OpenBuildingControl
—
Team meeting

Michael Wetter
Philip Haves
Jianjun Hu
Milica Grahovac

November 15, 2018

Lawrence Berkeley National Laboratory

Upcoming deadlines

By Q6, demonstrate with an actual measured control response that the controls verification can signal satisfied, undecided, and violated test results.

—

By Q7, release a version of the control library for primary systems, facade and lighting in Modelica on http://github.org/lbl-srg/modelica-buildings.

By Q7, release first version of the controls verification test module.

By Q8, release case study report.

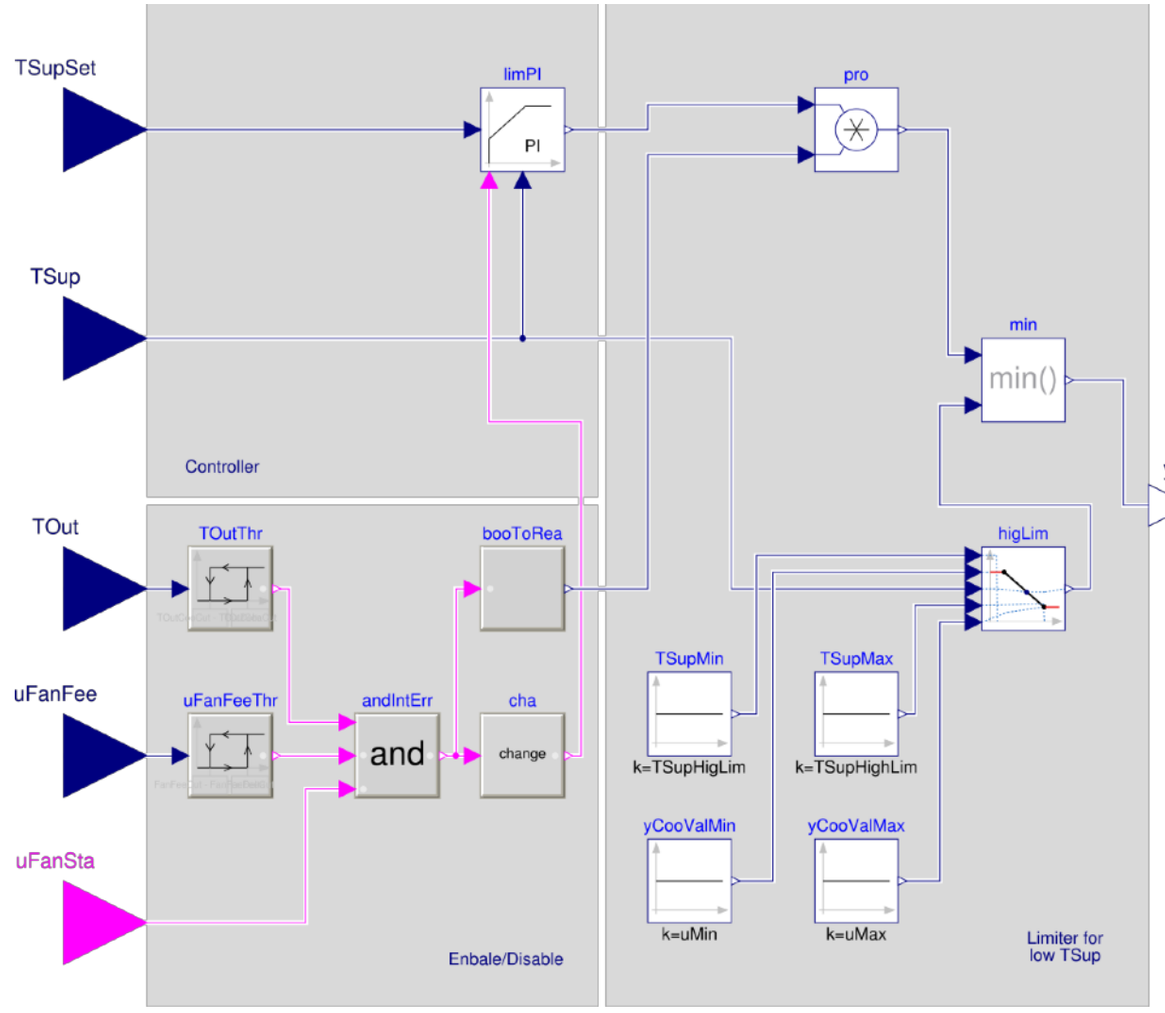
By Q8, demonstrate importing and exporting CDL in the control design tool.
Todo: Render connections that overlap or connect to the same input/output.

