Modeling, Simulation and Analysis of Integrated Building Energy and Control Systems

Michael Wetter

Simulation Research Group

Building Technologies Department

Energy and Environmental Technologies Division

Lawrence Berkeley National Laboratory

October 2010

Overview

- Introduction

  Trends - Problems - Needs

- Mono-Simulation with Modelica

  Modelica Standard Library - LBNL Buildings Library - Applications

- Co-Simulation with Building Controls Virtual Test Bed

  Building Controls Virtual Test Bed - Applications

- R&D Needs

Trends in Building Systems

Building systems become more complex

- Interaction: Continuous time, discrete time, events.
- States: Multiple operating modes.
- Controls integration: Floating setpoints, electrical grid, active façade.
- Thermal integration: Geothermal systems, heat storage and recovery.

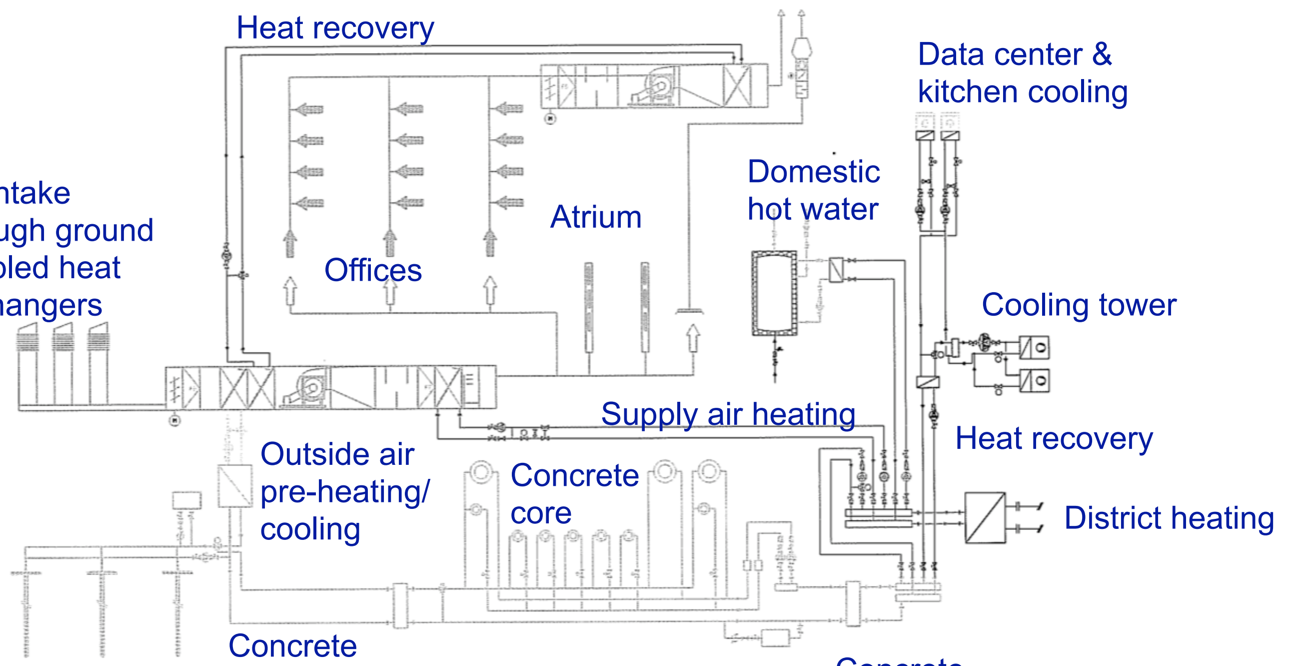


Figure adapted from: Oehler, Grosse Passivhaeuser, Kohlhammer, 2004

	Number of modes
Cooling “generation”	4
Heating “generation”	4
Space cooling	3
Space heating	3
Ventilation	2

Multi-scale, multi-physics systems

Building systems are multi-scale, multi-physics systems

- Flow friction in piping and duct networks
- Thermodynamics of moist air and refrigerants
- Heat and mass transfer through solids (walls, windows)
- Airflow inside and outside of building
- Distribution of direct and diffuse radiation
  - daylight
  - short-wave radiation
  - long-wave radiation
- Evolution (physics & controls)
  - continuous time
  - discrete time
  - event-driven

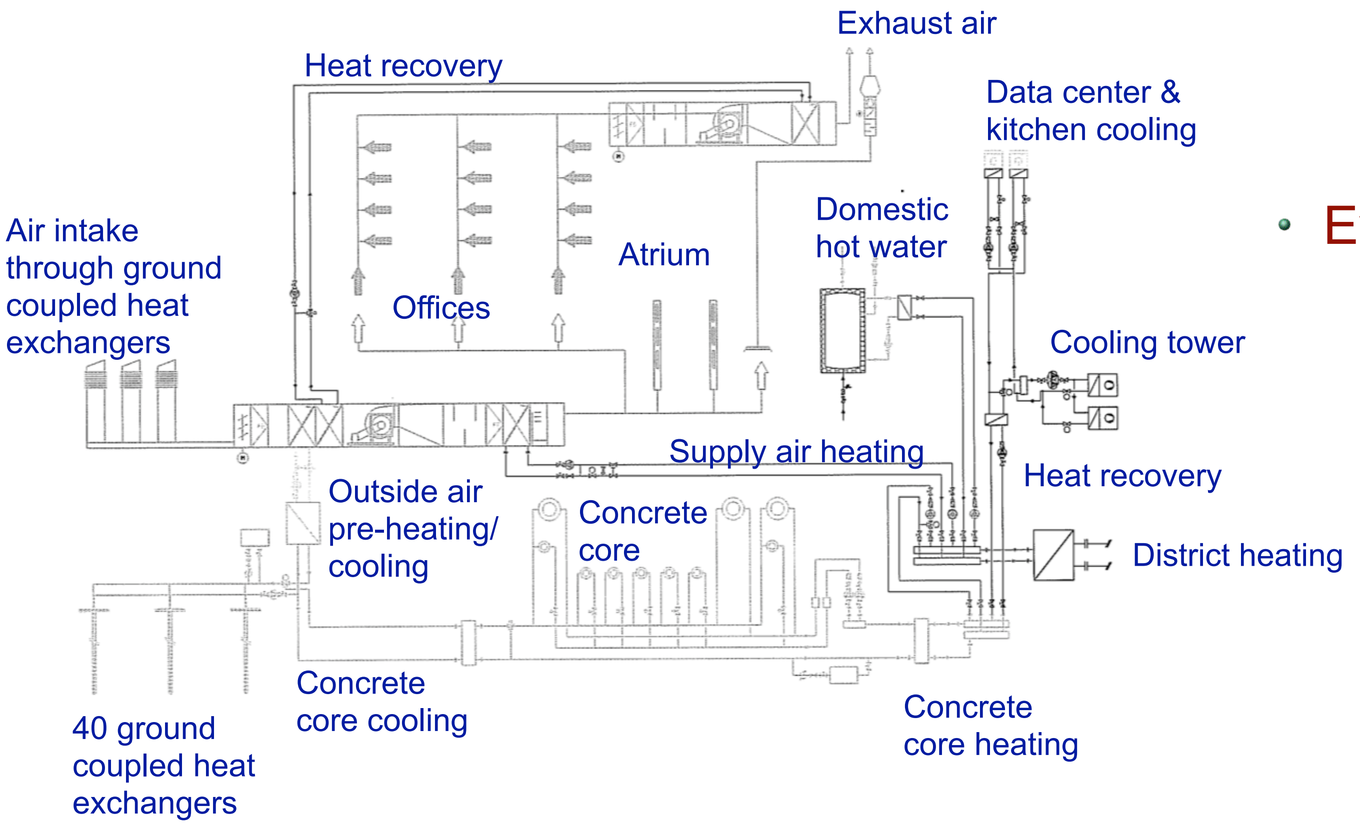
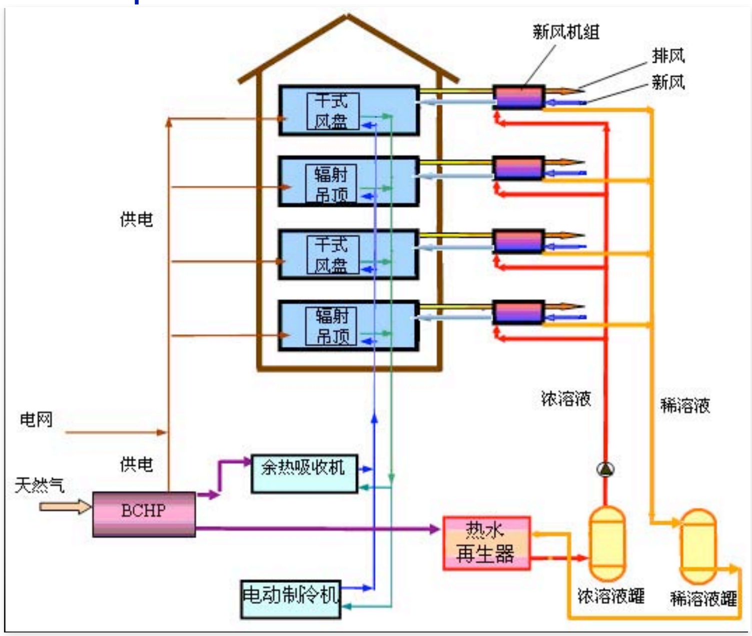


Figure adapted from: Oehler, Grosse Passivhaeuser, Kohlhammer, 2004

How do we accelerate RD&D of next generation technologies?

Decentralized dehumidification with liquid desiccant



Provide user-extensible tools for rapid prototyping, model-based design and controls, built on open-source standards.

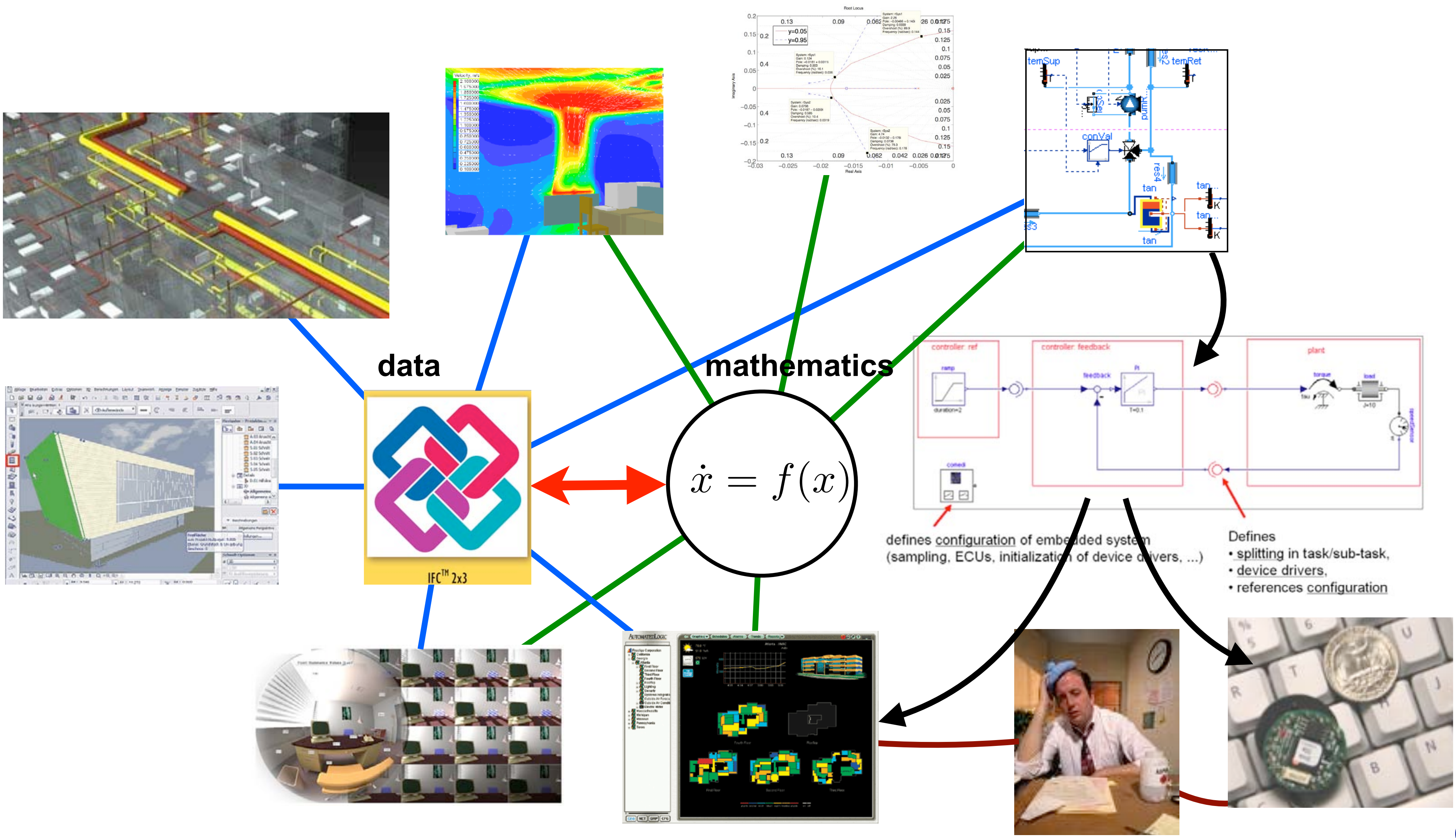
Phase change material to increase thermal storage

Cyprus grass to humidify supply air

Web-server at the size of 25 cents

R&D Needs for Whole-Building Model-Based Design and Operation

What modeling framework enables model-based design and operation?



Challenges -- Controls

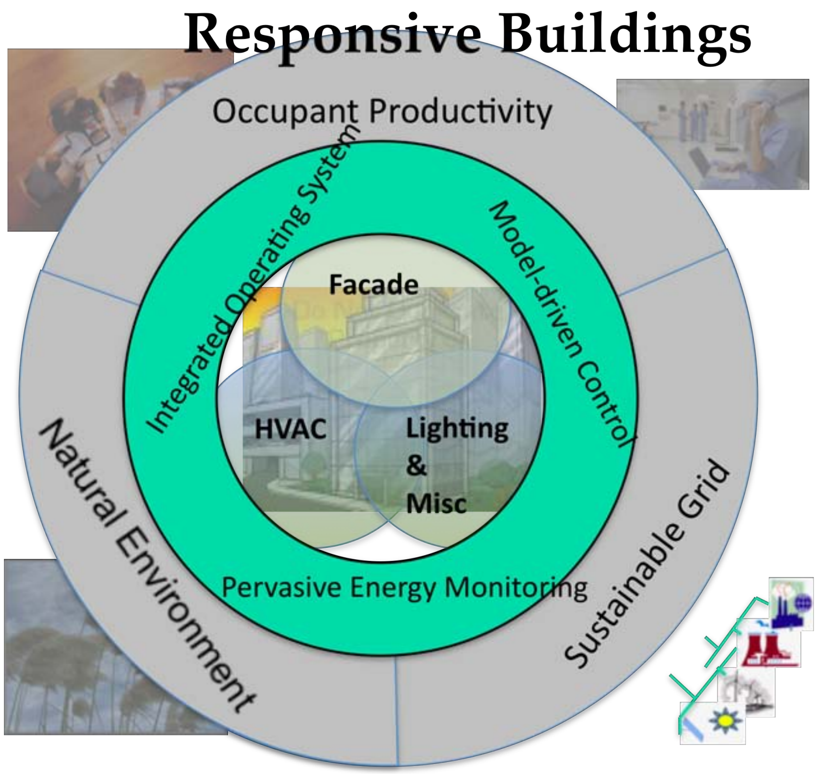
Integration of subsystems (grid, HVAC, active facade, lighting).

Enable optimal
- energy storage
- equipment scheduling
- setpoint scheduling
- grid interaction.

Making inefficiencies visible to end-user.

Detect and diagnose faults at the system-level.

Close disconnect between design and operation, using executable specification, to
- increase productivity.
- reuse design across platforms.
- enforce accountability across hand-off points.
- reduce human error.



Current State -- Building Simulation to Support Controls Design and Operation

Today's building simulation programs (EnergyPlus, DOE-2):

These tools have not been designed to support controls.

Large simulators (500,000 lines of code).

Dynamic models
Heat transfer (opaque constructions & room air).
Energy storage in water tanks.

Steady-state models
HVAC components and system (written for 10 minutes to 1 hour time step).
Controls.

Controls semantics
"Ideal controller" that meets load (cooling power etc.) by computing how much mass flow and what temperature a component needs to deliver, subject to capacity constraints.
No notion of continuous time systems or hybrid systems.

Solution method
10 to 20 nested solvers, no global error control, many components integrate their own solver, which introduces large discontinuities in approximate transfer function.

Challenges -- Building Simulation to Support Controls Design and Operation

Express models in a modeling language, not a programming language!

Modular objects with standard interfaces (thermodynamics, controls, ...)

Hierarchies to manage complexity vertically and horizontally

Code generation for control hardware

Different evolutions within modules

- continuous time (seconds to minute time scale)
- discrete time
- finite state machine

Model feedback control of measured states (temperature, pressure), not heating/cooling load. (This requires models and solvers that are different from today's standard building simulation.)

