KU LEUVEN

THERMAL SYSTEMS SIMULATION

Modelica, Dymola and IDEAS Crash Course 2022

Jelger Jansen

Javier Arroyo

Filip Jorissen

Lucas Verleyen

Who are we?

Jelger Jansen

Javier Arroyo

Lucas Verleyen

Filip Jorissen

The SySi Team

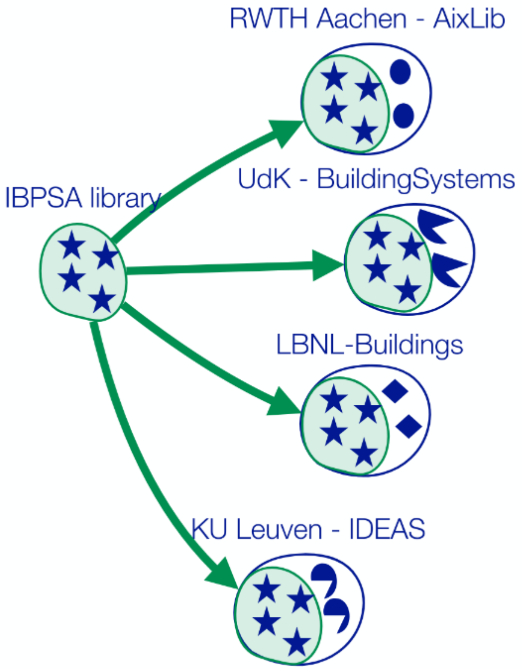
- Led by Professor Lieve Helsen
- Our Mission

  To sustainably use resources through integration and optimization of thermal systems performance in the built environment, including other energy vectors and sectors.

Motivation

- Why a crash course?
  - Introduction for our students and others who are interested in Modelica
  - Broaden the user base

- About IDEAS
  - Modelica users and library development since 2010
  - IDEAS v3.0, BaseClasses inherited from IBPSA project 1
    - https://github.com/open-ideas/IDEAS
    - https://github.com/ibpsa/modelica-ibpsa
  - Many models are validated in academic research
  - Main user base: researchers, students



- Builtwins: spin-off for sustainable control of buildings, using IDEAS

Agenda

Morning: Dymola and Modelica

- 9:30 - 10:00 Lecture 1
  - What is Modelica? What is Dymola? What is OpenModelica?
  - Modelica/Dymola basics
- 10:00 - 10:30 Exercise 1
- 10:30 - 10:45 Break
- 10:45 - 11:30 Lecture 2
  - Create new models/packages
  - Modelling with several components
  - Use connectors
  - Set parameters/propagate parameters
- 11:30 - 12:30 Exercise 2
- 12:30 - 13:30 Lunch break

Afternoon: IDEAS

- 13:30 - 16:00 Lecture 3 + Exercise 3
  - What is IDEAS?
  - IDEAS workflow
- 16:00 - 16:30 Break
- 16:30 - 18:00 Lecture 4 + Exercise 4
  - Hydronic models
  - Discrete control logic