By Q8, write first version of commercialization and market transformation plan with the goal to show value and obtain commitment from large owners and design firms for the process.

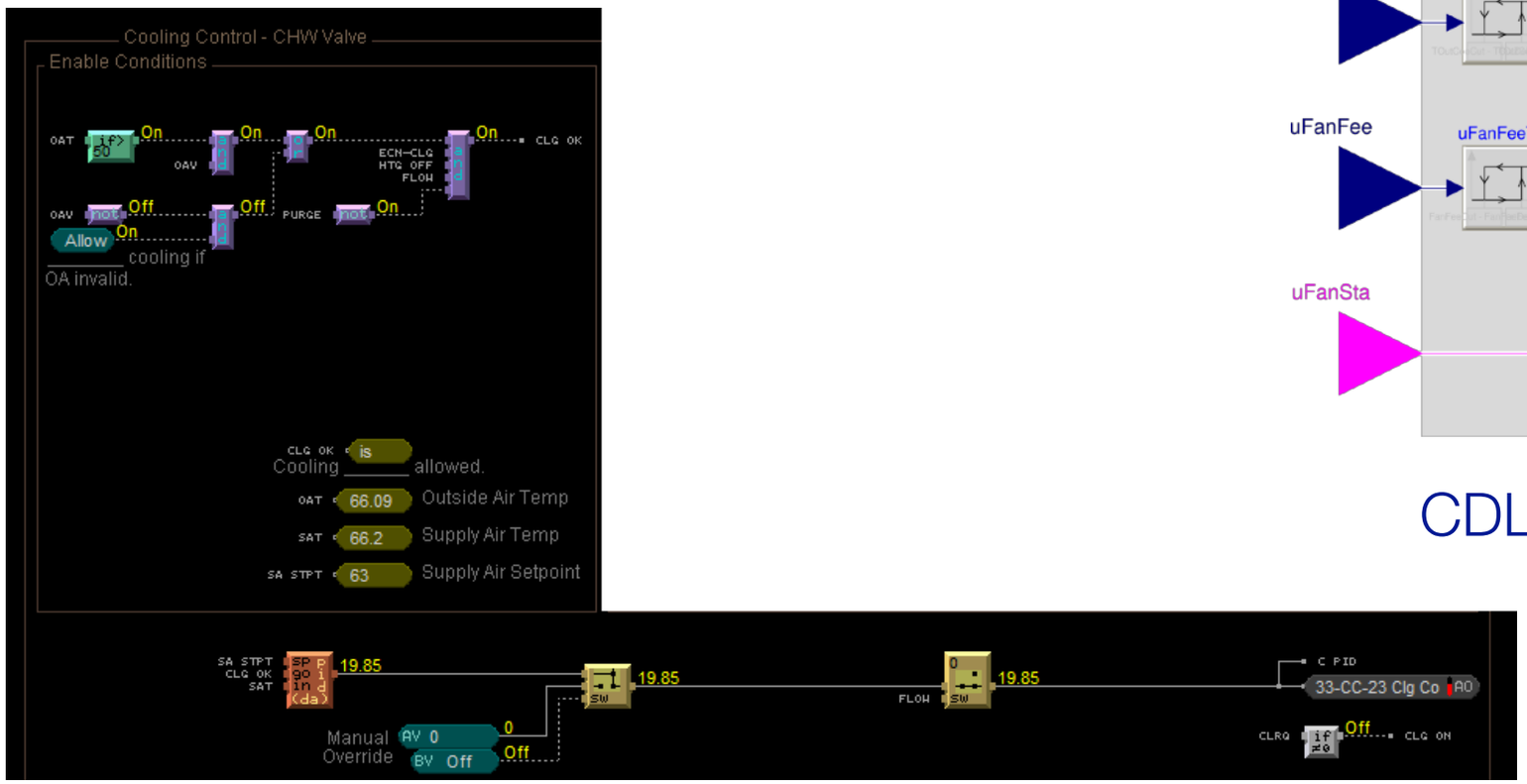
Verification test with a measured control response - Sequence specification

We validated a **trended output** of a control sequence that defines the **cooling coil valve** position.