Analysis support through application programming interfaces (API)

- Building Information Models
- link to domain-specific tools (such as MATLAB/Simulink)
- reduced order model extraction
- optimization

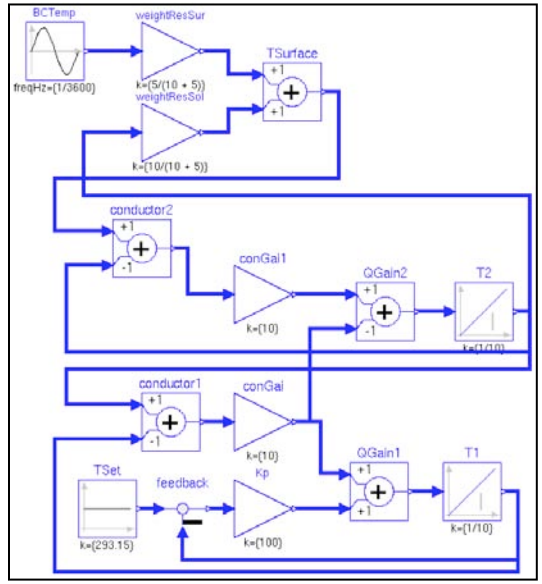
Increase Level of Abstraction

Higher-level of abstraction to
- increase productivity
- facilitate model-reuse
- preserve system topology
- enable analysis
- generate code for target hardware

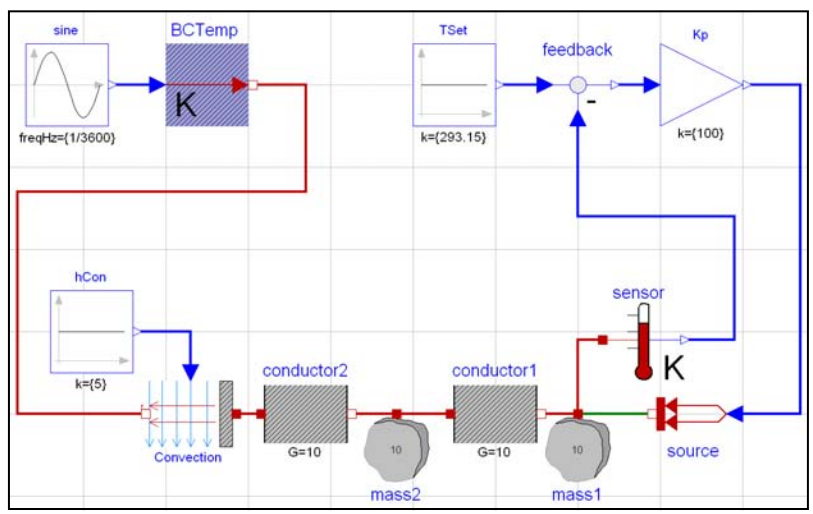
Procedural modeling ≈ 1970

```
program euler
implicit none
double precision, parameter :: tFin = 7200   ! Final time
integer, parameter :: N = 72000              ! Number of steps
integer, parameter :: NCom = 10              ! Communication interval
integer :: iCom                              ! Communication counter

double precision dT                          ! Time step

double precision, parameter :: T0 = 293.15   ! Initial temp.
double precision :: T1, T2                   ! Temperature
double precision :: TBC                      ! Temp. boundary condition
double precision :: TSur                     ! Surface temperature
double precision, parameter :: TSet = 293.15 ! Set point temp.
double precision :: derT1, derT2             ! Temperature derivative

double precision :: QSou, QCon1, QCon2       ! Heat flux

double precision, parameter :: Kp = 100      ! P Gain
double precision, parameter :: h = 5         ! Convective heat transfer coefficient
double precision, parameter :: G = 10        ! Conductivity
double precision, parameter :: C = 10        ! Capacity


integer :: i                                 ! Loop counter
integer, parameter :: lun = 6                ! logical unit number

double precision, parameter :: amp = 5       ! Amplitude
double precision :: time                     ! Simulation time

character(LEN=*), parameter  :: FMT = "(4F14.7)"

! Initialize variables
dt = tFin/N
T1 = T0
T2 = T0
time = 0
iCom = 1
open (lun, FILE='results.txt')
! Perform integration
do i = 1, N, 1
   TBC = T0 + amp * dsin(2*3.14159*time/3600)
   TSur = T2 * ( G / (G+h) ) + TBC * ( h / (G+h) )
   QSou = Kp * (TSet - T1)
   QCon1 = G * (T2 - T1)
   QCon2 = G * (TSur - T2)
   derT1 = 1/C * (  QCon1 + QSou )
   derT2 = 1/C * ( -QCon1 + QCon2)
   if (i.EQ.iCom) then
      write(lun,FMT) time, T1, T2, QSou
      iCom = iCom + NCom
   endif
! Update variables
   T1 = T1 + dt * derT1
   T2 = T2 + dt * derT2
   time = time + dt
end do
write(lun,FMT) time, T1, T2, QSou
close(lun)
write(*,*) 'Program finished'
end program
```

Block diagram modeling ≈ 1990

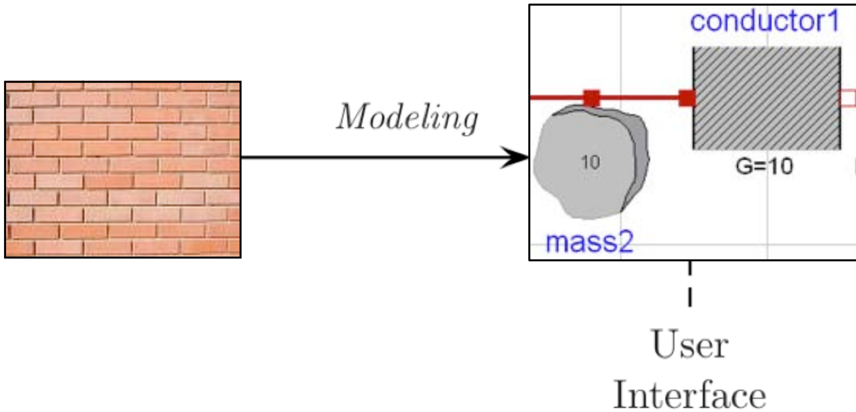


Equation-based, object-oriented modeling ≈ 2000



Separation of Concerns

Modeling



Compilation & Simulation

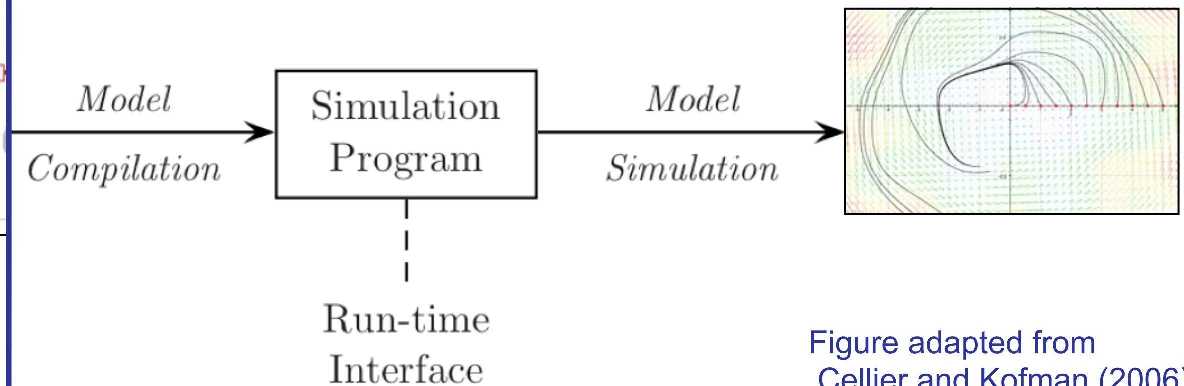


Figure adapted from
Cellier and Kofman (2006)

Describes the phenomena

- Standardized interfaces
- Acausal models
- Across & through variables
- Hierarchical modeling
- Class inheritance

Solves the equations

- Partitioning
- Tearing
- Inline integration
- Adaptive solver
  - Integration
  - Nonlinear equations

Overview

- Introduction

  Trends - Problems - Needs

- Mono-Simulation with Modelica

  Modelica Standard Library - LBNL Buildings Library - Applications

- Co-Simulation with Building Controls Virtual Test Bed

  Building Controls Virtual Test Bed - Applications

- R&D Needs

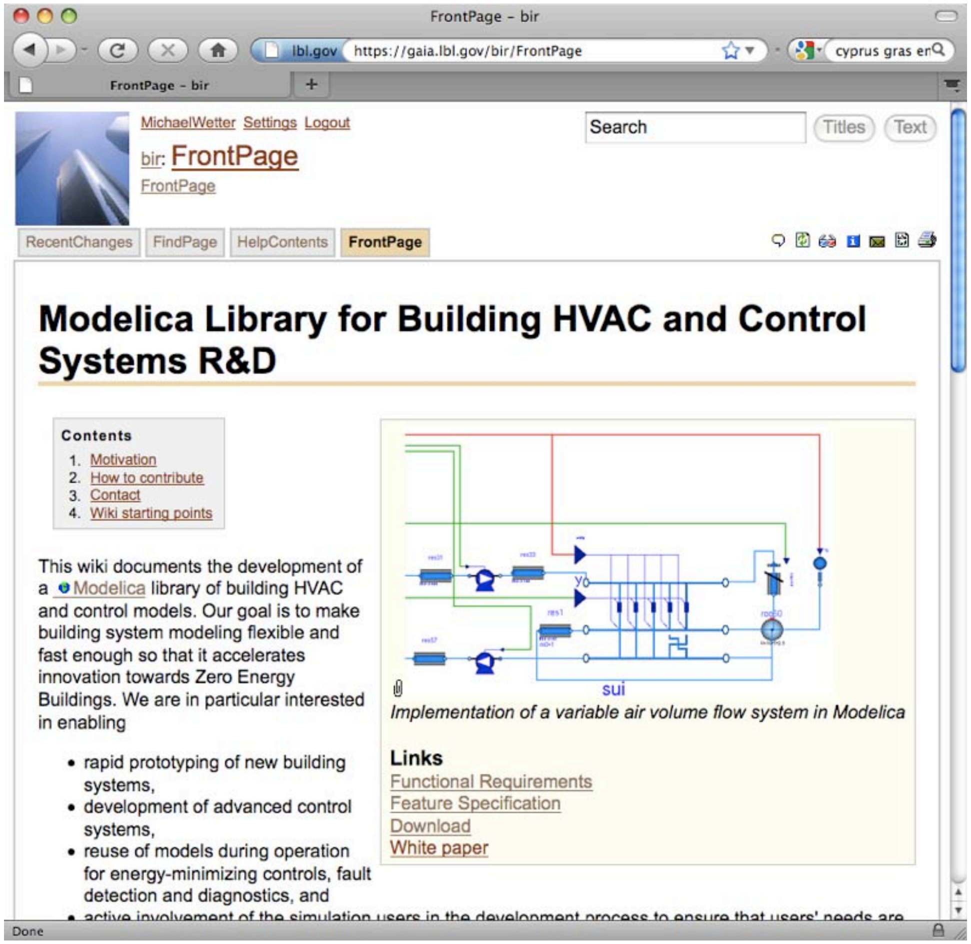
Modelica Buildings Library

Enable

- Rapid prototyping of innovative systems
- Controls design
- Model-based operation

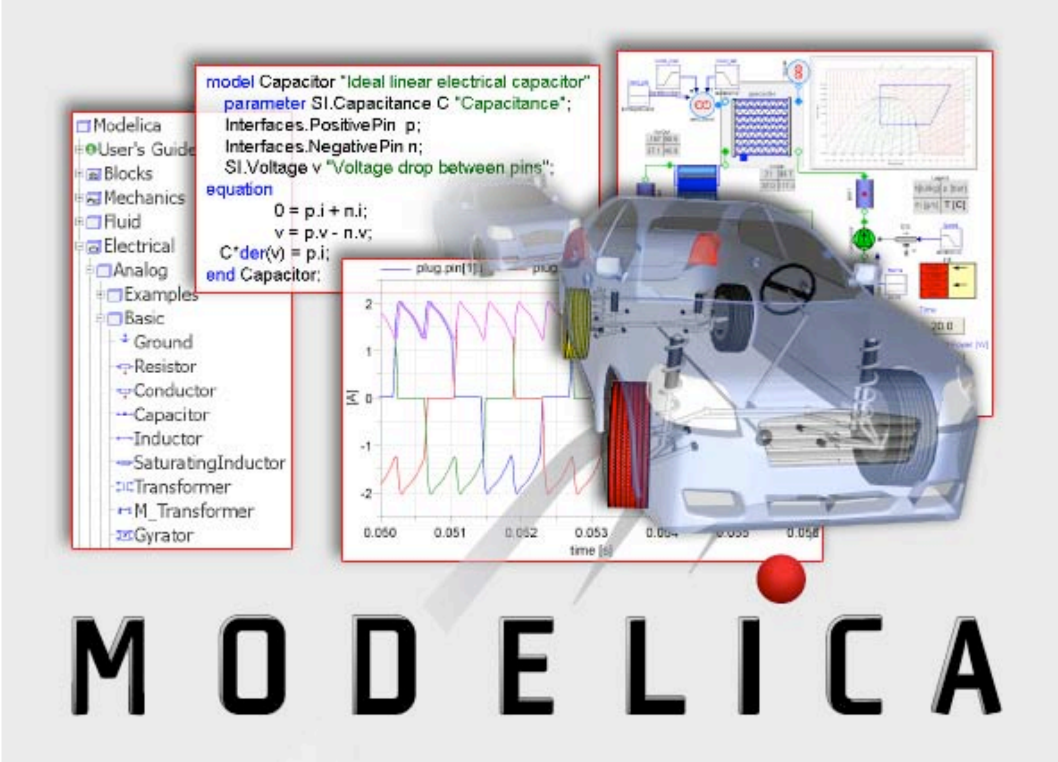
Available from

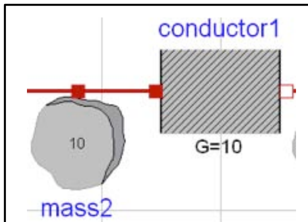
http://simulationresearch.lbl.gov/modelica

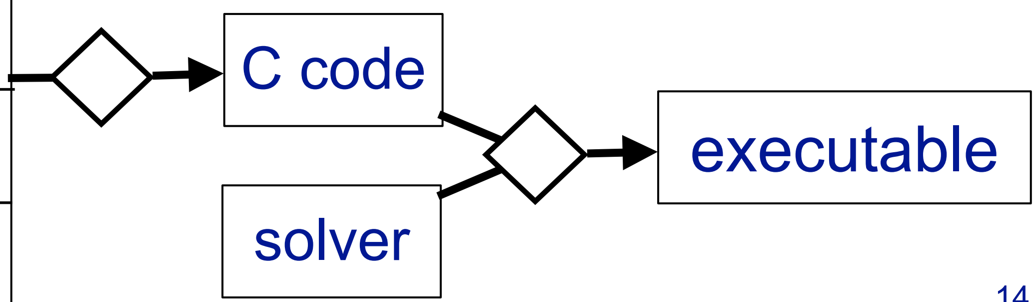


What is Modelica?

- Object-oriented equation-based modeling language
- Open standard
- Developed since 1996 because conventional approach for modeling was inadequate for integrated engineered systems
- Well positioned to become de-facto open standard for modeling multi-engineering systems
  - ITEA2: 370 person years investment over next three years.



	Graphical modeling - input/output free - block-diagram - state machines - bond-graphs
`a:=2;` `b:=2*a;`	Algorithmic code
`C*der(T) = Q_flow;` `0 = T - TBoundary;`	Acausal equations



Modeling of Components with Dissimilar Mathematics

Schematic diagram couples PDE, ODE, algebraic equations, state graph and discrete time systems.

differential equation

```
T = port.T;
C*der(T)=port.Q_flow
```

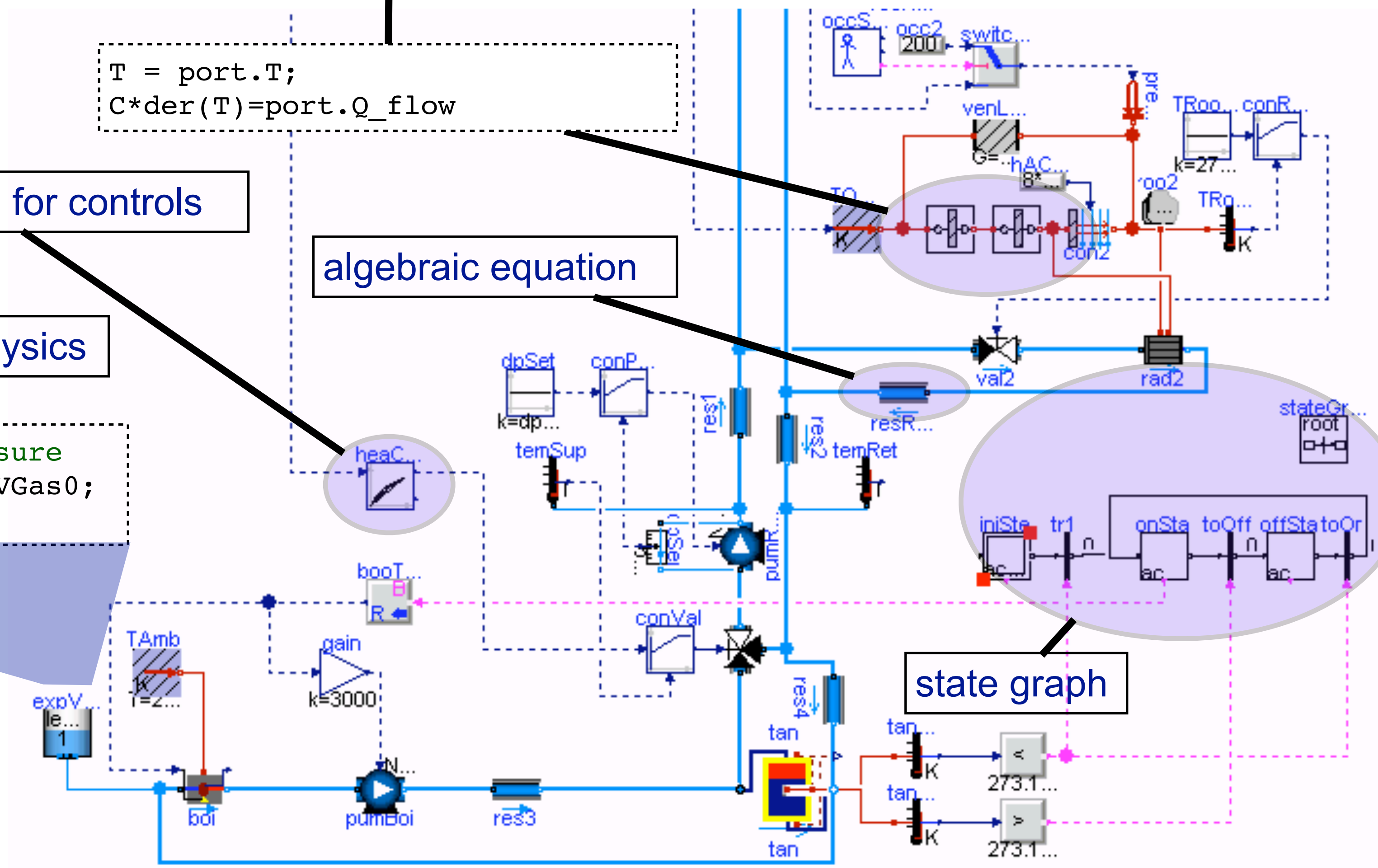
algorithmic code for controls

algebraic equation

acausal equations for physics

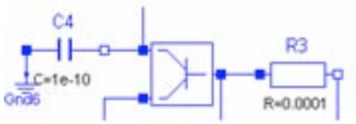
```
// Water content and pressure
port_a.p * (VGas) = p0 * VGas0;
medium.p = port.p;
```

state graph

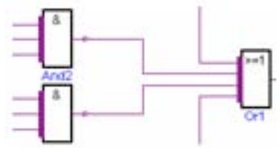


Modelica Standard Library. 1300 models & functions.