Part 1: Introduction to Modelica and Dymola

Javier Arroyo

Modelica

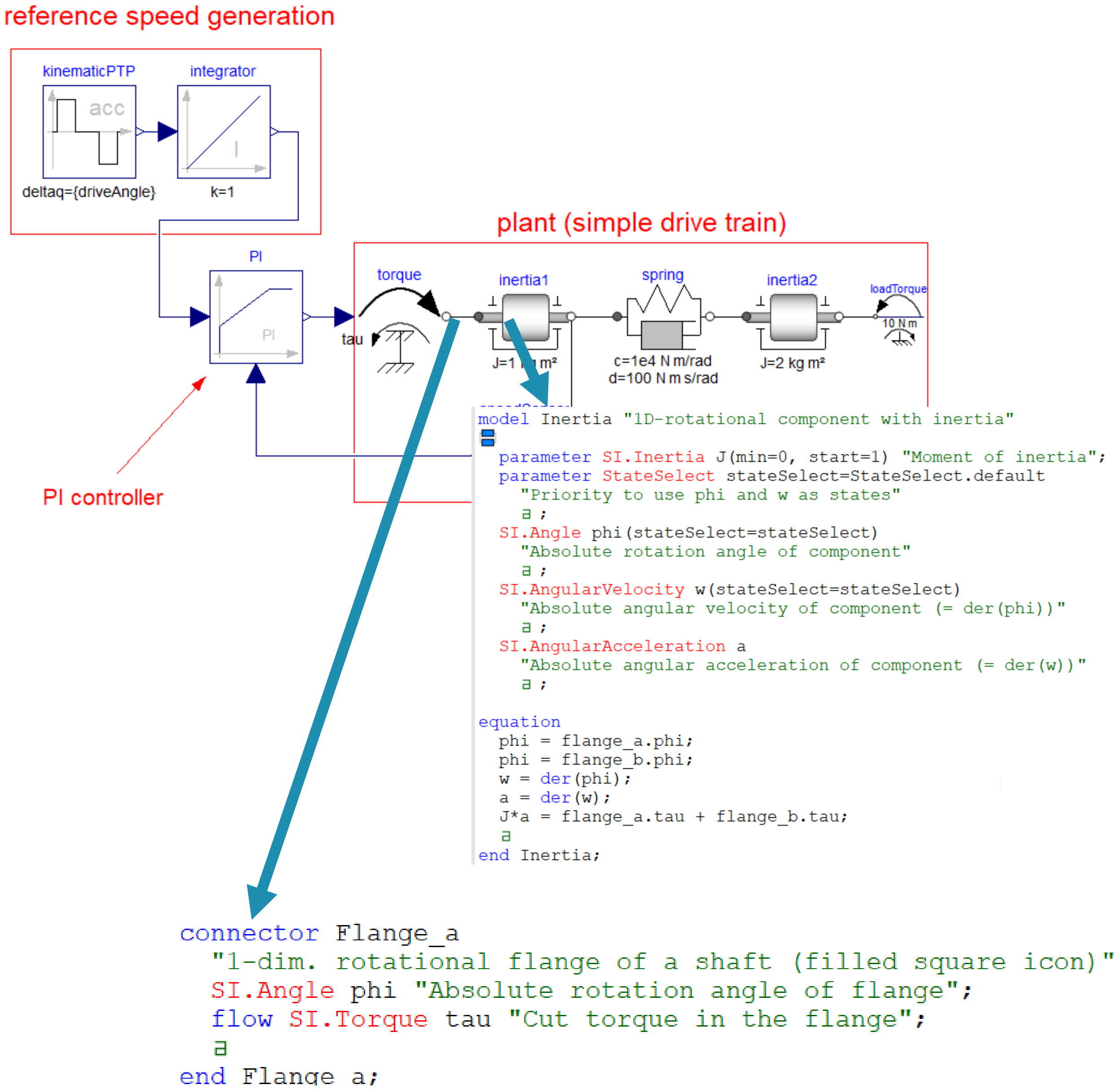
Modelica is a **modelling language** for modelling physical systems

- Language specification is open source
- object oriented
- Acausal modeling (equation-based)
- Multi-domain
- Primarily for simulation, but usable for optimization
- Small and large models (> 100 000 equations)
- Large community with many model libraries, especially in automotive industry (free and commercial)
- Textual and graphical modelling

Modelica

Object-oriented physical equation-based modelling

- A model represents a physical component
- Component is composed of sub-components and/or is described by equations
  → hierarchical structure
- Components can be connected to each other using connectors (=physical coupling)
- To simulate Modelica models, a Modelica simulation environment is needed



```
model Inertia "1D-rotational component with inertia"

  parameter SI.Inertia J(min=0, start=1) "Moment of inertia";
  parameter StateSelect stateSelect=StateSelect.default
    "Priority to use phi and w as states"
    ə ;
  SI.Angle phi(stateSelect=stateSelect)
    "Absolute rotation angle of component"
    ə ;
  SI.AngularVelocity w(stateSelect=stateSelect)
    "Absolute angular velocity of component (= der(phi))"
    ə ;
  SI.AngularAcceleration a
    "Absolute angular acceleration of component (= der(w))"
    ə ;

equation
  phi = flange_a.phi;
  phi = flange_b.phi;
  w = der(phi);
  a = der(w);
  J*a = flange_a.tau + flange_b.tau;
  ə
end Inertia;
```