The cooling coil valve sequence is a part of the ALC EIKON control logic implemented in building 33 at LBNL.



CDL specification

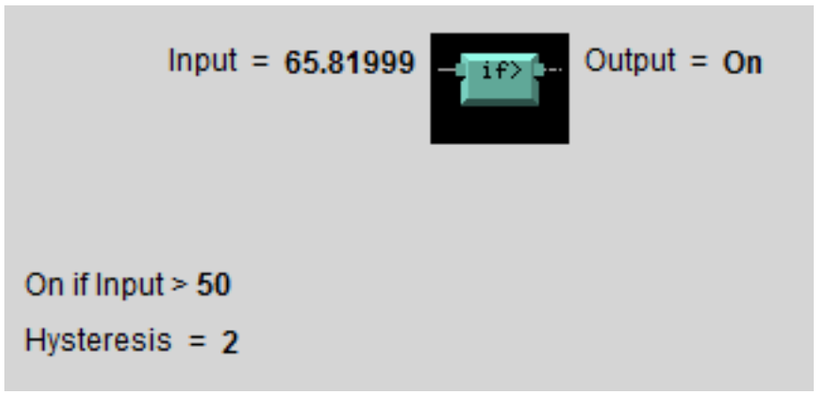


ALC EIKON specification

Verification test with a measured control response - ALC EIKON parameter collection

We recorded ALC EIKON sequence parameters and input trends with a 5s interval:

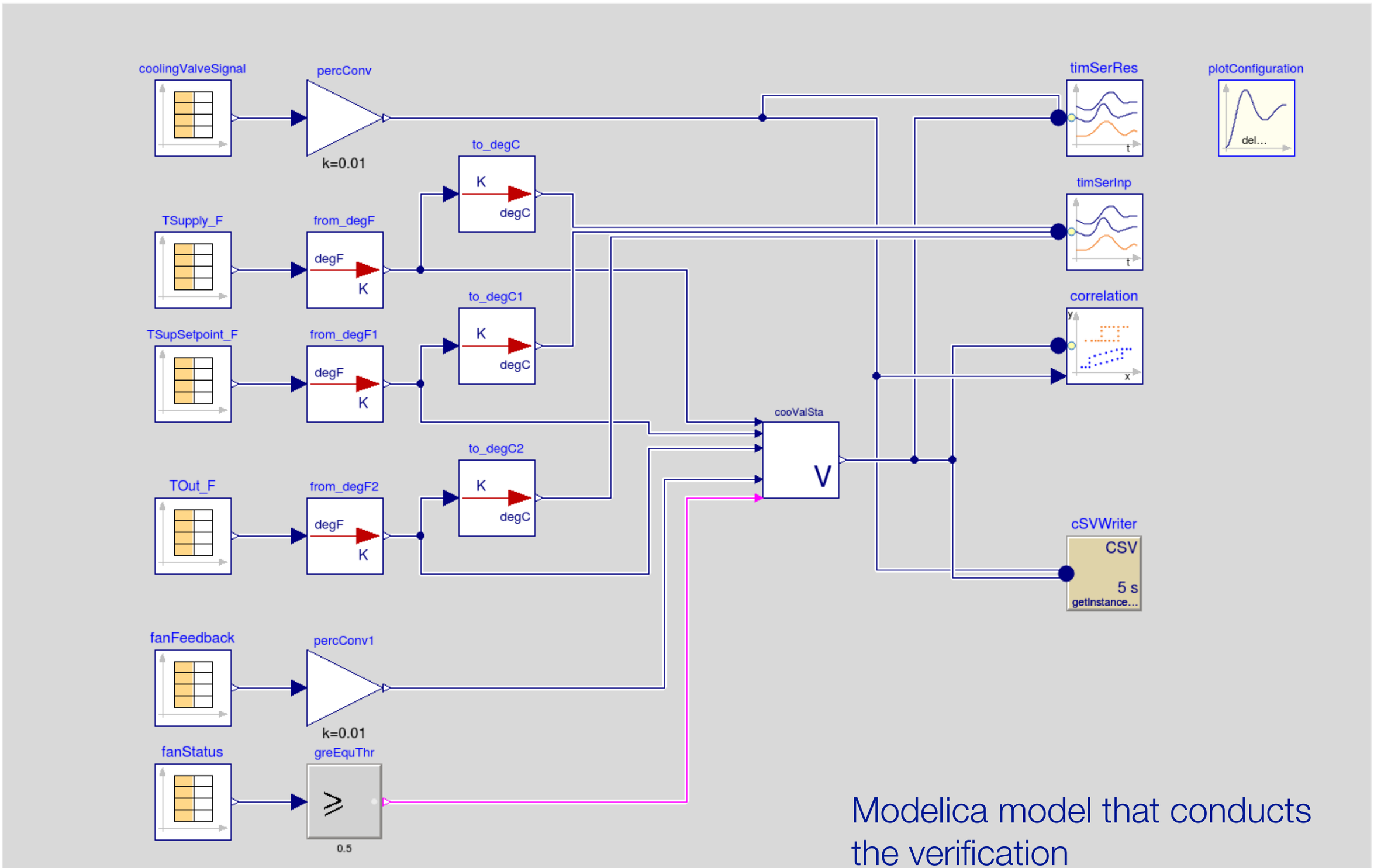
- Supply air temperature [F]
- Supply air temperature setpoint [F]
- Outdoor air temperature [F]
- VFD fan enable status [0/1]
- VFD fan feedback [%]