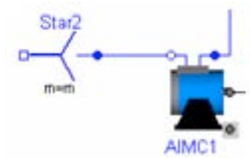
Analog



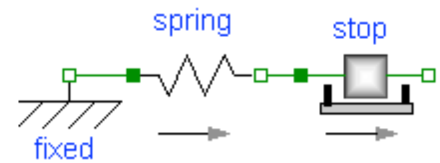
Digital



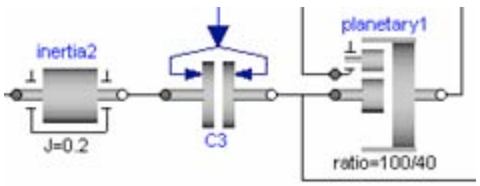
Machines



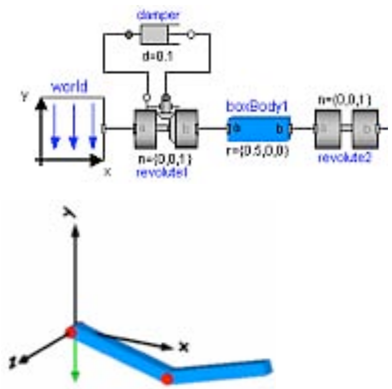
Translational



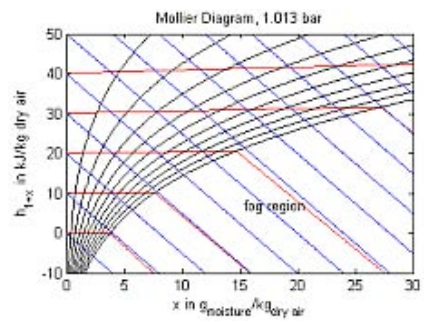
Rotational



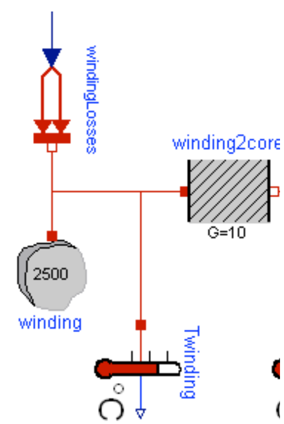
MultiBody



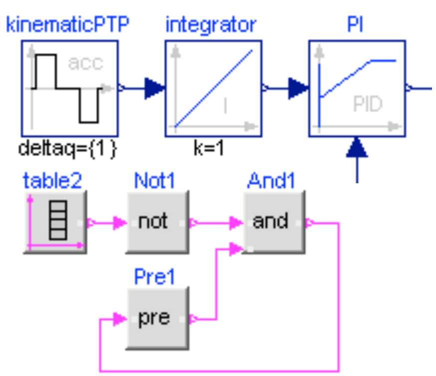
Media



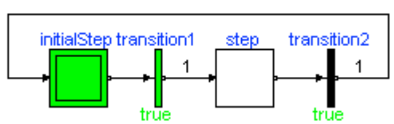
HeatTransfer



Blocks



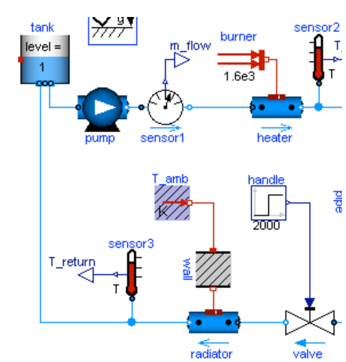
StateGraph



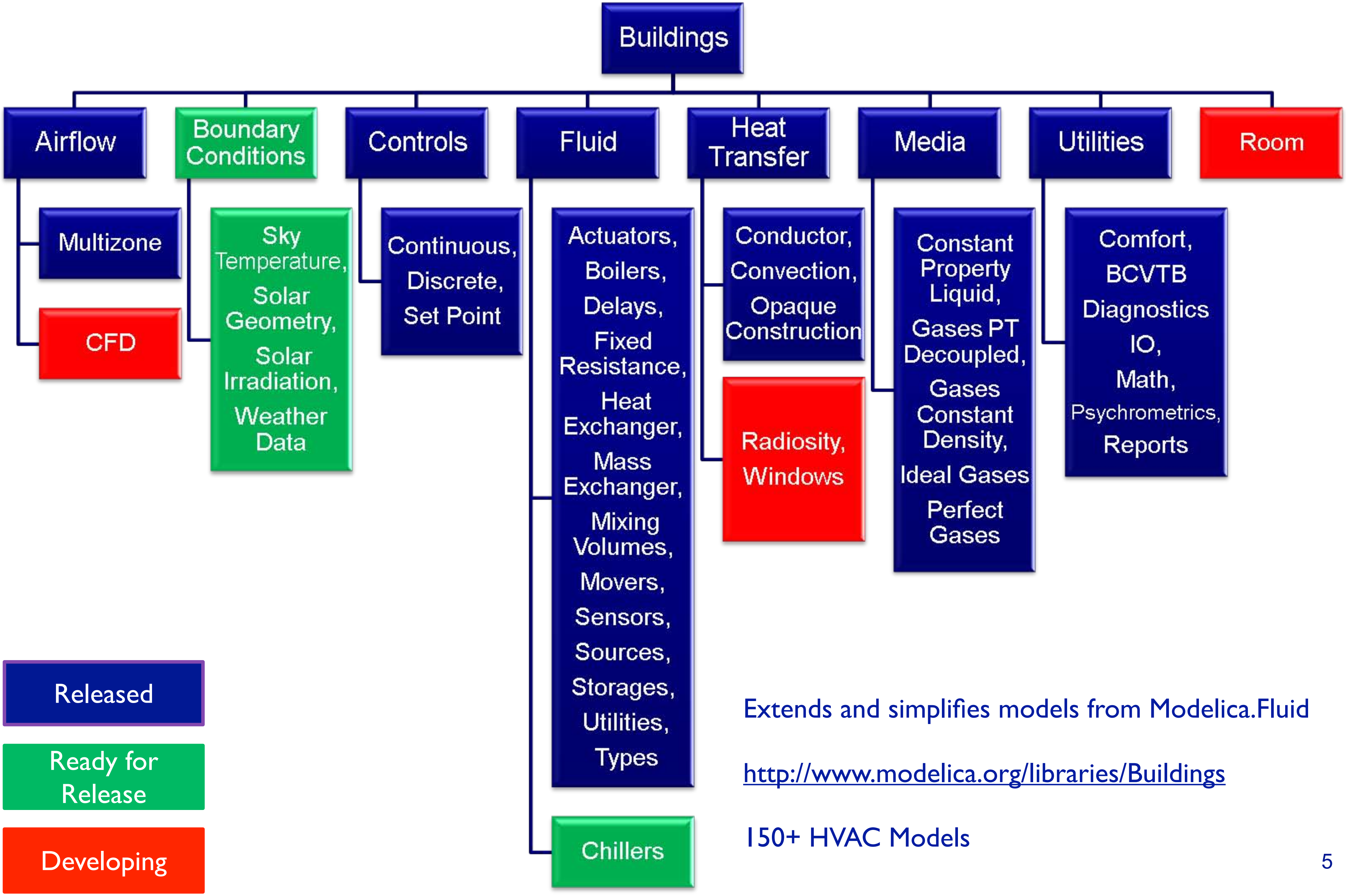
Math

$$\|v\|_p = \left(\sum_{i=1}^{n} |v[i]|^p\right)^{1/p}, \quad 1 \le p \le \infty$$

Fluid



Modelica Buildings Library



Extends and simplifies models from Modelica.Fluid

http://www.modelica.org/libraries/Buildings

150+ HVAC Models

Current Applications

Advanced Control
- Dynamic evaluation of control sequences

- Tool chain for model-based design and operation, including automatic code generation from models (proposed for FY11)

- Reusability of models from different domains

Integrated Building System
- Rapid prototyping of new energy systems

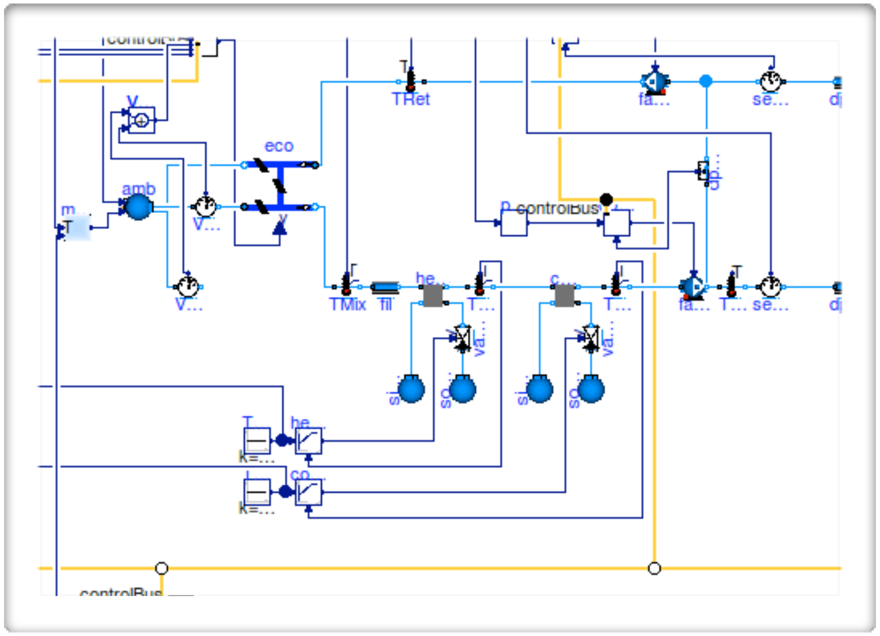
- Coupled simulation of HVAC, control and energy.
E.g. EnergyPlus + Modelica

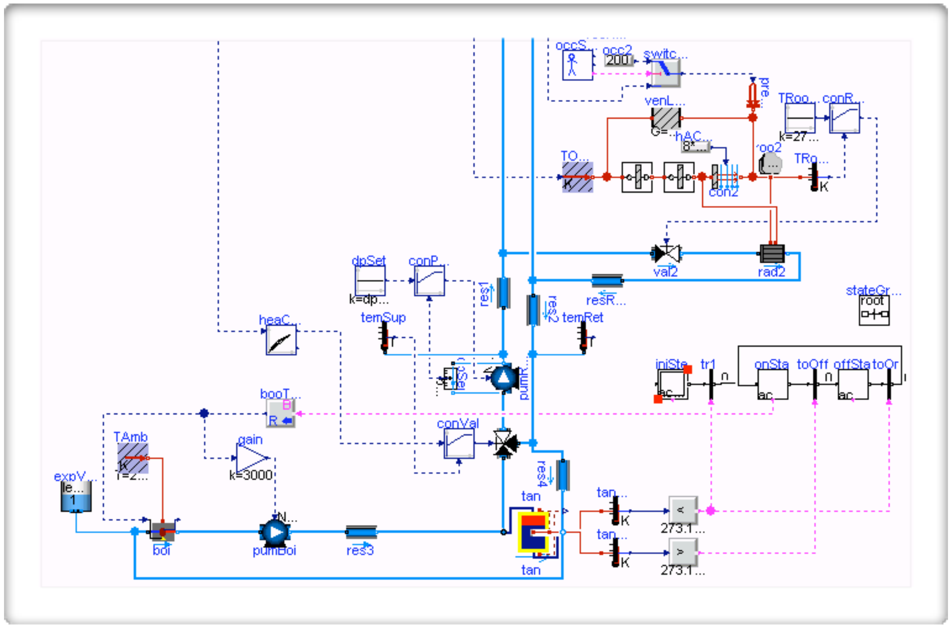
Education
Virtual Building Lab.
E.g. LearnHVAC

Modelica Buildings Library

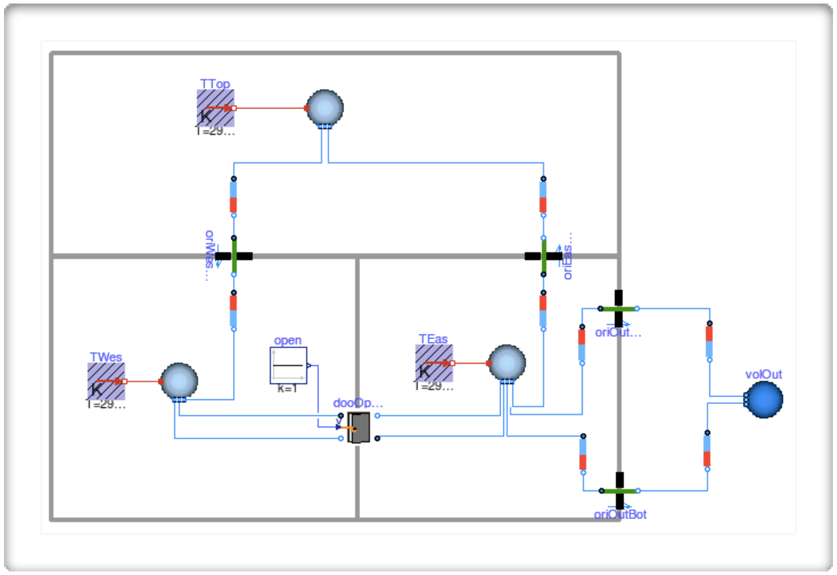
LBNL Buildings Library. 150+ models and functions.

Provides 150+ HVAC specific models based on “Modelica.Fluids” library.
http://www.modelica.org/libraries/Buildings





The biggest barrier to adoption is the lack of robustness of numerical solvers for these DAE systems with continuous and discrete state variables.



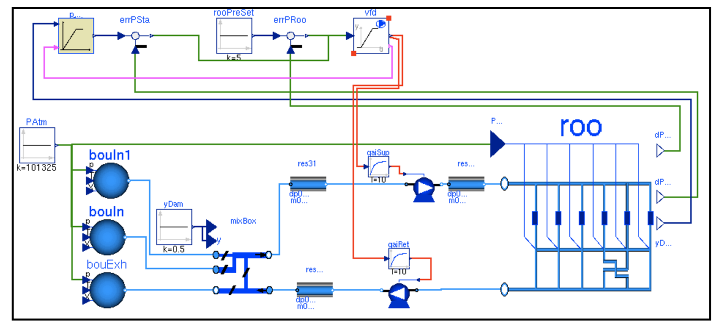
Applications

1) Rapid prototyping

Analyzed novel hydronic heating system with radiator pumps and hierarchical system controls.

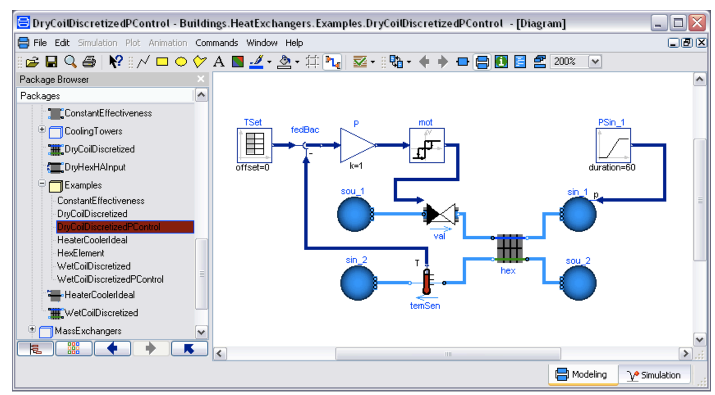
2) Supervisory controls

Simulated & auto-tuned “trim and response” sequence for variable air volume flow systems.

