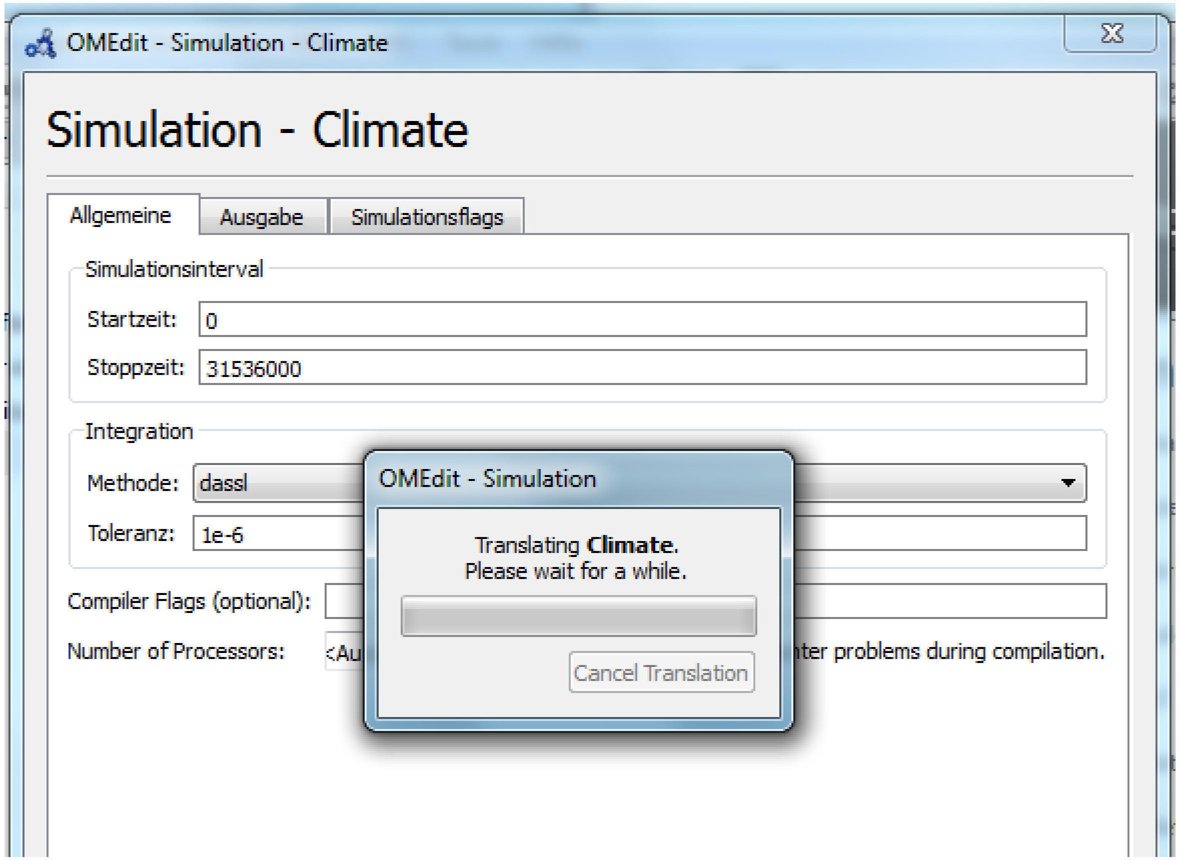
```
> python ascii2cpp.py climate.mo -o climate.c
```

```
const char * const CLIMATE_MO =
  "model Climate\n"
  "  type RelativeHumidity = Real(unit = \"1\", min = 0, max = 1) \"Relative humidity\";\n"
  "  output Modelica.SIunits.Temperature AmbientTemperature \"Ambient temperature\";\n"
  "  output RelativeHumidity AmbientRelativeHumidity \"Ambient relative humidity\";\n"
…
```