ALC outdoor air temperature hysteresis to enable/disable the controller

ALC PI controller parameters

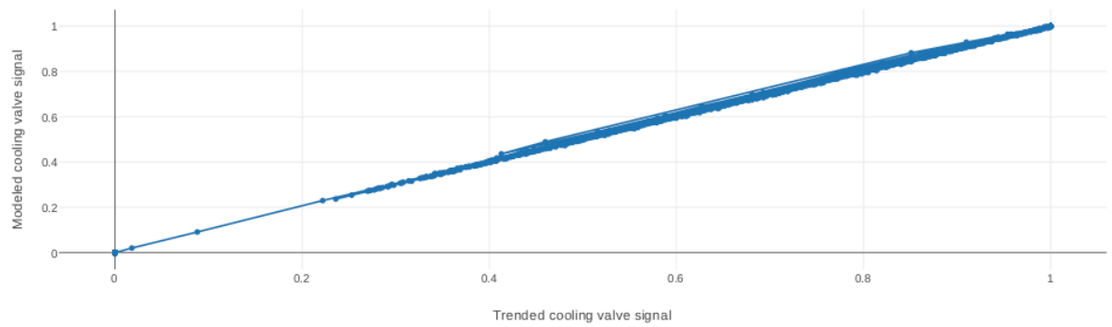
Verification test with a measured control response - Conducting the verification



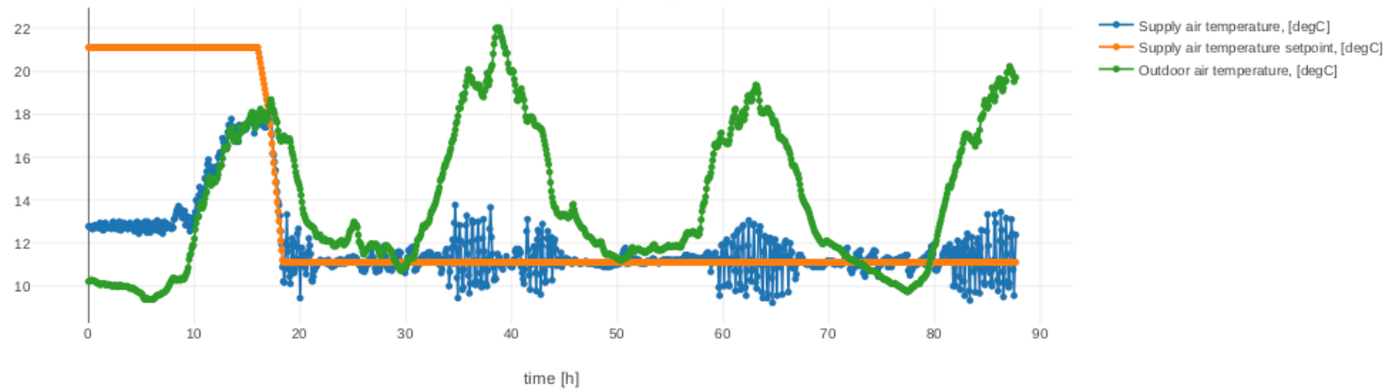
Modelica model that conducts the verification