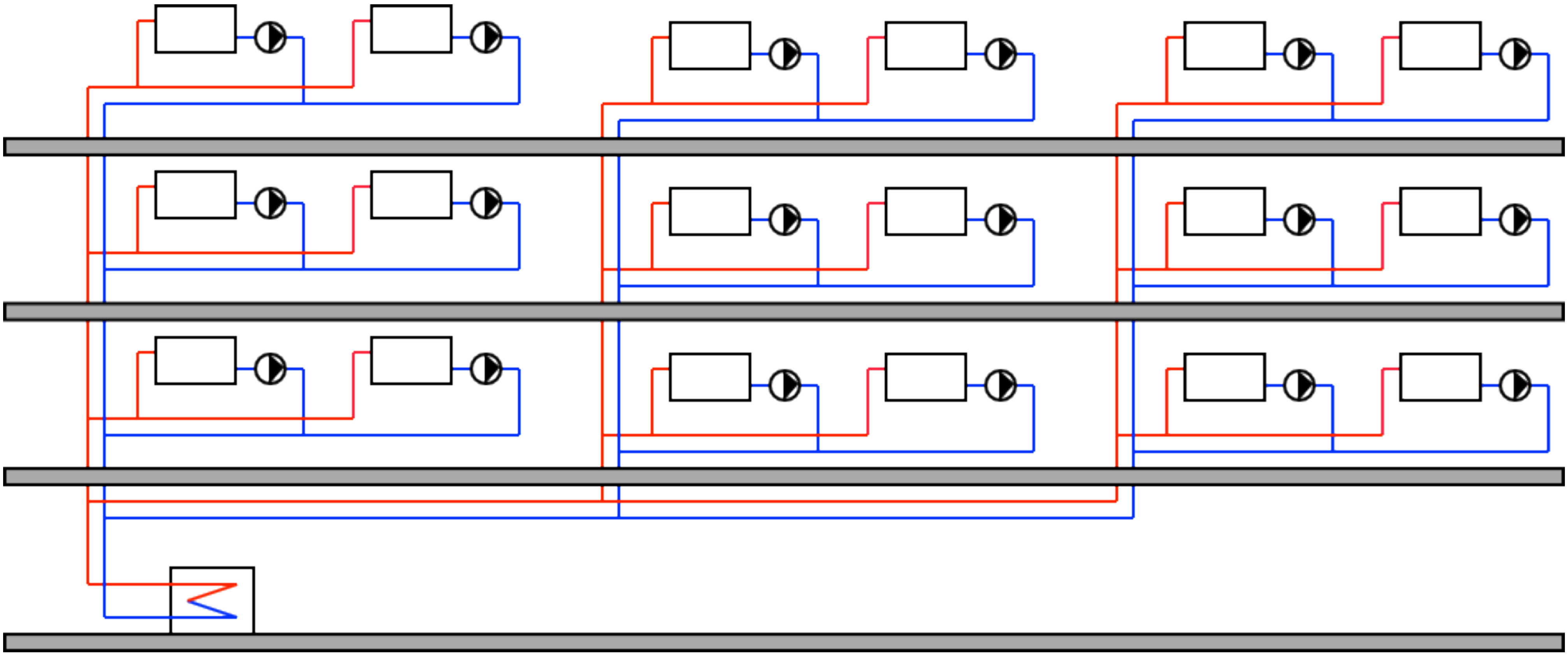


3) Local loop controls

Reused high-order model for controls design in frequency domain.



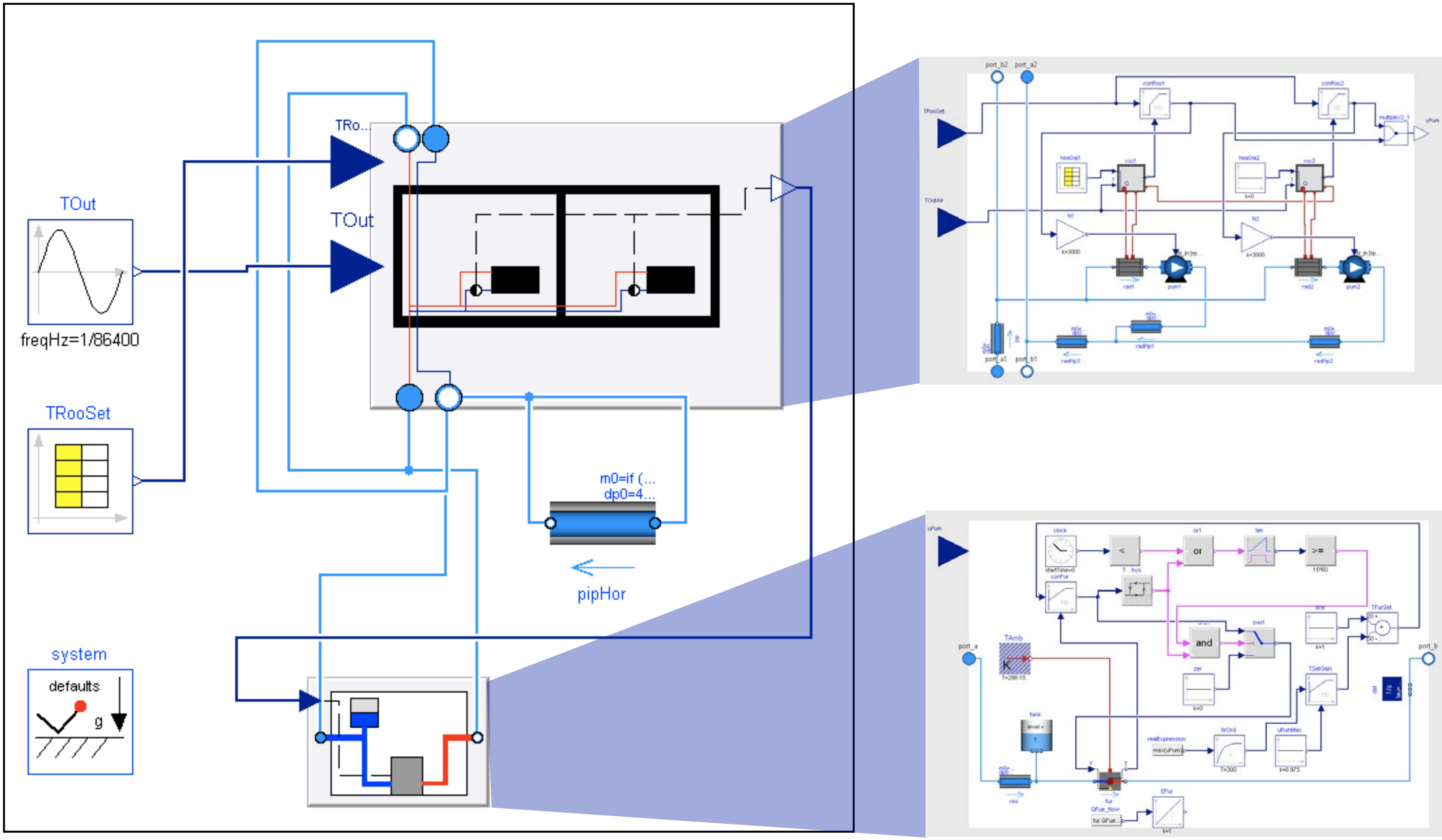
Rapid Prototyping: Wilo GENIAX



Original system model
2400 components
13,200 equations

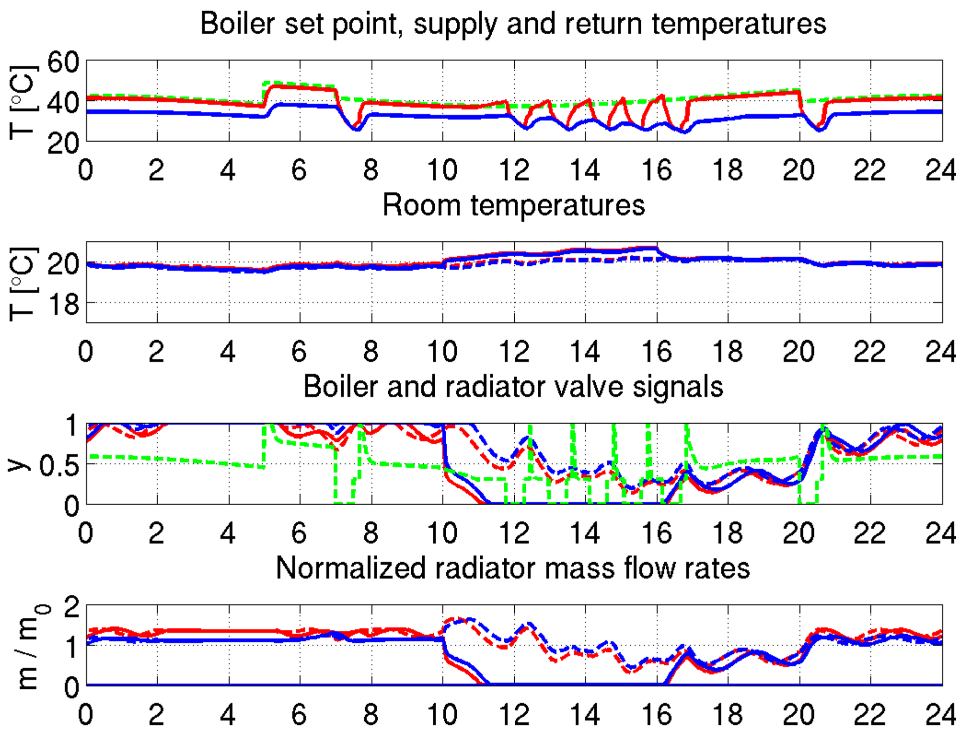
After symbolic manipulations
300 state variables
8,700 equations

Rapid Prototyping: Wilo GENIAX

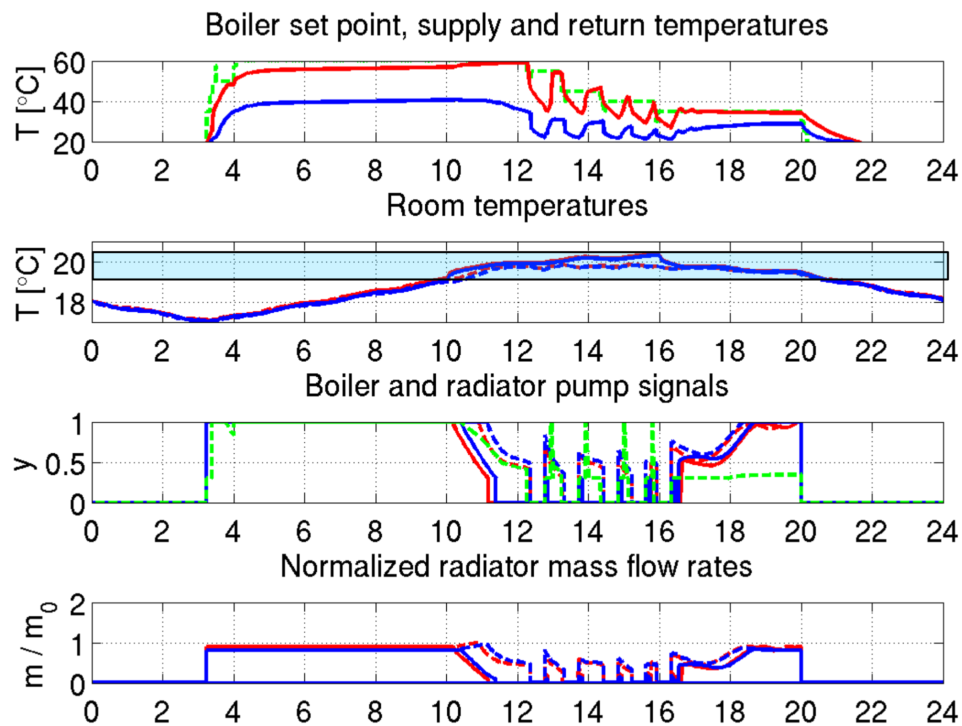


Rapid Prototyping: Wilo GENIAX

Thermostatic radiator valves



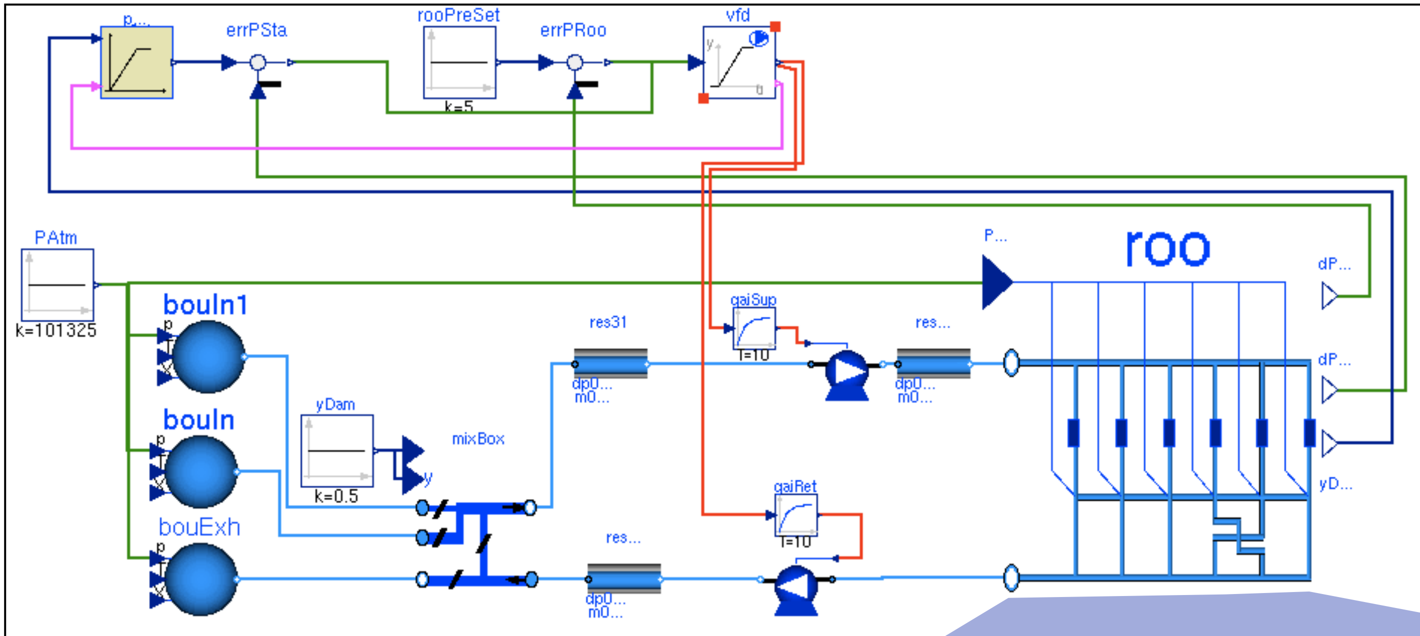
Radiator pumps



Reproduced trends published by Wilo.

Developed boiler model, radiator model, simple room model and both system models in one week.

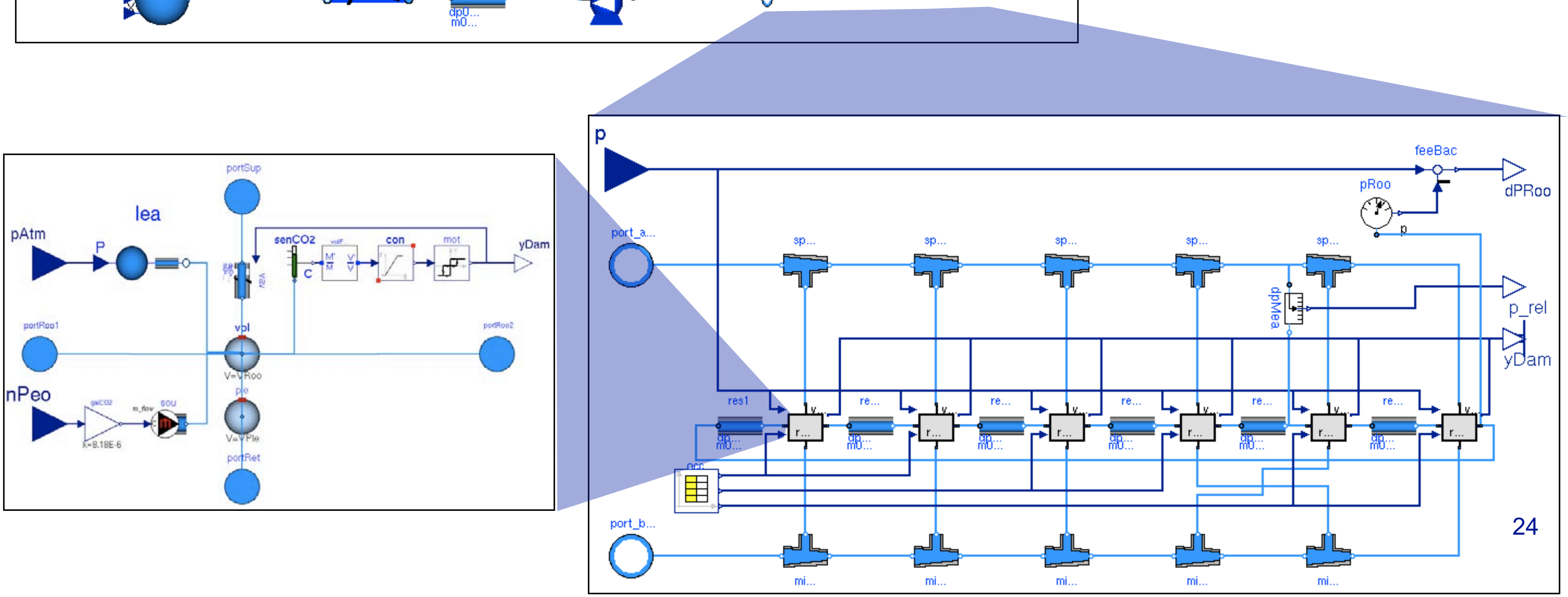
Applications – VAV System Controls



VAV System
(ASHRAE 825-RP)