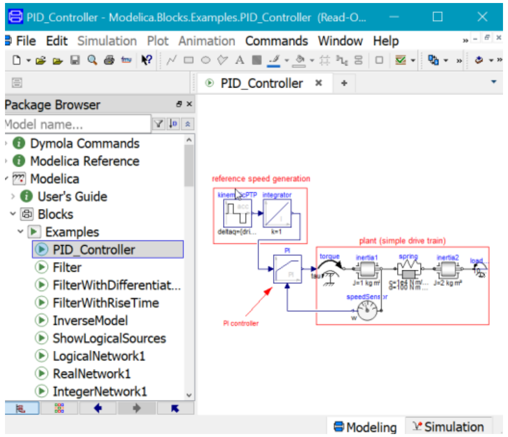
```
connector Flange_a
  "1-dim. rotational flange of a shaft (filled square icon)"
  SI.Angle phi "Absolute rotation angle of flange";
  flow SI.Torque tau "Cut torque in the flange";
  ə
end Flange_a;
```

Dymola

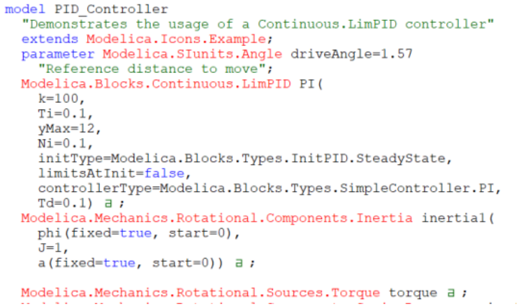
Dymola is a **commercial Modelica simulation environment**

Live demo of features:

- Icon, Diagram, Editor, Info
- Package browser, modelling, simulation
- Set up (compiler), run
- Adapt parameter
- Load libraries
- Look at simulation results: plot, zoom, filter variable, plot as a function of other variable.
- Try Simulate and plot (IDEAS library)
- Open sub-components
- Documentation



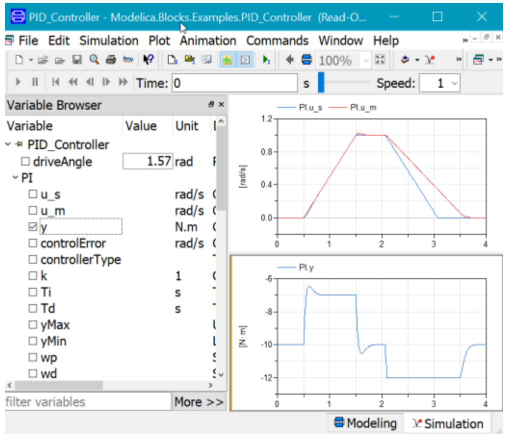
- Graphical editor
- Modelica simulation environment



```
model PID_Controller
  "Demonstrates the usage of a Continuous.LimPID controller"
  extends Modelica.Icons.Example;
  parameter Modelica.SIunits.Angle driveAngle=1.57
    "Reference distance to move";
  Modelica.Blocks.Continuous.LimPID PI(
    k=100,
    Ti=0.1,
    yMax=12,
    Ni=0.1,
    initType=Modelica.Blocks.Types.InitPID.SteadyState,
    limitsAtInit=false,
    controllerType=Modelica.Blocks.Types.SimpleController.PI,
    Td=0.1) a ;
  Modelica.Mechanics.Rotational.Components.Inertia inertial(
    phi(fixed=true, start=0),
    J=1,
    a(fixed=true, start=0)) a ;

  Modelica.Mechanics.Rotational.Sources.Torque torque a ;
```

- Textual description (Modelica language)