Verification test with a measured control response - Verification results

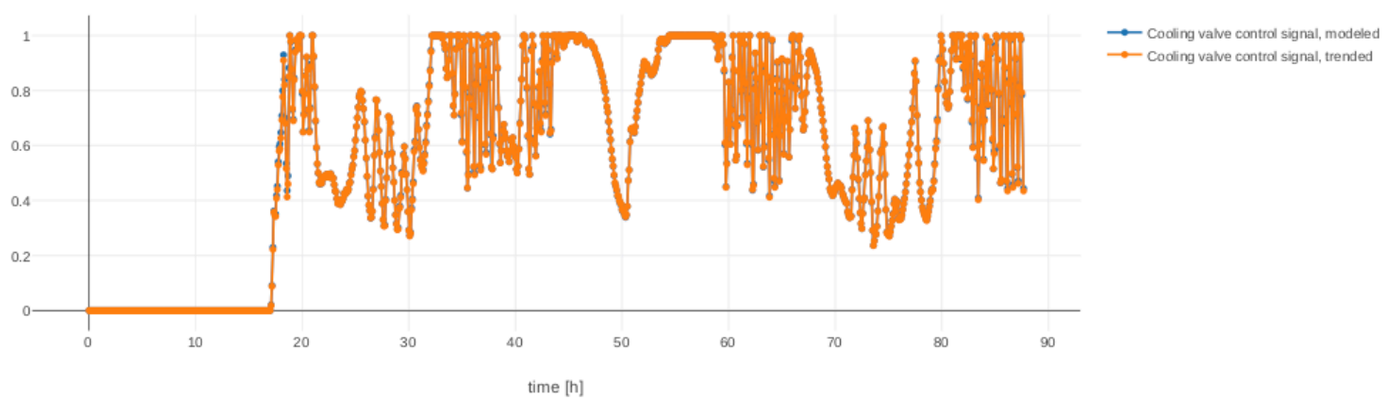
Modeled result/recorded trend correlation