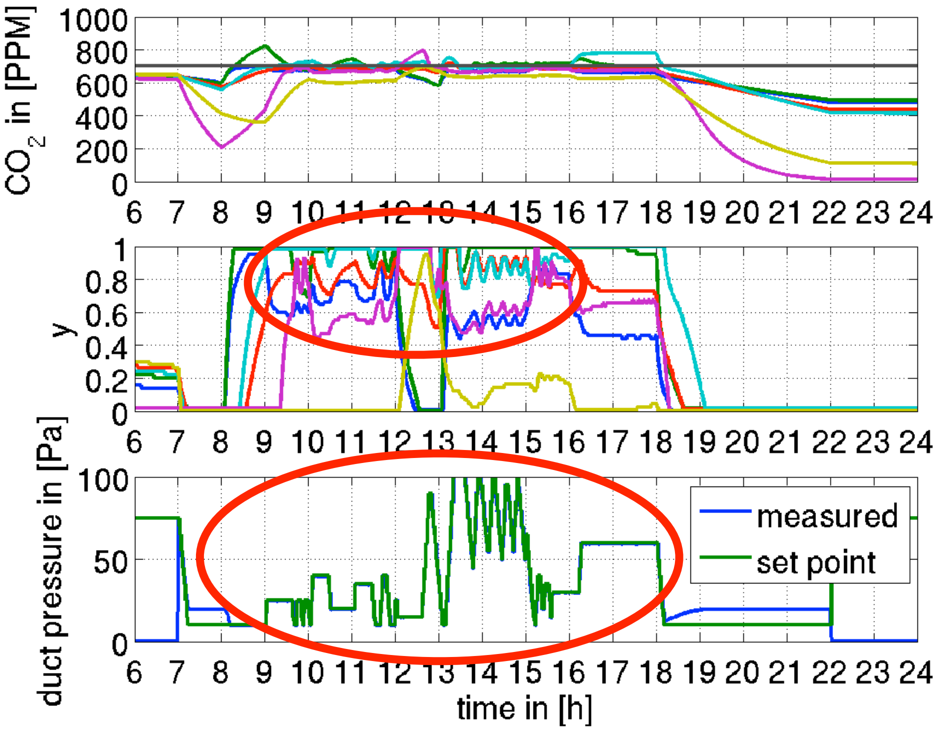
Trim & response control for fan static pressure reset
(Taylor, 2007)

Original system model
730 components
4,420 equations
40 state variables



Applications – VAV System Controls

Stabilized control and reduced energy by solving optimization problem with state constraints

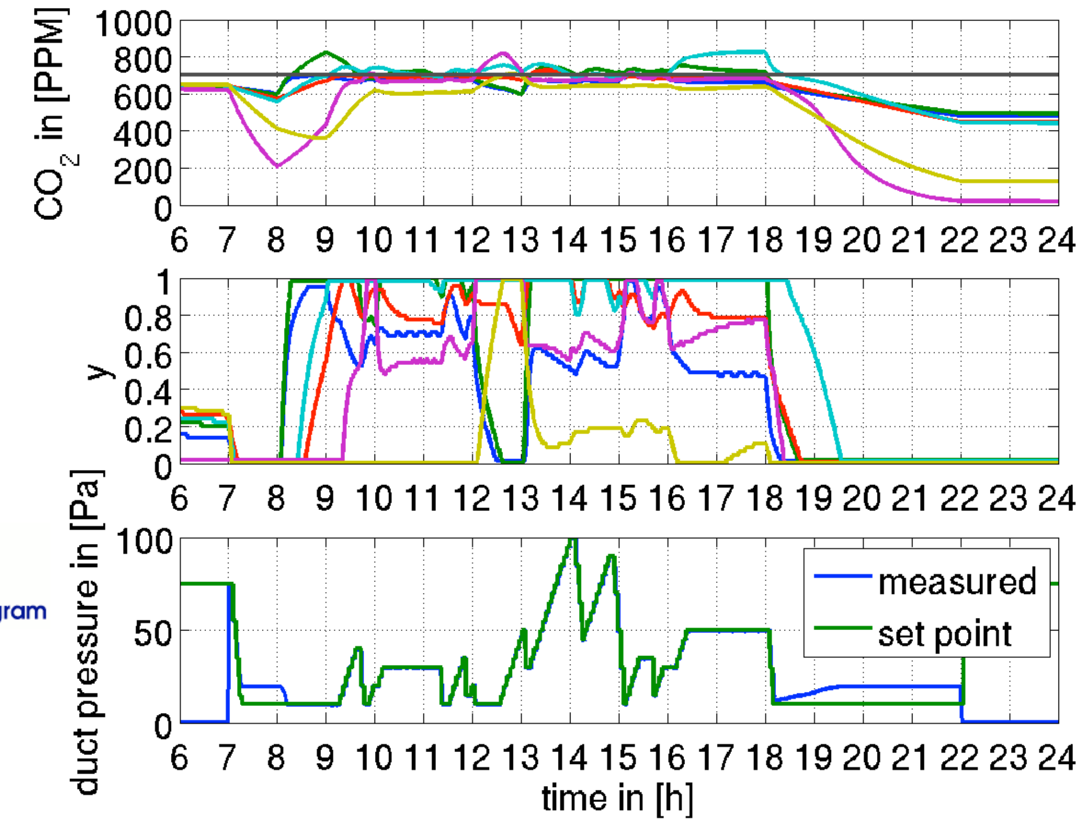


$$\min_{x \in \mathbf{X}} \{ f(x) \mid g(x) = 0 \},$$

$$f(x) = \frac{1}{E_0} \int_0^T P_f(x,t)\, dt,$$

$$g(x) = \frac{1}{T} \int_0^T \big(\max\{0, \ (y_j(x,t)/\hat{x}_s) - 1/(2\,K_p) - 1 \mid j \in \mathbf{J}(x,t)\} \big)^2 \, dt$$

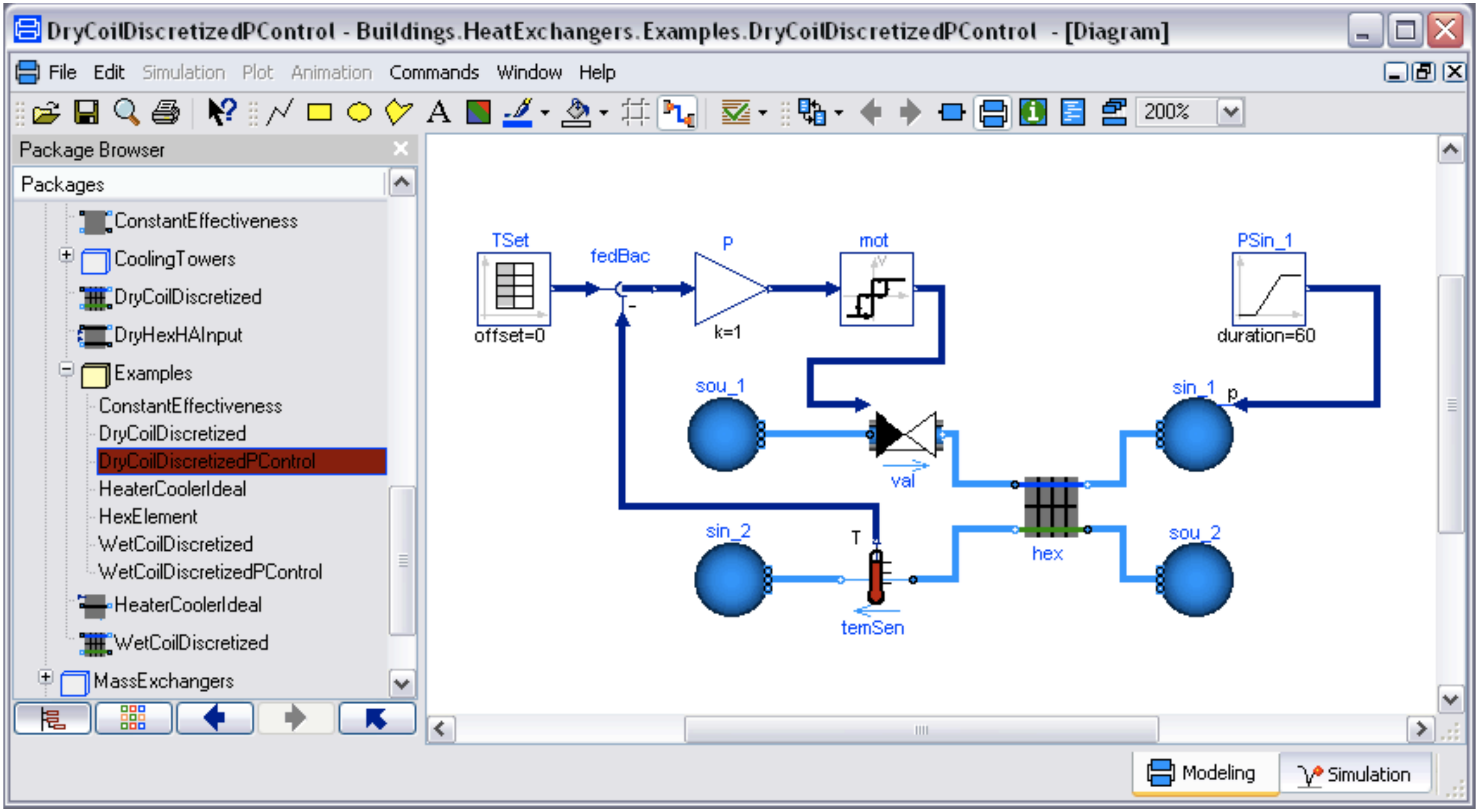
GenOpt
Generic Optimization Program



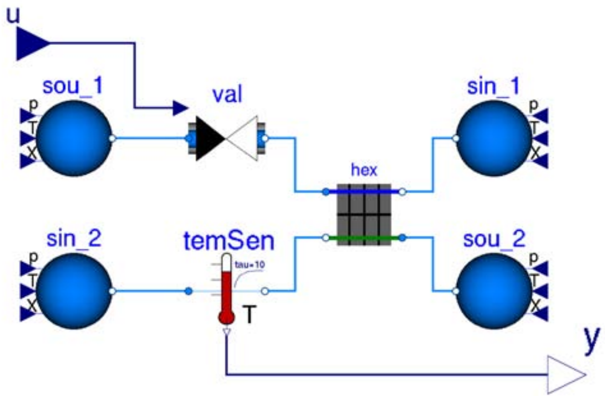
Applications – Feedback Loop Stability

Heat exchanger feedback control

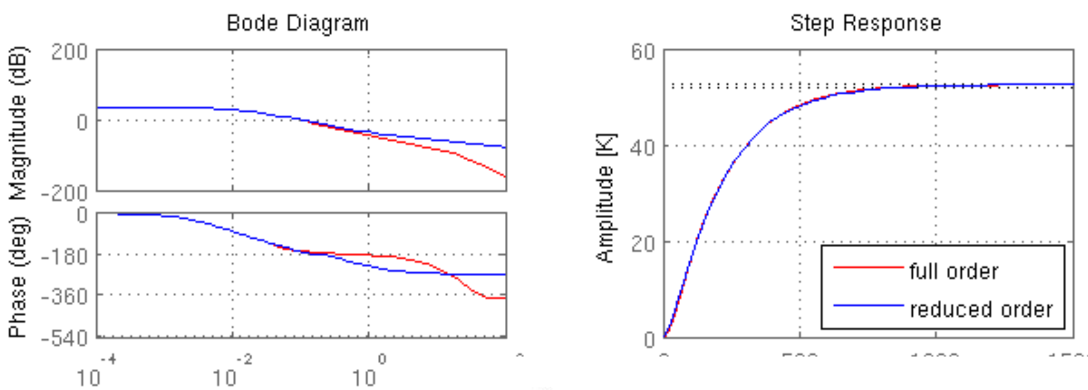
2632 equations
40 states
37x37 (linear) + 6x6 (nonlinear) → 0 (linear) + 2x2 (nonlinear)



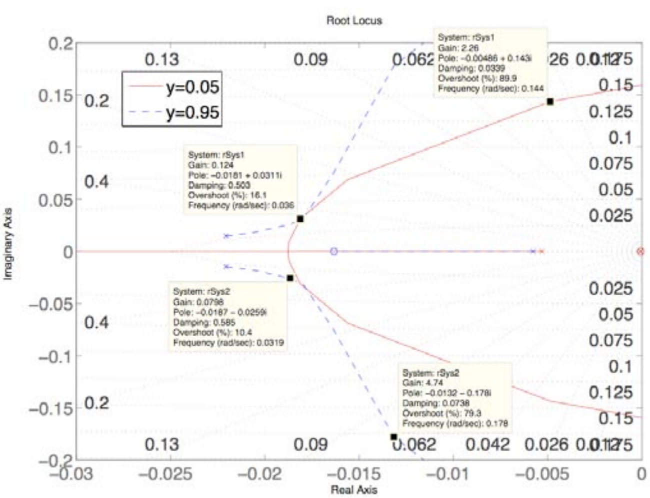
Applications – Feedback Loop Stability



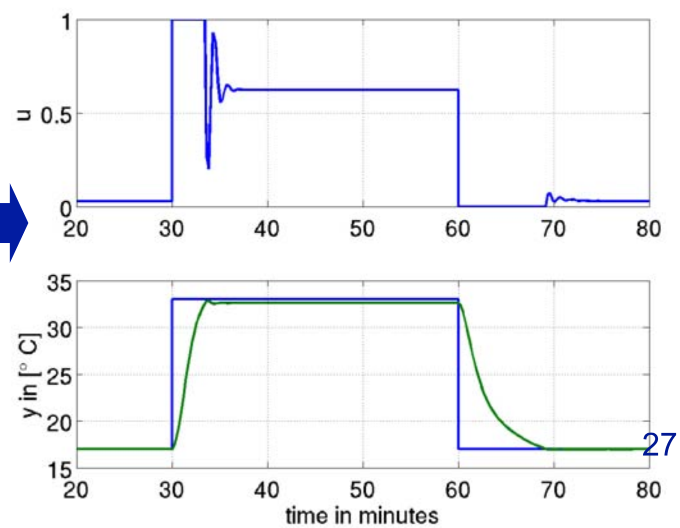
$$\dot{x}(t) = A\,x(t) + B\,u(t)$$
$$x(t) \in \Re^{40}$$



$$\dot{\tilde{x}}(t) = \tilde{A}\,\tilde{x}(t) + \tilde{B}\,u(t)$$
$$\tilde{x}(t) \in \Re^{3}$$



$$u(t) = K(y)\,e(t)$$



Overview

- Introduction

  Trends - Problems - Needs

- Mono-Simulation with Modelica

  Modelica Standard Library - LBNL Buildings Library - Applications

- Co-Simulation with Building Controls Virtual Test Bed

  Building Controls Virtual Test Bed - Applications

- R&D Needs

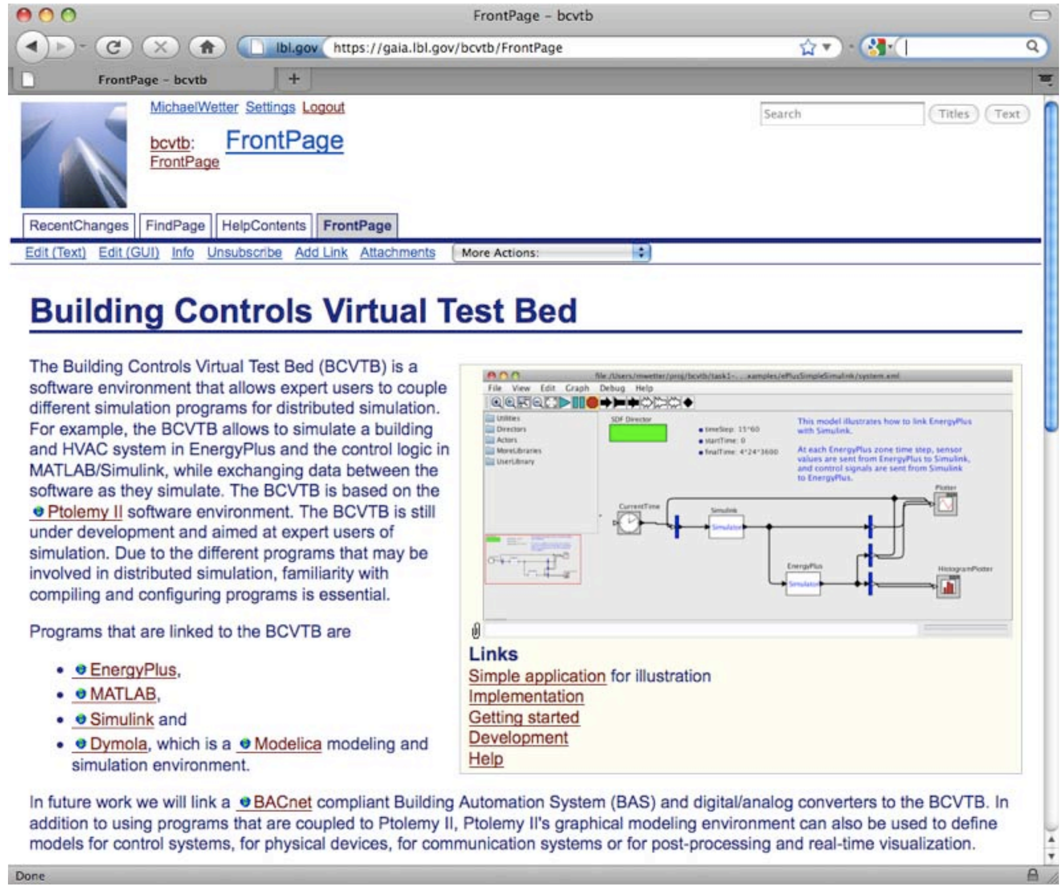
Building Controls Virtual Test Bed (BCVTB)

Enable

- Co-simulation for integrated multi-disciplinary analysis
- Use of domain-specific tools
- Model-based system-level design
- Model-based operation

Available from
http://simulationresearch.lbl.gov/bcvtb

Based on Ptolemy II from UC Berkeley, which will include BCVTB interface.