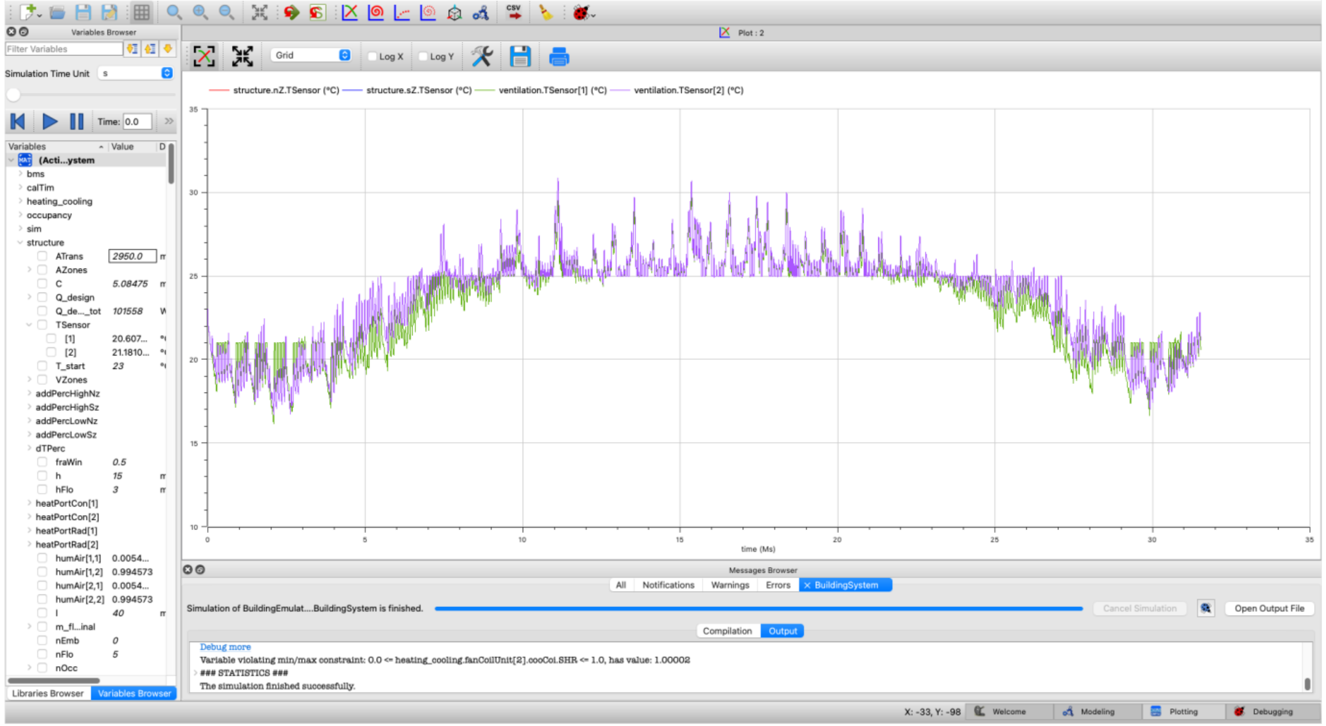


- Translation of Modelica code into executable C-code
- Coupling with a solver
- Visualization of results