Generate model with real-world data (90 kB)

```
model Climate
  type RelativeHumidity = Real(unit = "1", min = 0, max = 1) "Relative humidity";
  output Modelica.SIunits.Temperature AmbientTemperature "Ambient temperature";
  output RelativeHumidity AmbientRelativeHumidity "Ambient relative humidity";
  Modelica.Blocks.Tables.CombiTable1D temperatureCombiTable(tableOnFile = false, table = [1,-2.6; 2,-
  3.9; 3,-4.6; 4,-3.9; 5,-3.3; 6,-2.9; 7,-2; 8,-1.5; 9,-1.2; 10,-0.6; 11,0.1; 12,0.5; 13,0.4; 14,0.9;
  15,1.2; 16,1.5; 17,1.7; 18,1.8; 19,1.6; 20,1.8; 21,1.8; 22,1.7; 23,1.9; 24,1.7; 25,2; 26,2.6; 27,2.
  6; 28,2.3; 29,2.3; 30,2; 31,1.6; 32,1.8; 33,1.5; 34,1.8; 35,2; 36,1.4; 37,1; 38,0.5; 39,-0.3; 40,-
  1.7; 41,-1.9; 42,-2.6; 43,-3.8; 44,-4.4; 45,-4.6; 46,-4.6; 47,-5; 48,-5.6; 49,-5.8; 50,-6.4; 51,-6.
  5; 52,-7.2; 53,-7.5; 54,-7.7; 55,-7.5; 56,-7.7; 57,-7.3; 58,-6.4; 59,-5.2; 60,-4.1; 61,-3; 62,-3.6;
  63,-3.9; 64,-4.8; 65,-5.9; 66,-7.4; 67,-7.9; 68,-8.5; 69,-9; 70,-9.6; 71,-10; 72,-10.6; 73,-11; 74,
  -11.3; 75,-11.6; 76,-12; 77,-12.2; 78,-12.5; 79,-12.6; 80,-12.9; 81,-13.4; 82,-12.3; 83,-10.6; 84,-
  7.9; 85,-3.7; 86,-4.6; 87,-5.3; 88,-6.2; 89,-6.7; 90,-6.8; 91,-6.9; 92,-7.2; 93,-8; 94,-9.2; 95,-9.
  8; 96,-9.9; 97,-10.8; 98,-10; 99,-10.5; 100,-10.7; 101,-10.8; 102,-11.6; 103,-11.3; 104,-12; 105,-
  11.9; 106,-10.9; 107,-9.3; 108,-5.1; 109,-1.3; 110,-3; 111,-2.8; 112,-3.5; 113,-4.9; 114,-6.1; 115,
  -6.3; 116,-7.1; 117,-7.2; 118,-7.1; 119,-7.2; 120,-6.5; 121,-6.8; 122,-5.1; 123,-4; 124,-3.4; 125,-
  2.7; 126,-1.8; 127,-1.6; 128,-0.8; 129,-0.4; 130,0; 131,1.1; 132,1.6; 133,1.6; 134,1.2; 135,1.2;
  136,1.3; 137,1.2; 138,1.5; 139,1.5; 140,1.6; 141,1.8; 142,2.1; 143,2.1; 144,2.1; 145,2.3; 146,2.3;
  147,2.3; 148,2.1; 149,2.1; 150,2.1; 151,1.8; 152,1.6; 153,1.3; 154,1.2; 155,1.3; 156,1.2; 157,1.1;
  158,1.2; 159,1.3; 160,1.1; 161,0.9; 162,0.7; 163,0.8; 164,0.6; 165,0.4; 166,0.3; 167,0.2; 168,0.1;
  169,0.1; 170,0.1; 171,0; 172,0; 173,-0.2; 174,-0.5; 175,-0.7; 176,-0.8; 177,-0.8; 178,-0.6; 179,-0.
  4; 180,-0.1; 181,0; 182,0.3; 183,0.2; 184,0.1; 185,0; 186,-0.1; 187,-0.1; 188,-0.2; 189,-0.4; 190,-
  0.6; 191,-1; 192,-1; 193,-1.1; 194,-1.1; 195,-1.1; 196,-1.3; 197,-1.2; 198,-1.1; 199,-0.9; 200,-0.
  8; 201,-0.8; 202,-0.7; 203,-0.6; 204,-0.5; 205,-0.5; 206,-0.6; 207,-0.8; 208,-1; 209,-0.9; 210,-1.
  1; 211,-1.3; 212,-1.4; 213,-1.3; 214,-1.4; 215,-1.4; 216,-1.7; 217,-1.9; 218,-1.9; 219,-1.7; 220,-
  1.5; 221,-1.3; 222,-1.2; 223,-1.1; 224,-1; 225,-0.8; 226,-0.9; 227,-0.9; 228,-0.9; 229,-0.9; 230,-
  0.9; 231,-0.9; 232,-1; 233,-1.1; 234,-1.2; 235,-1.3; 236,-1.4; 237,-1.6; 238,-1.8; 239,-2.1; 240,-
  2.1; 241,-2; 242,-2; 243,-2.2; 244,-2.6; 245,-3.1; 246,-3.7; 247,-4.1; 248,-4.8; 249,-5.1; 250,-6.
  1; 251,-5.8; 252,-5.1; 253,-4.6; 254,-4.5; 255,-4.3; 256,-4.1; 257,-4; 258,-3.8; 259,-3.8; 260,-4.
```

Performance Test?