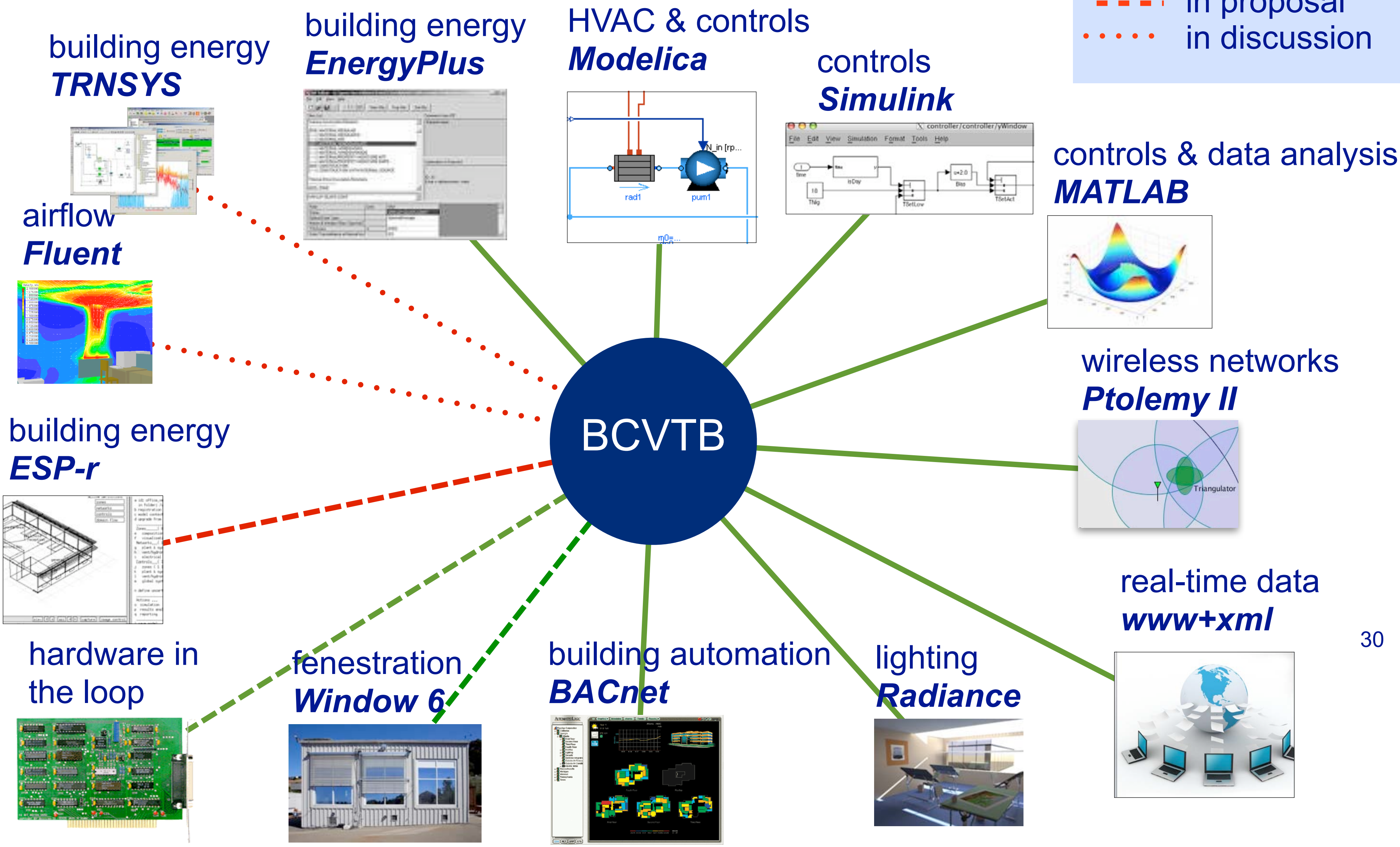


Acknowledgements

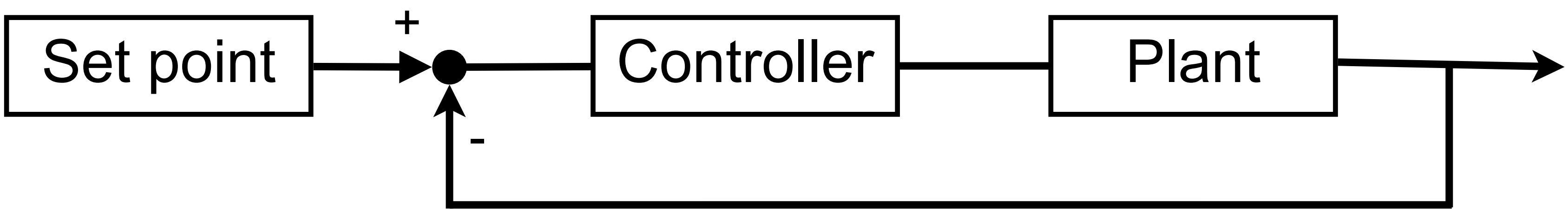
Prof. Edward A. Lee and Christopher Brooks (University of California at Berkeley), Gregor Henze, Charles Corbin, Anthony Florita and Peter May-Ostendorp (University of Colorado at Boulder), Rui Zhang (Carnegie Mellon), Zhengwei Li (Georgia Institute of Technology), Phil Haves and Andrew McNeill (LBNL).

Building Controls Virtual Test Bed

Open-source middle-ware based on UC Berkeley's Ptolemy II program.
Synchronizes and exchanges data as (simulation-)time progresses.



Simple Example: Room Heater



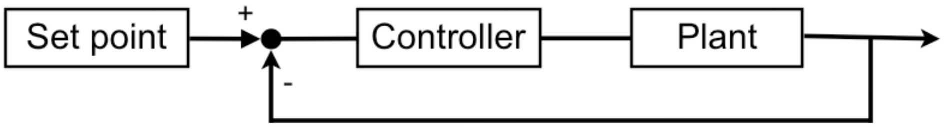
Controller: Discrete time proportional controller

$$y(k+1) = \max(0, \min(1, K_p\,(T_{set} - T(k))))$$

Plant: Room model

$$T(k+1) = T(k) + \frac{\Delta t}{C}\left(UA\,(T_{out} - T(k)) + \dot{Q}_0\,y(k)\right)$$

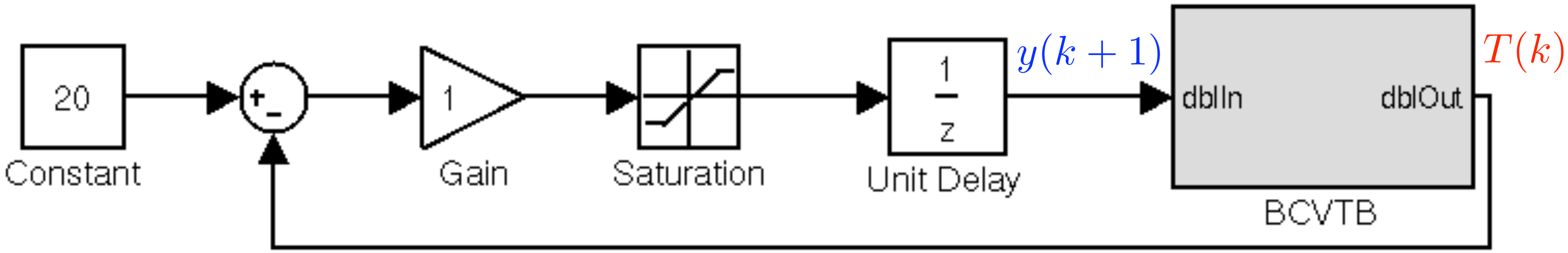
Simple Example: Room Heater



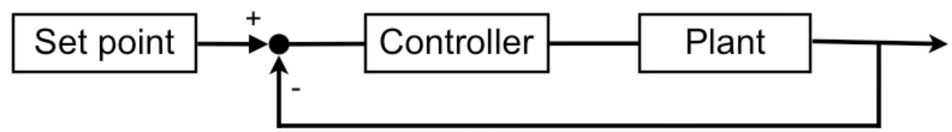
Discrete Time Proportional Controller

$$y(k+1) = \max(0, \min(1, K_p\,(T_{set} - T(k))))$$

Implementation in Simulink



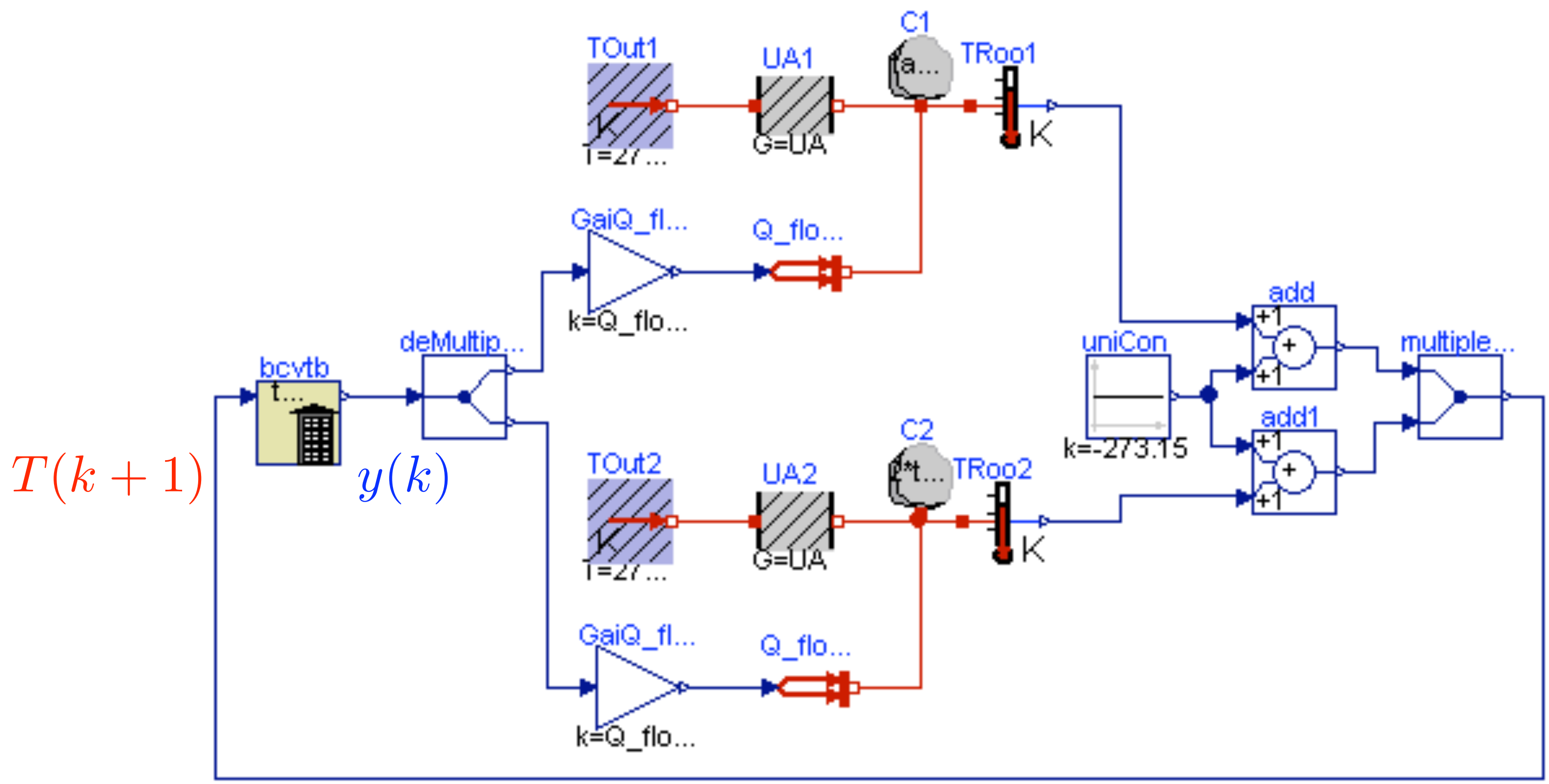
Simple Example: Room Heater



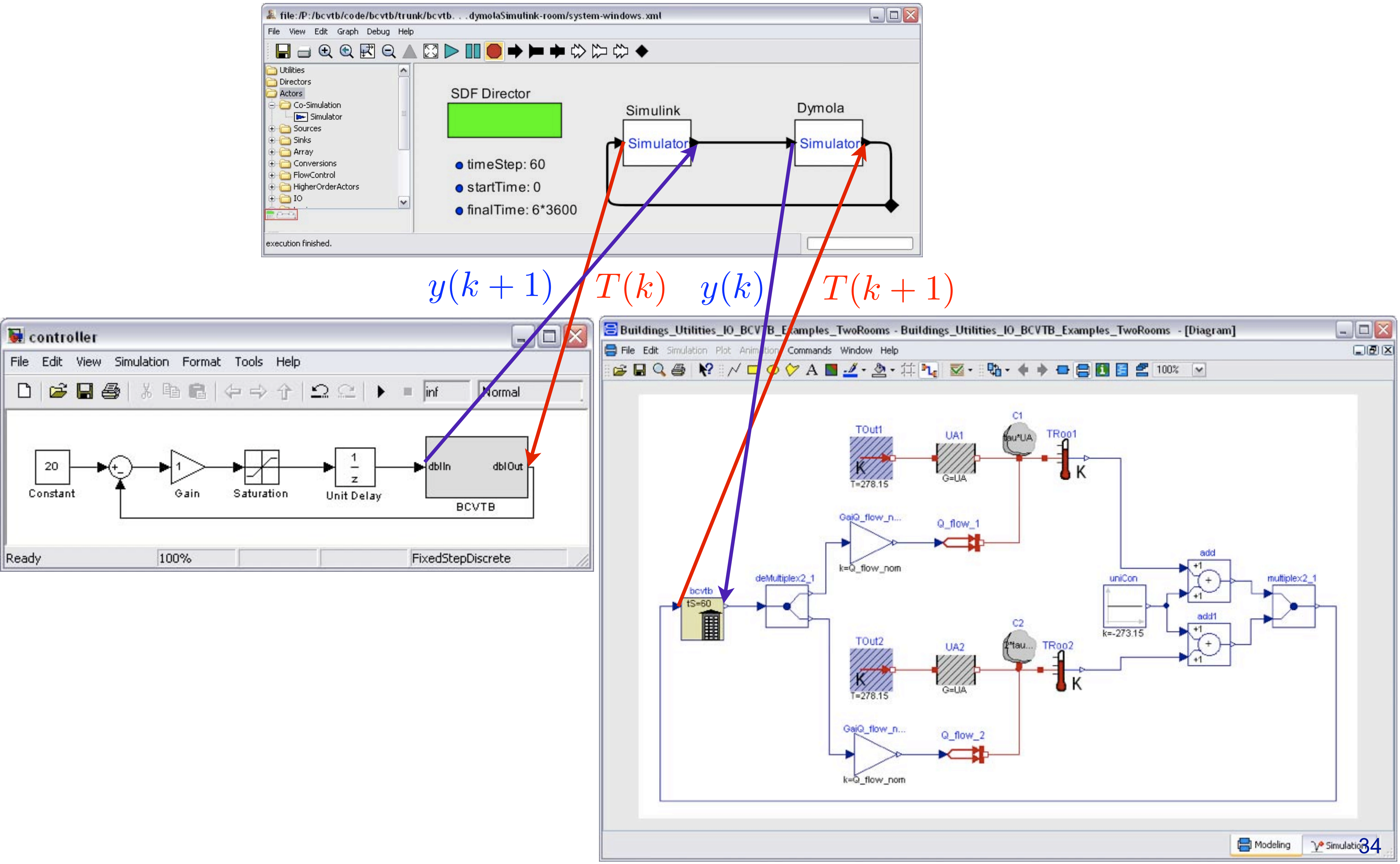
Room model

$$\color{red}{T(k+1)} = T(k) + \frac{\Delta t}{C}\left(UA\,(T_{out} - T(k)) + \dot{Q}_0\,\color{blue}{y(k)}\right)$$

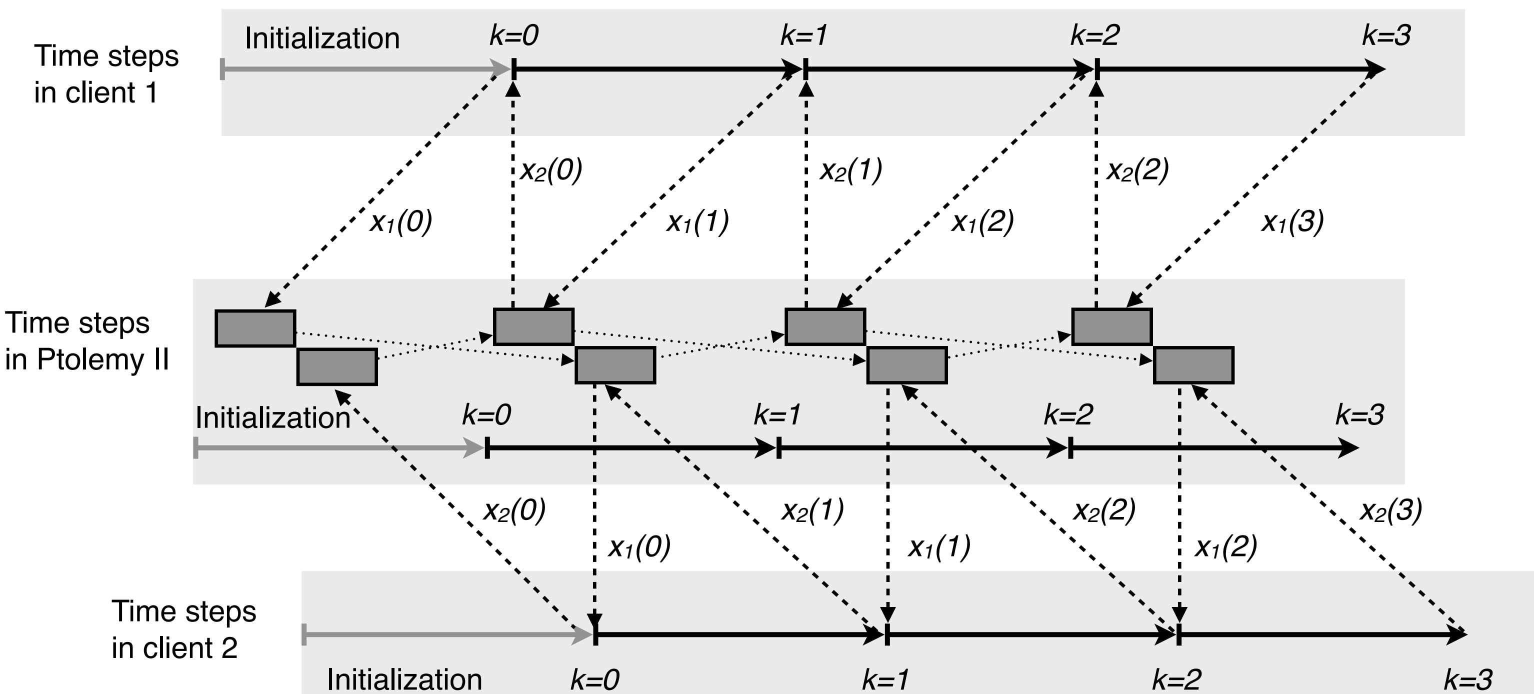
Implementation in Modelica



Simple Example: Room Heater



BCVTB Time Synchronization



Legend

Call to a Ptolemy II `Simulator` actor.