OpenModelica

OM is a **free Modelica simulation environment**

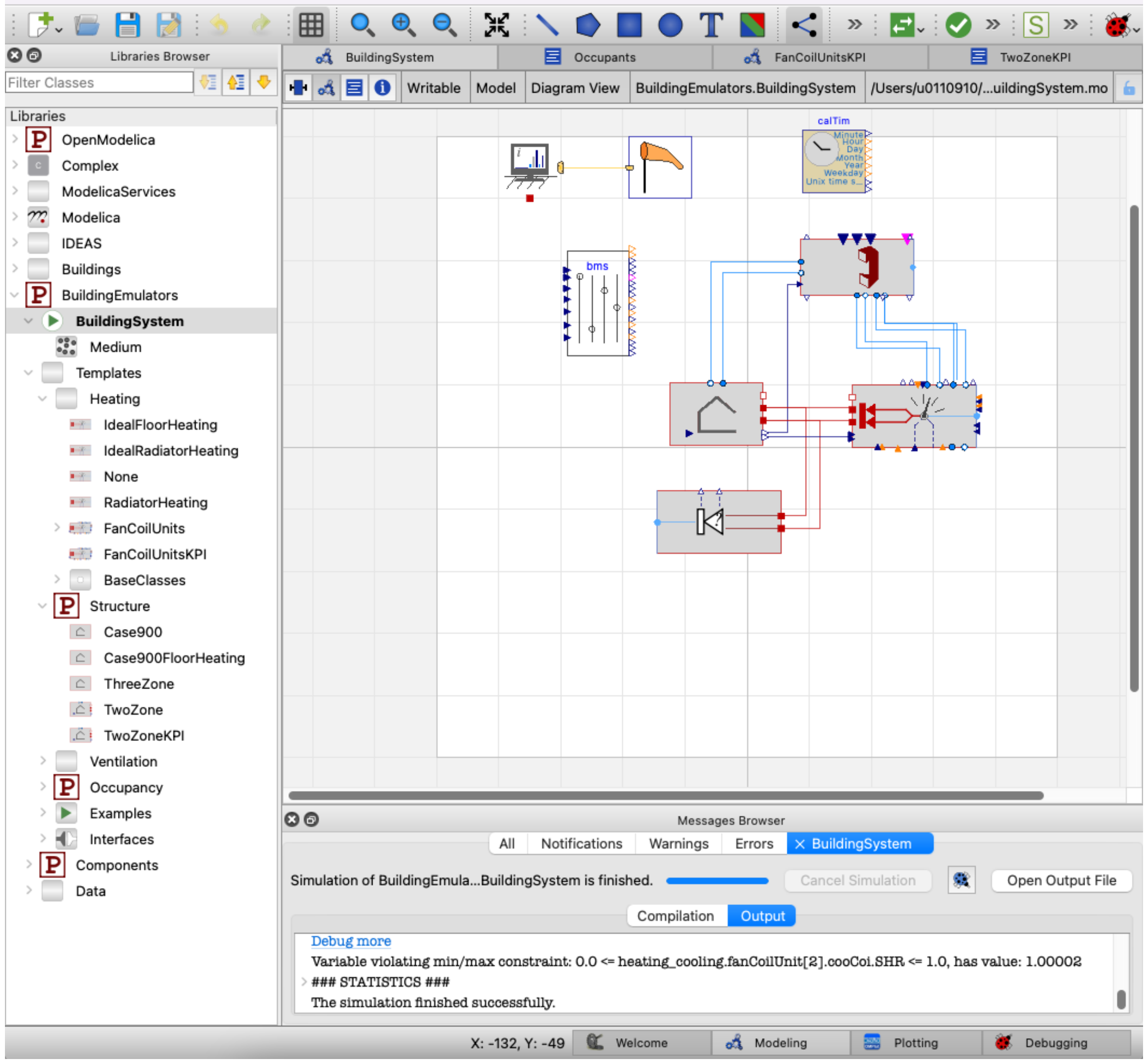
Has very similar features than Dymola, like those we have just seen



Variables	Value
(Acti...ystem	
bms	
calTim	
heating_cooling	
occupancy	
sim	
structure	
ATrans	2950.0
AZones	
C	5.08475
Q_design	
Q_de..._tot	101558
TSensor	
[1]	20.607...
[2]	21.1810...
T_start	23
VZones	
addPercHighNz	
addPercHighSz	
addPercLowNz	
addPercLowSz	
dTPerc	
fraWin	0.5
h	15
hFlo	3
heatPortCon[1]	
heatPortCon[2]	
heatPortRad[1]	
heatPortRad[2]	
humAir[1,1]	0.0054...
humAir[1,2]	0.994573
humAir[2,1]	0.0054...
humAir[2,2]	0.994573
l	40
m_fl...inal	
nEmb	0
nFlo	5
nOcc	

Simulation of BuildingEmulat....BuildingSystem is finished.

```
Debug more
Variable violating min/max constraint: 0.0 <= heating_cooling.fanCoilUnit[2].cooCoi.SHR <= 1.0, has value: 1.00002
### STATISTICS ###
The simulation finished successfully.
```