TECHNISCHE UNIVERSITÄT DRESDEN

Performance Test?

Compiling the model : over 3 min

Simulation:

```
  0.00627968s [  4.1%] pre-initialization
  0.00263811s [  1.7%] initialization
1.47931e-005s [  0.0%] steps
   0.0167639s [ 10.9%] creating output-file
 0.000918406s [  0.6%] event-handling
  0.00669471s [  4.3%] overhead
    0.120671s [ 78.4%] simulation
    0.153981s [100.0%] total
```

Storing results for analysis: 6 seconds

Tool comparison:

OpenModelica 1.9.2: > 3 min (if portability is an issue, don‘t use this version)

SimX 3.6: 12 secs

Alternative version with external data files

```
model Climate
  type RelativeHumidity = Real(unit = "1", min = 0, max = 1) "Relative humidity";
  output Modelica.SIunits.Temperature AmbientTemperature "Ambient temperature";
  output RelativeHumidity AmbientRelativeHumidity "Ambient relative humidity";
  Modelica.Blocks.Tables.CombiTable1D temperatureCombiTable(tableOnFile = true, fileName =
  "/home/ghorwin/Documents/svn/Therakles_trunk/bin/debug/mo/climate_data.txt", tableName =
  "Temperature");
  Modelica.Blocks.Tables.CombiTable1D relhumCombiTable(tableOnFile = true,fileName =
  "/home/ghorwin/Documents/svn/Therakles_trunk/bin/debug/mo/climate_data.txt", tableName =
  "RelativeHumidity");
equation
  temperatureCombiTable.u[1] = time / 3600;
  AmbientTemperature = temperatureCombiTable.y[1] + 273.15;
  relhumCombiTable.u[1] = time / 3600;
  AmbientRelativeHumidity = relhumCombiTable.y[1] / 100.0;
  annotation(experiment(start = 0, stop = 31536000));
end Climate;
```

Absolute file paths (problem?!) but compilation time reduced to 1 second!

SimX and OpenModelica now run at same speed.

Wall model prototype development

Versions (→ `Prototype` folder):

- Single layer (`Wall1.mo`)
- Single layer, material referenced (`Wall2.mo`)
- Three layers, 2 materials referenced (`Wall3.mo`)
- 6 Layers, flat data model, full FVM code (`Wall4.mo`)
- left boundary condition uses input (`Wall5.mo`)
- right boundary condition uses temperature input and flow output (`Wall6.mo`)
- right boundary condition uses heatPort_a (`Wall7.mo`)

TECHNISCHE UNIVERSITÄT DRESDEN

Wall model prototype development

```
model Wall7 "A 1D Wall Model generated from Therakles"
  type HeatConductionCoefficient = Real(unit = "W/(m.K)", min = 0) "Effective heat
cunduction coefficient";
  type ThermalConductivity = Real(unit = "W/(m2.K)", min = 0) "Thermal conductivity";
  parameter Integer n = 6 "Number of discretization elements";
  parameter Modelica.SIunits.Temperature T0 = 293.15 "Initial temperature";
  parameter Modelica.SIunits.Density rho[n] = {2100.0, 2100.0, 2100.0, 1100, 1100, 1100}
"Material density";
  parameter Modelica.SIunits.SpecificHeatCapacity cp[n] = {780.0, 780.0, 780.0, 820,
820, 820} "Material specific heat capacity";
  parameter ThermalConductivity lambda[n] = {1.8, 1.8, 1.8, 0.04, 0.04, 0.04} "Material
thermal conductivity";
  parameter Modelica.SIunits.Length dx[n] = {0.005, 0.2, 0.02, 0.02, 0.04, 0.02}
"Element discretization";

...

  // heat flux across interior cell interfaces
  for i in 2:n loop
    q[i] = (dx[i - 1] + dx[i]) / (dx[i - 1] / lambda[i - 1] + dx[i] / lambda[i]) * (T[i
- 1] - T[i]);
  end for;
  for i in 1:n loop
    der(u[i]) = (q[i] - q[i + 1]) / dx[i] "Energy conservation equation";
  end for;
  // mind sign definition, q[n + 1] is positive when heat flows out of right side of
construction
  // but HeatPort.Q_flow is positive when heat is flowing into construction
  HeatPortRoom.Q_flow = -q[n + 1] * A;
end Wall7;
```