Exchange of data between two Ptolemy II `Simulator` actors.

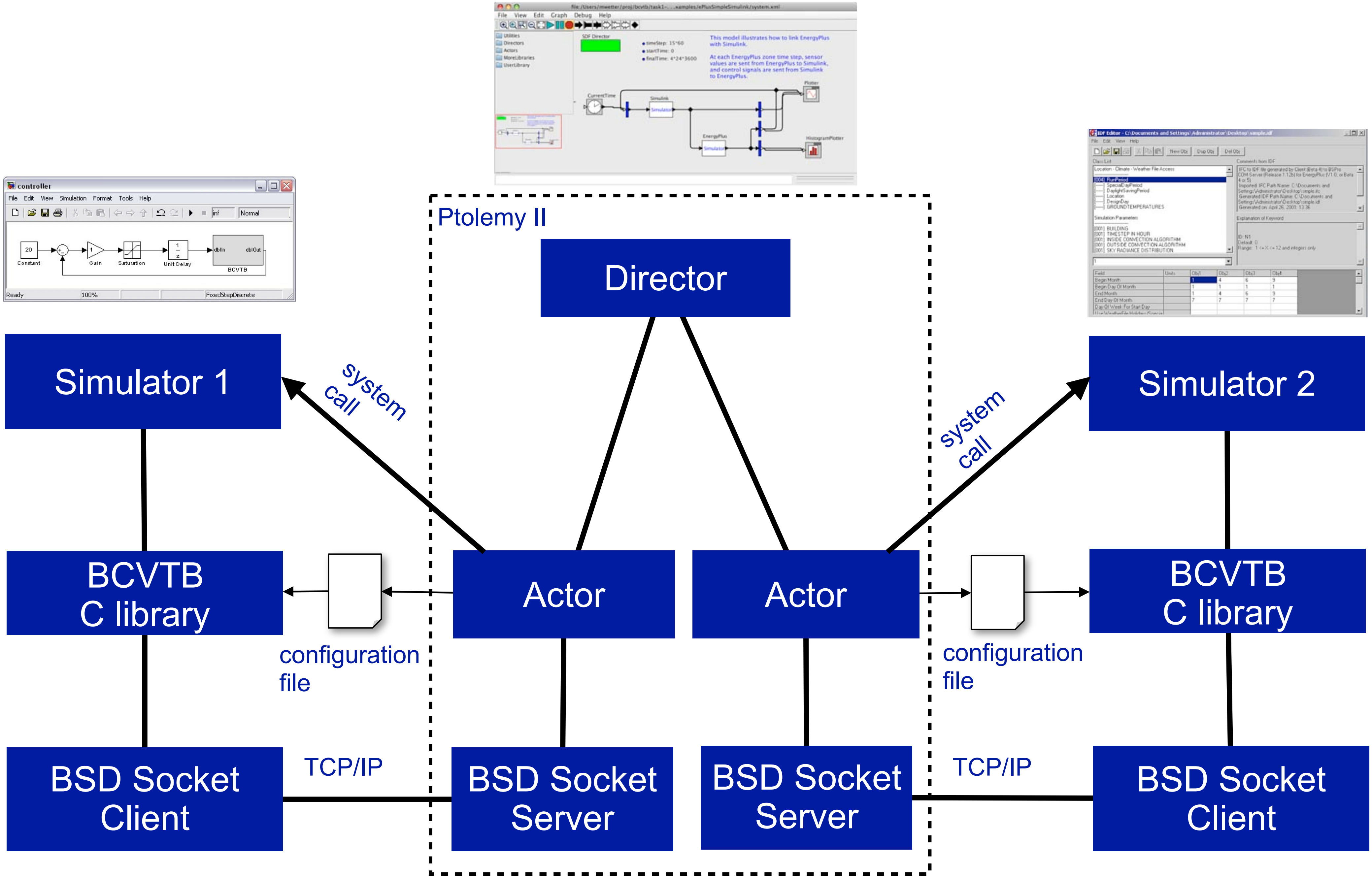
Exchange of data between a Ptolemy II `Simulator` actor and a simulation program.

Initialization step in Ptolemy II, and in simulation programs.

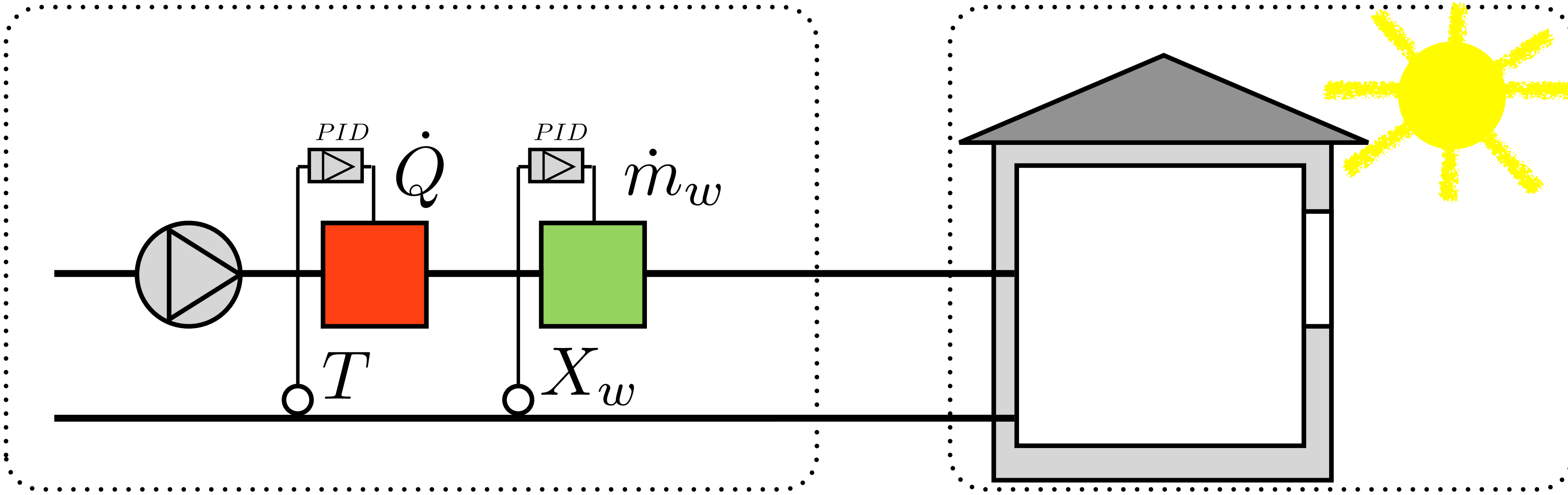
Time step in Ptolemy II, and in simulation programs.

$x_1(3)$ Variable *x* of simulator 1 at time step 3.

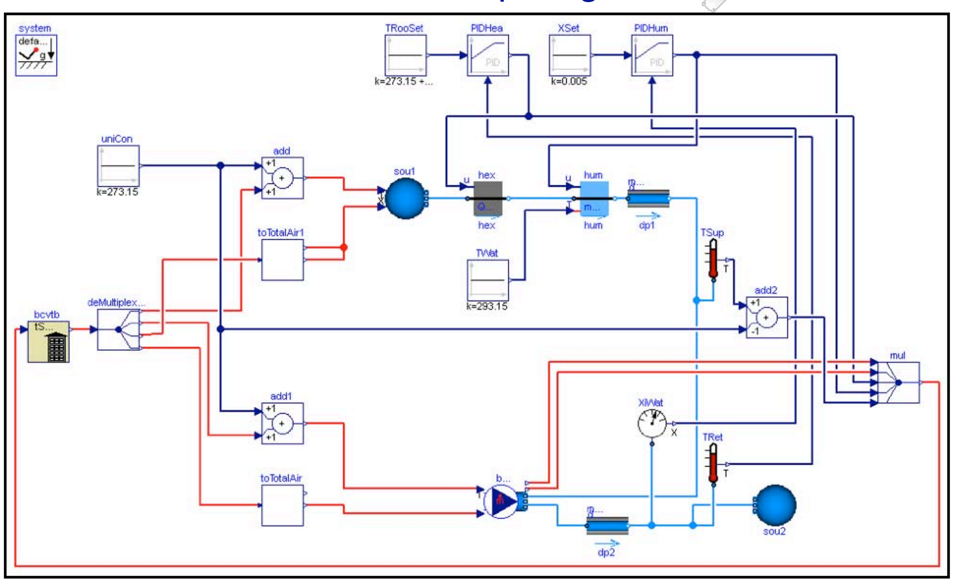
BCVTB Architecture



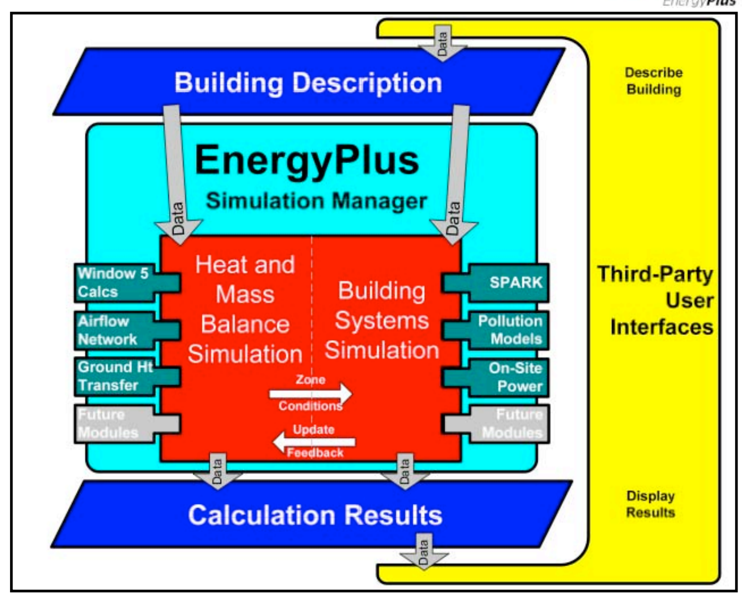
Ex: HVAC in Modelica, Building in EnergyPlus



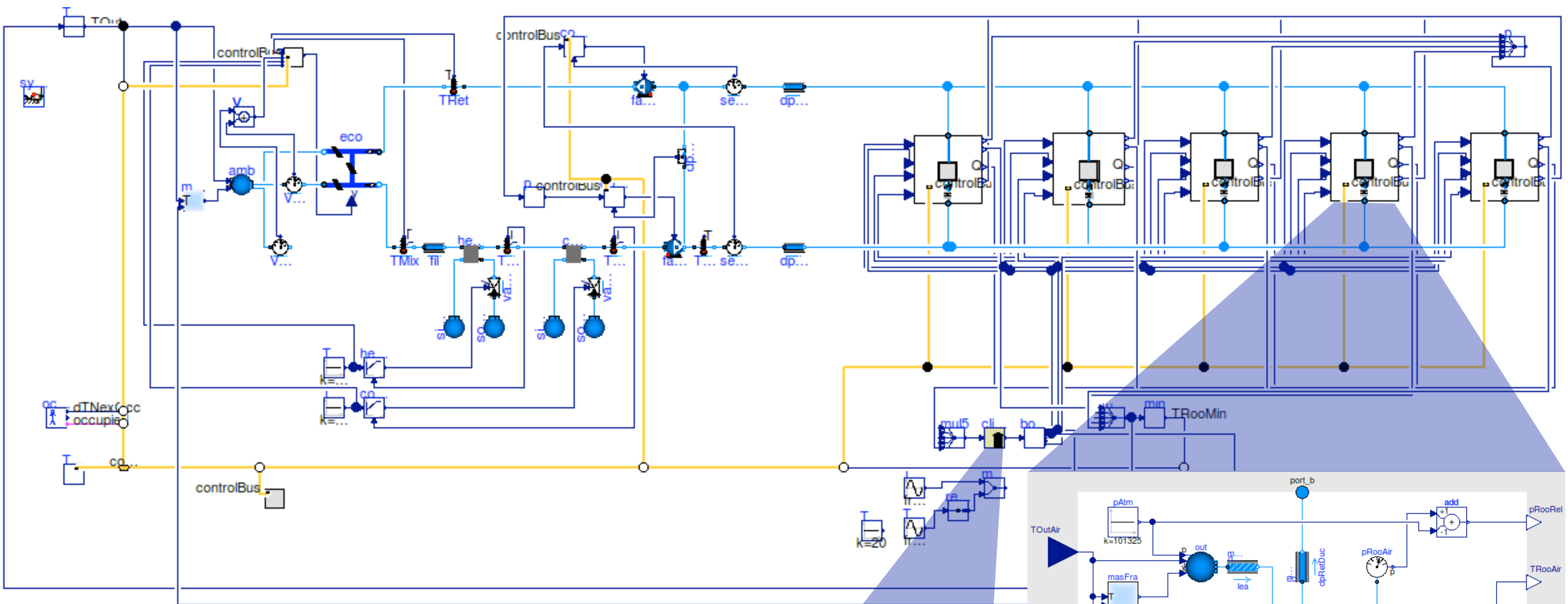
Rapid virtual prototyping.
Path towards embedded computing.

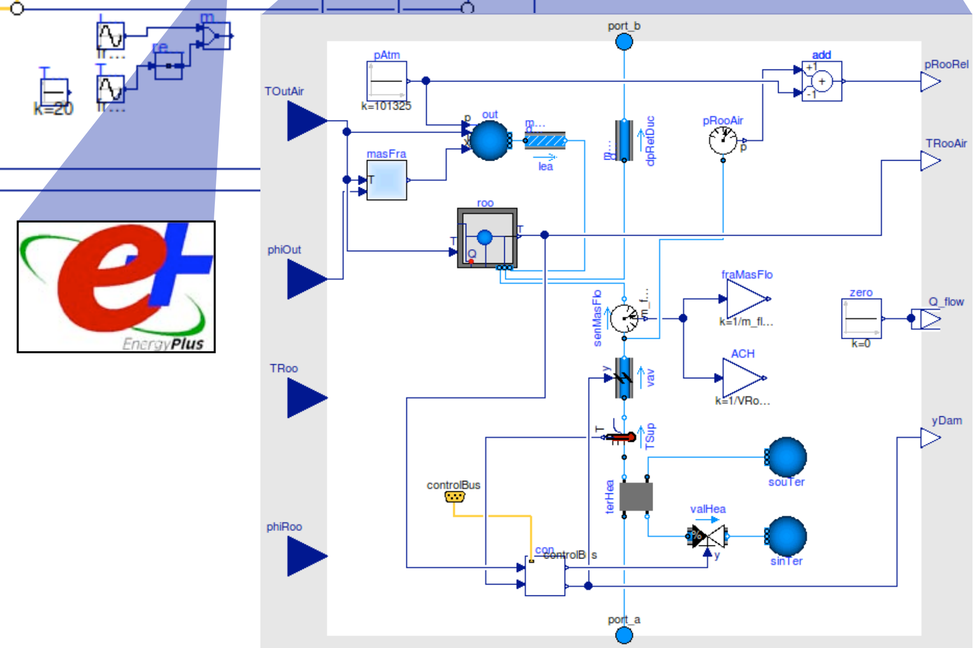


Whole building energy analysis.
Reuse of 500,000 lines of code.



Modelica Implementation of VAV System with ASHRAE Standard Sequence of Controls, linked to EnergyPlus DOE Benchmark Bldg.