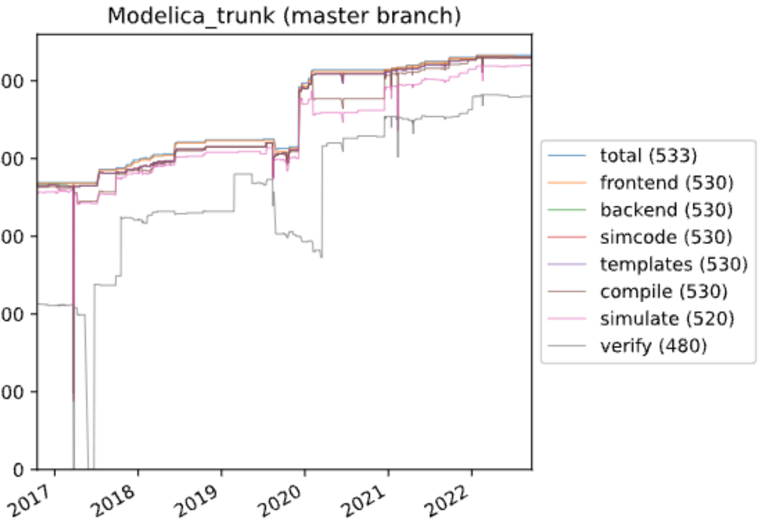
Simulation of BuildingEmula...BuildingSystem is finished.

```
Debug more
Variable violating min/max constraint: 0.0 <= heating_cooling.fanCoilUnit[2].cooCoi.SHR <= 1.0, has value: 1.00002
### STATISTICS ###
The simulation finished successfully.
```

OpenModelica

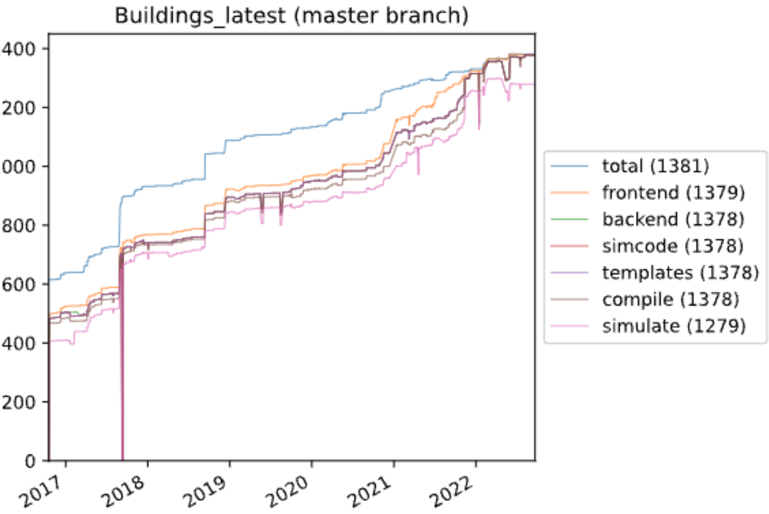
Historically less performant than Dymola, but has radically improved over the last years:

Modelica Standard Library (MSL) coverage



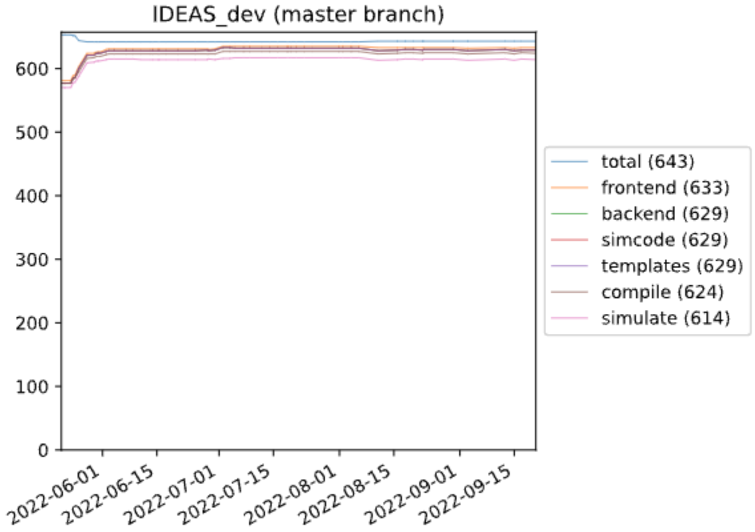
https://libraries.openmodelica.org/branches/history/master/Modelica_trunk.svg

Buildings library coverage



https://libraries.openmodelica.org/branches/history/master/Buildings_latest.svg

IDEAS coverage (recently added!)



https://libraries.openmodelica.org/branches/history/master/IDEAS_dev.svg

The only reason not to use it this year: still lacks a few graphical features. OM team is working on them

Useful links

General

- www.modelica.org
- www.openmodelica.org
- www.jmodelica.org DISCONTINUED
- http://www.claytex.com/tech-blog/