RoomSim model pulls everything together (but no room yet):

```
model RoomSim
  Climate clima;
  Wall1 theWall1;
  Wall2 theWall2;
  Wall3 theWall3;
  Wall4 theWall4;
equation
  connect(clima.AmbientTemperature, theWall1.T_left);
  connect(clima.AmbientTemperature, theWall2.T_left);
  connect(clima.AmbientTemperature, theWall3.T_left);
  connect(clima.AmbientTemperature, theWall4.T_left);
end RoomSim;
```

More Prototype development:

Towards working room model:

- model for room energy balance
- connectors for all components
- Therakles sim model combines all components
- Creating a package to bind them all

No graphical annotations yet!

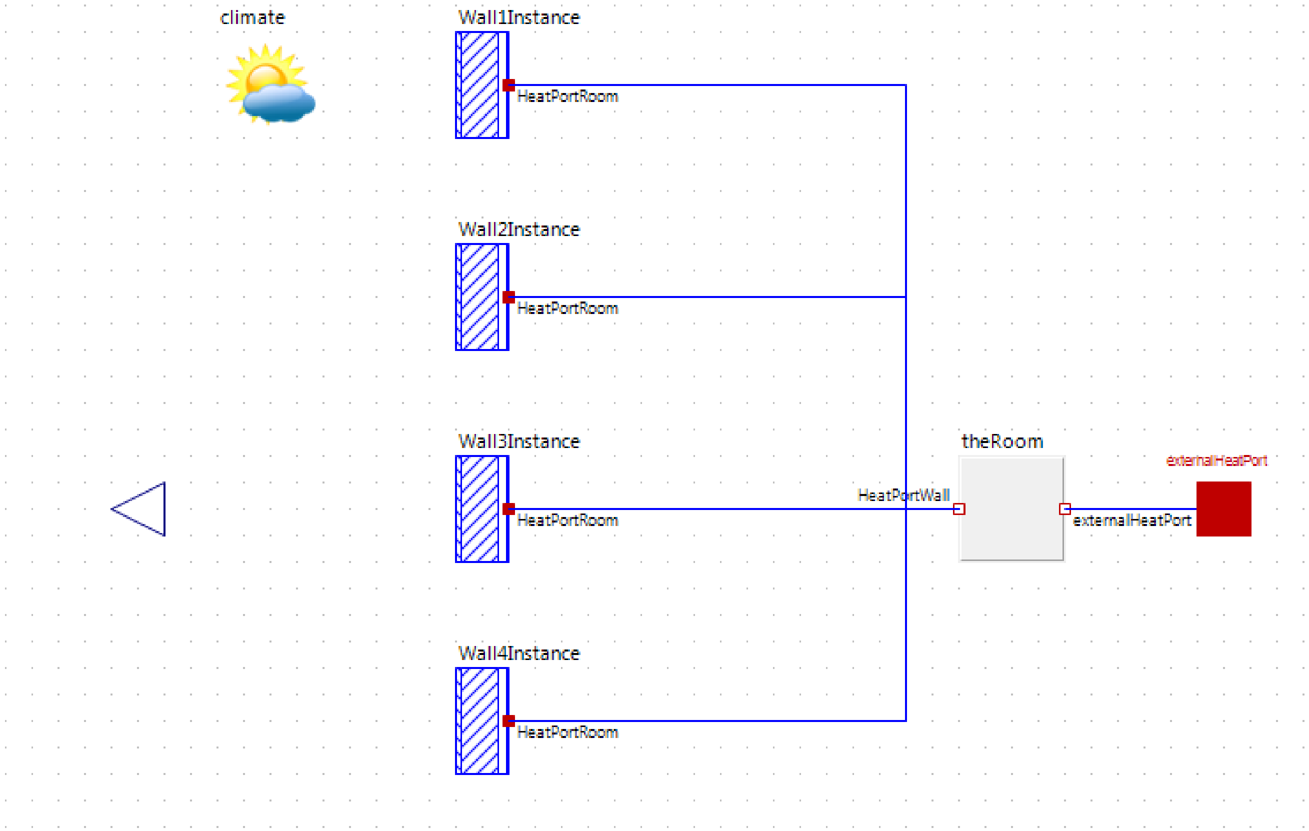
Adding annotations