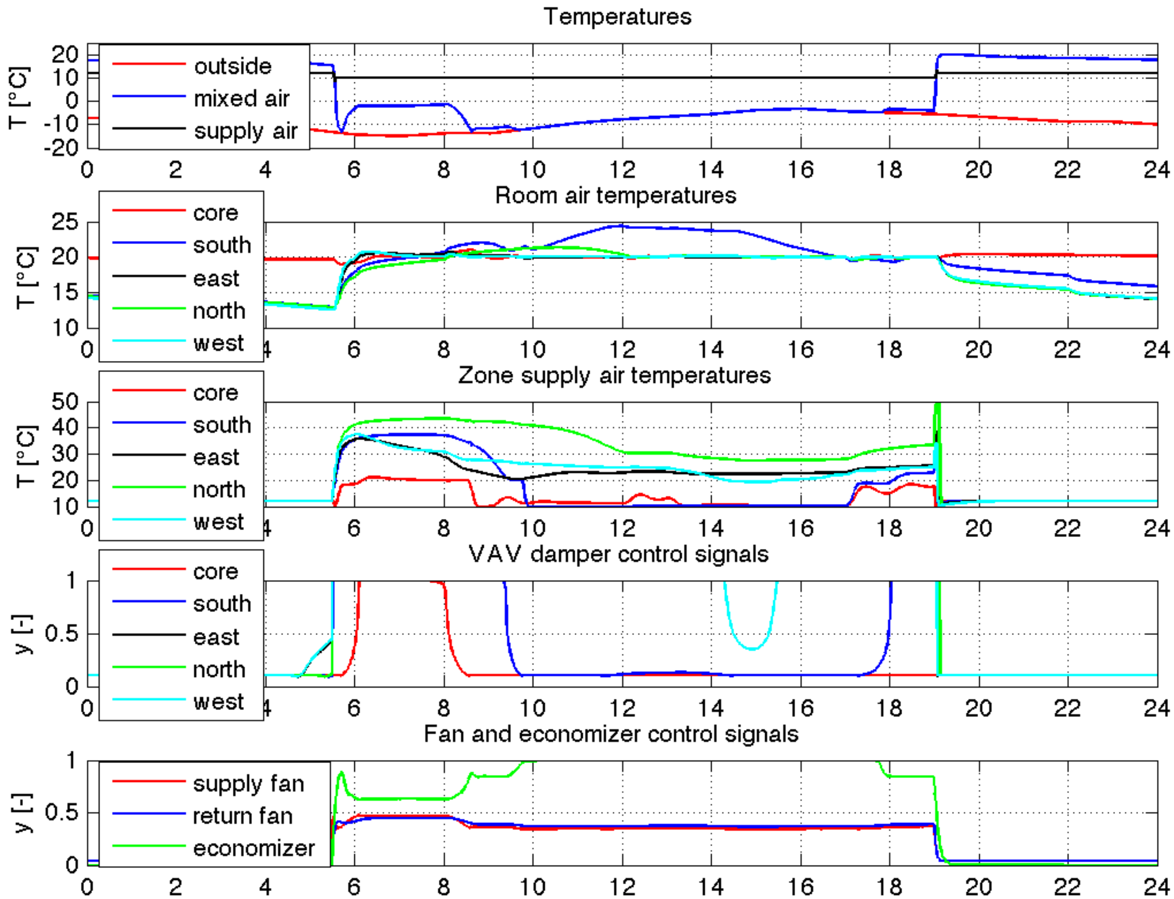




Variable Air Volume Flow (VAV) System
Terminal reheat.
Finite state machine for supervisory mode transition.
Local loop control with P and PI controllers.

Original system model
1200 components
4,300 equations
150 state variables

Modelica Implementation of VAV System with ASHRAE Standard Sequence of Controls, linked to EnergyPlus DOE Benchmark Bldg.



Can execute realistic supervisory and local loop control in Modelica linked to EnergyPlus through co-simulation.

R&D Needs:

Support for rapid virtual prototyping.

Robustness of DAE solver.

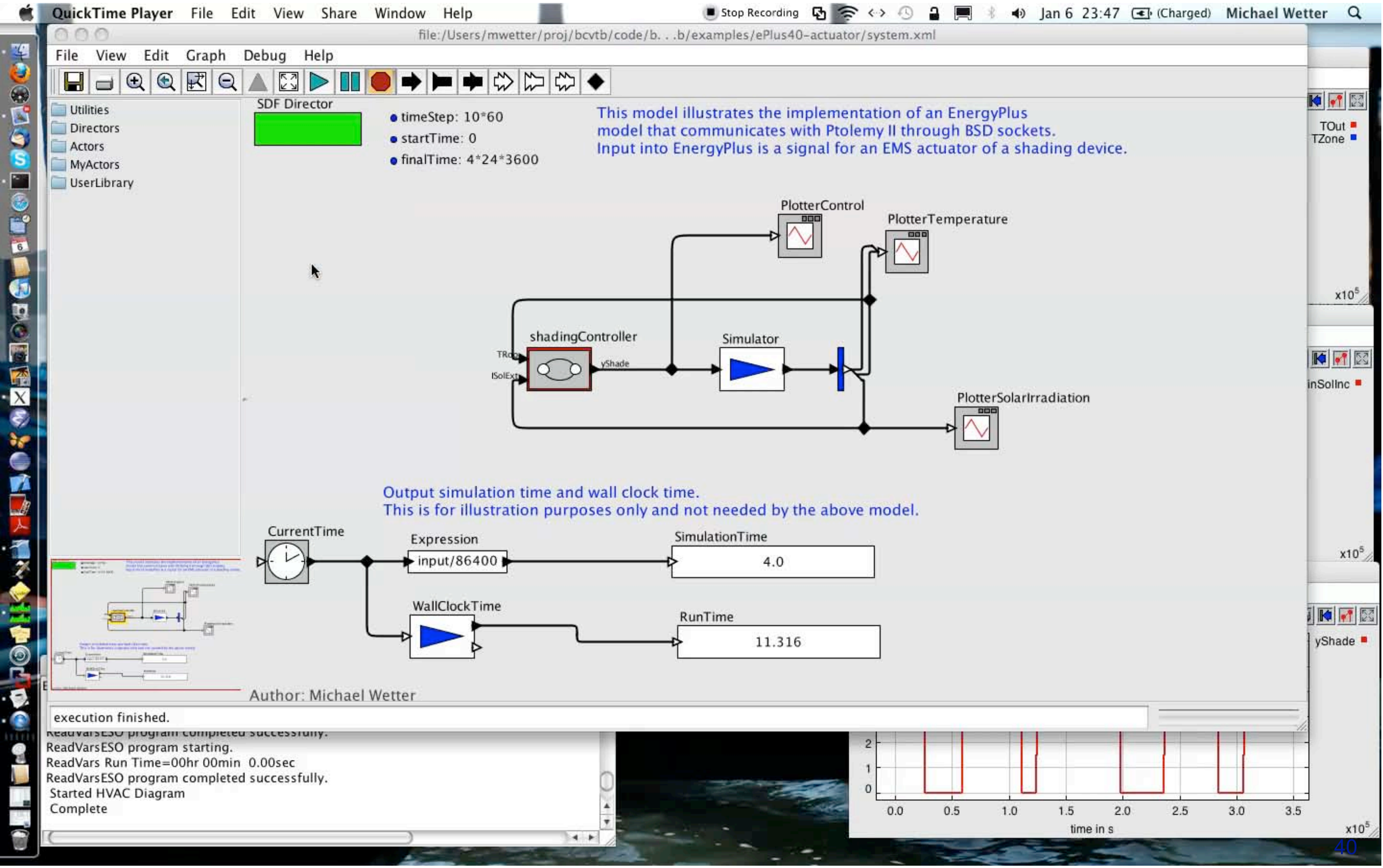
Computing time.

“Packaging” to make technology easier accessible to non-experts.

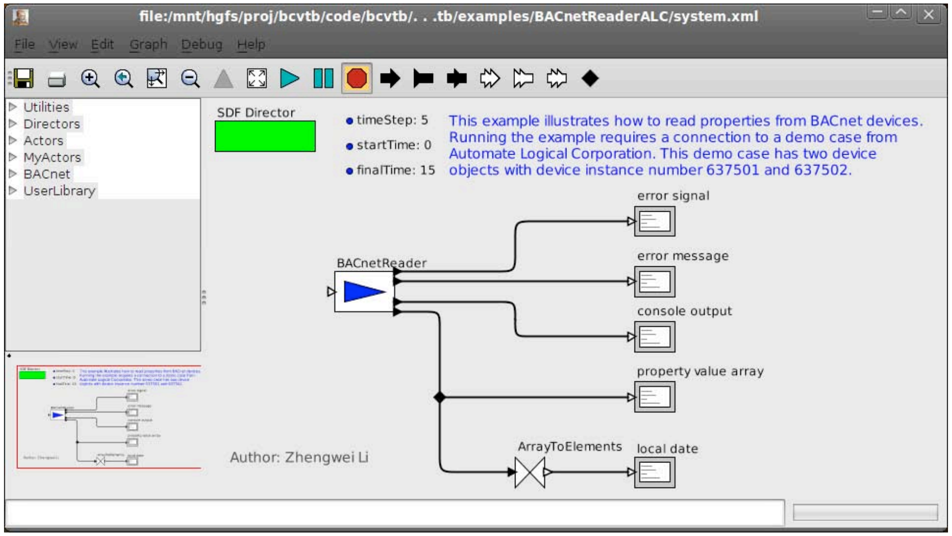
Scope of validated model library (HVAC & control components and systems).

Code generation to link design to operation.

Control in BCVTB

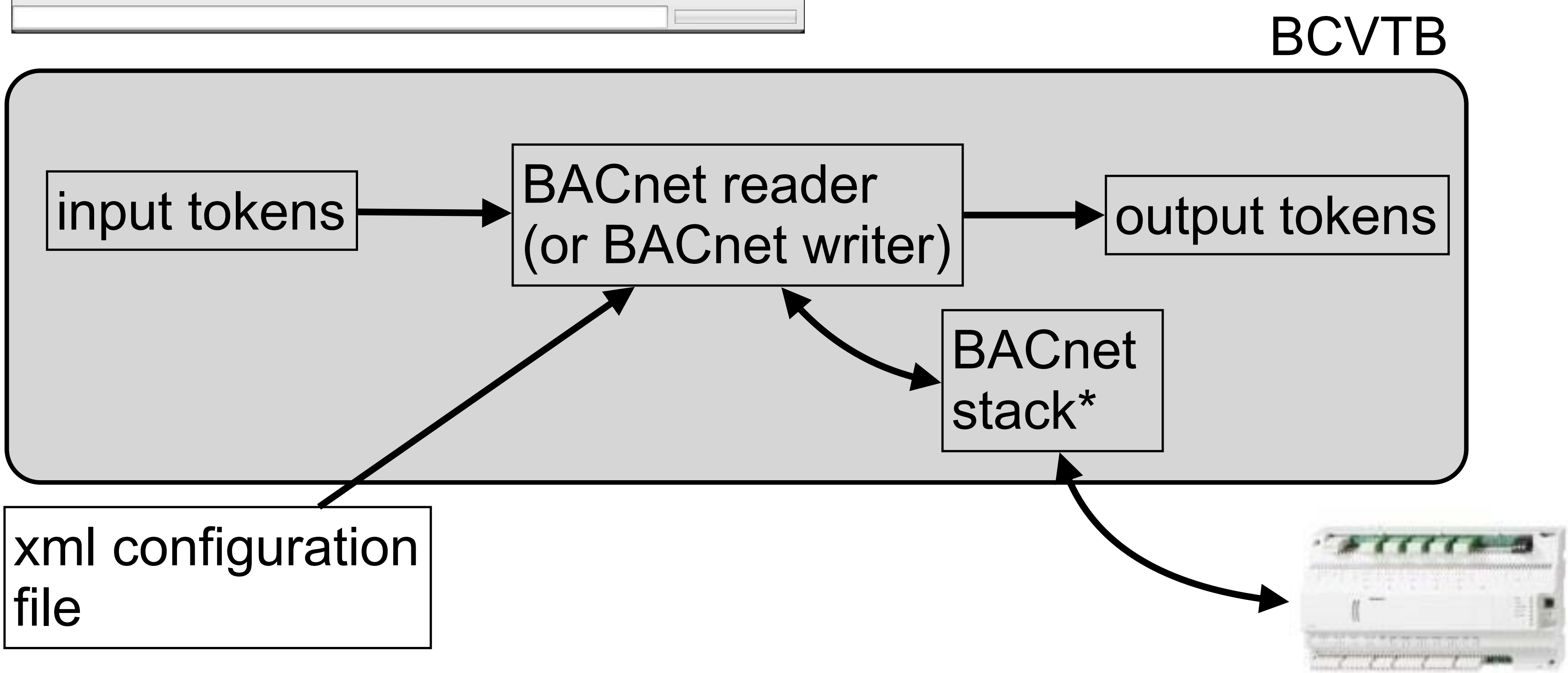


BACnet



Enables simulation and/or data analysis coupled to Building Automation Systems.

Can read to and write from BACnet devices.

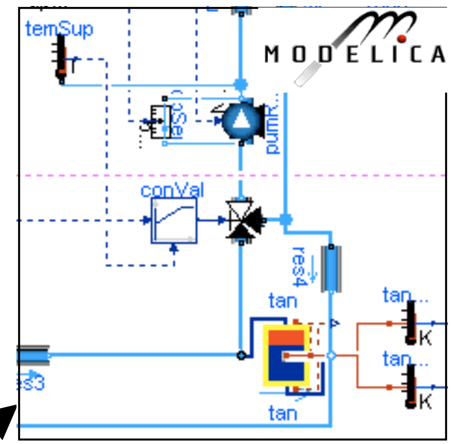


* BACnet stack from http://bacnet.sourceforge.net (Steve Karg)

Reusable modules for model-based operation

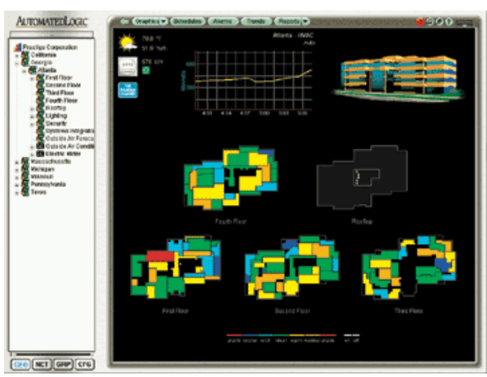
Tool selection depends on required
- model resolution
    - emulation of actual control & operation
    - dynamics of equipment
- analysis capabilities
    - smoothness
    - state initialization

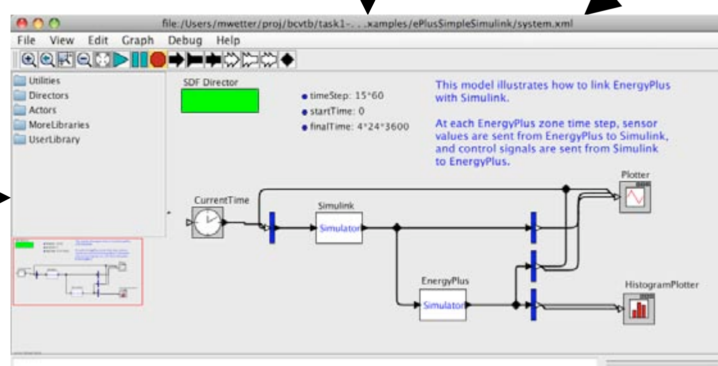
Hybrid systems, emulate actual feedback control

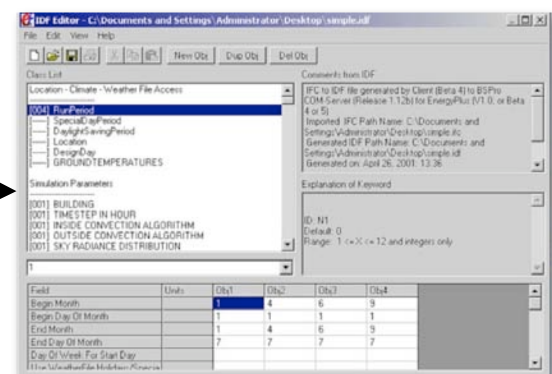


www/xml

Discrete time, idealized controls







EnergyPlus

Overview

- Introduction

  Trends - Problems - Needs

- Mono-Simulation with Modelica

  Modelica Standard Library - LBNL Buildings Library - Applications

- Co-Simulation with Building Controls Virtual Test Bed

  Building Controls Virtual Test Bed - Applications

- R&D Needs

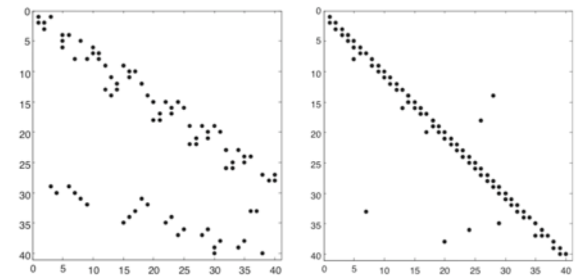
R&D Needs

Model-based design process & deployment

- development of design flows that are based on executable specifications (process & supporting tools)
- extension of Building Information Models to support controls through executable controls specifications

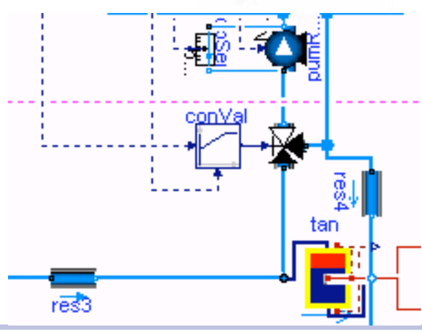
Equation-based object-oriented simulation

- robust DAE solvers for thermo-fluid & control systems
- multi-rate solvers
- mapping equations to parallel hardware



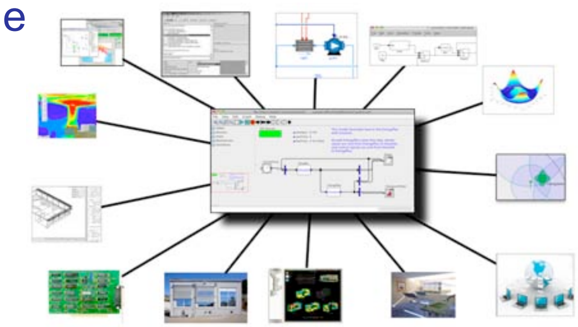
Equation-based object-oriented modeling

- computationally efficient and numerically robust model formulation
- standardized libraries



Co-simulation

- semantics of exchanged data
- standardized data exchange
- adaptive step size



Model Predictive Controls

- MPC that can be deployed to 100,000’s of buildings

Others....

- system-level fault detection and diagnostics
- cybernetic buildings

Vision

Use open standards to enable model-based design and operation for very low energy buildings.

http://www.bacnet.org/
http://www.buildingsmart.com/

http://functional-mockup-interface.org/
http://www.modelica.org/