Modelica language

- http://book.xogeny.com/
- http://doc.modelica.org
- http://specification.modelica.org/

Libraries:

- IDEAS
  https://github.com/open-ideas
- Buildings
  https://simulationresearch.lbl.gov/modelica
  (look at Buildings.Examples.Tutorial)
- IBPSA Project 1
  https://github.com/ibpsa/modelica-ibpsa

Dymola user guide

- Online
- Via Dymola > help

Exercise 1

- See exercise sheet on Github
  https://github.com/open-ideas/__CrashCourse__/blob/master/Exercises/Exercise%201/Latex/Exercise1.pdf

Part 2: Modelling and simulating in Dymola

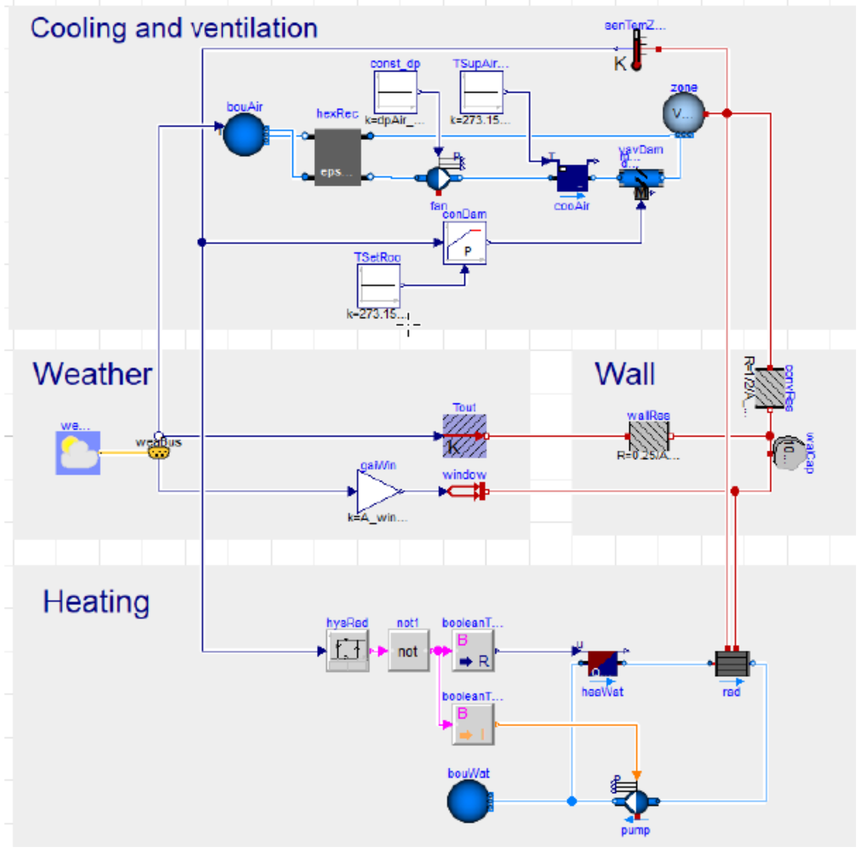
Jelger Jansen

Live demonstration

- Create package, create model
- Units
- Search, drag and drop subcomponents. Instantiate model convention
- Simulation tab and adapt parameters
- Connect components
- Propagate parameters
- Use check/translate in Dymola and debug:
    - Syntax error
    - Modeling error: singularity
    - Model with external input
- Saving and exporting results as .csv, .mat, .txt
- Simulation log/Statistics

Exercise 2: Simple house model

- See exercise sheet on Github
  https://github.com/open-ideas/__CrashCourse__/blob/master/Exercises/Exercise%202/Latex/Exercise2.pdf


Cooling and ventilation
senTemZ...
K
const_dp
k=dpAir_...
TSupAir...
k=273.15...
zone
V...
bouAir
hexRec
eps...
fan
cooAir
vavDam
conDam
P
TSetRoo
k=273.15...
Weather
we...
weaBus
Tout
K
gaiWin
k=A_win...
window
Wall
walRes
R=0.25/A...
R=1/2/A...
convRes
walCap
Heating
hysRad
not1
not
booleanT...
B
R
booleanT...
B
I
heaWat
rad
bouWat
pump