- Images for components
- Overall simulation model
- Wall placement within Therakles component
- Dynamic connector placement (wall dependend)

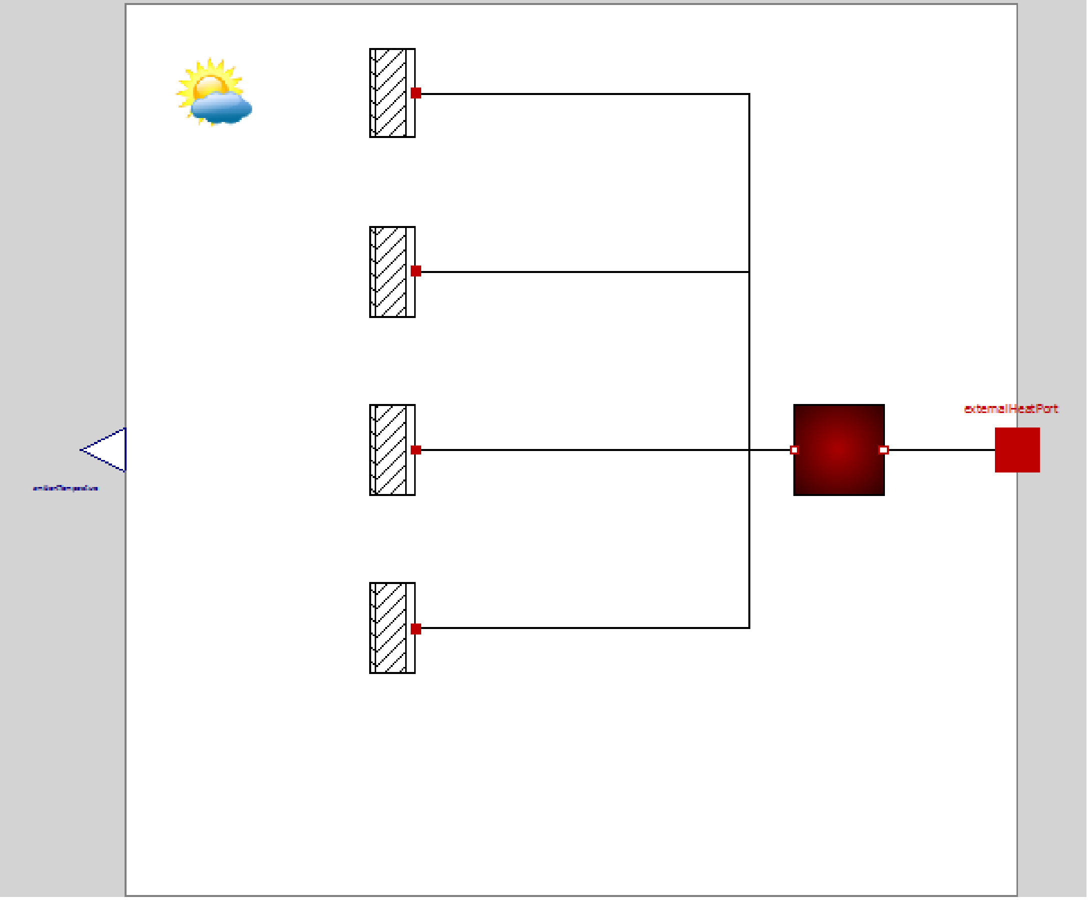
Problems: Tool-specific annotations

- annotations created in SimX won‘t load in OM
- shading types in OM won‘t show in SimX

Adding annotations (SimX View)



Adding annotations (OM View)



Summarizing – Workflow with Code Generator

- Specify simulation model in Therakles
- Export Modelica Model
- Import package into simulation environment
- Use just as any other component package...

Code Generator Development Effort

- 20 hours development time, including research of Modelica and tool-specific issues