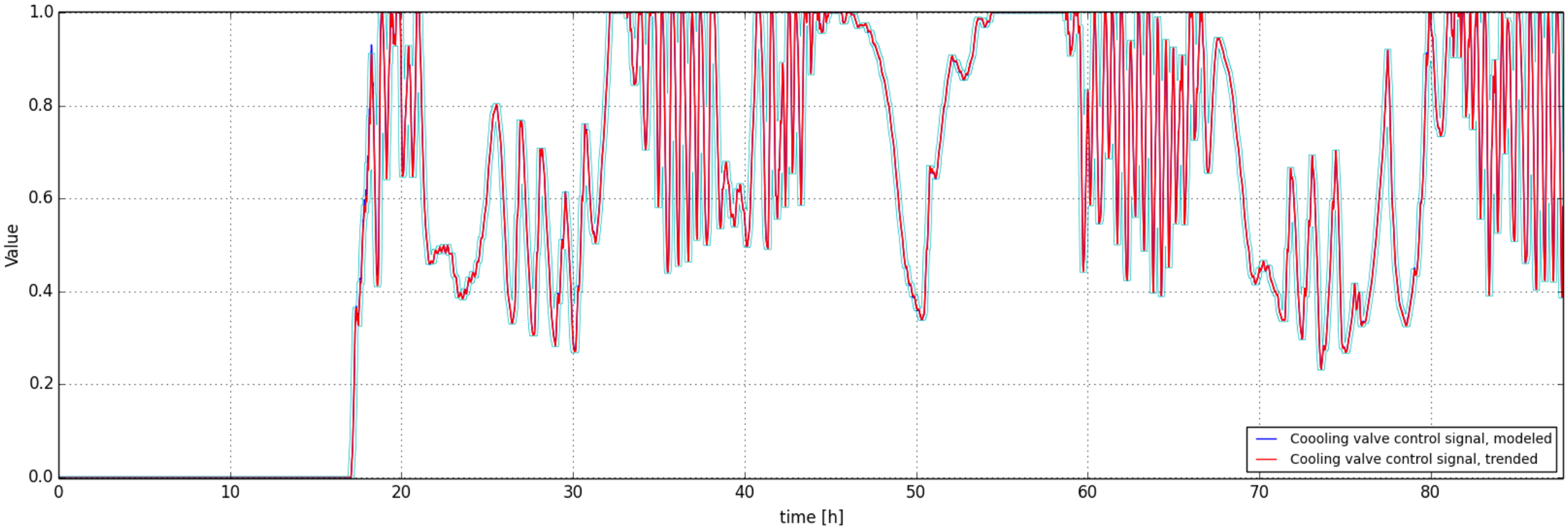
Trended input signals



Cooling valve control signal: reference trend vs. modeled result

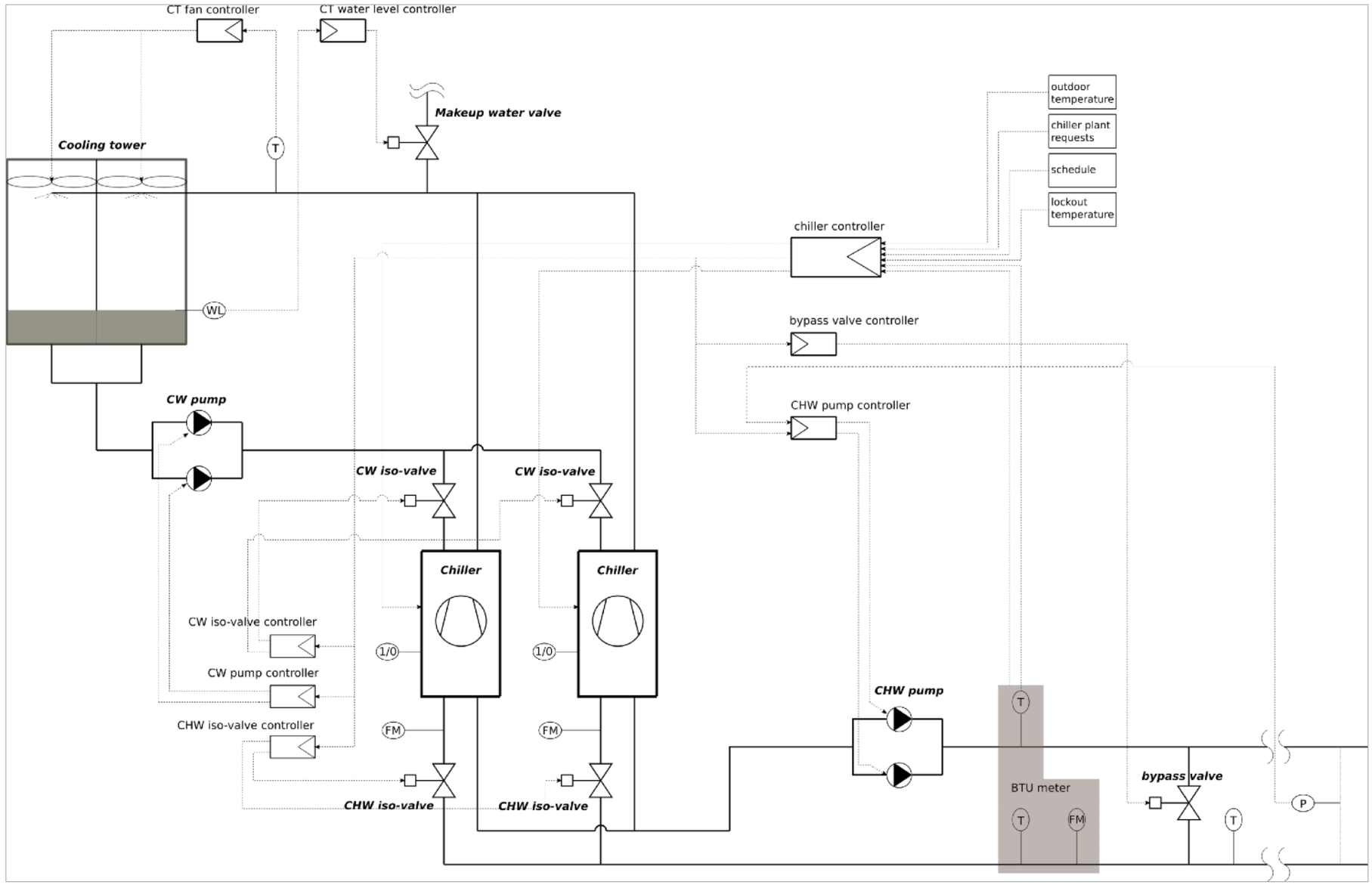


Verification test with a measured control response - Verification results using the funnel tool



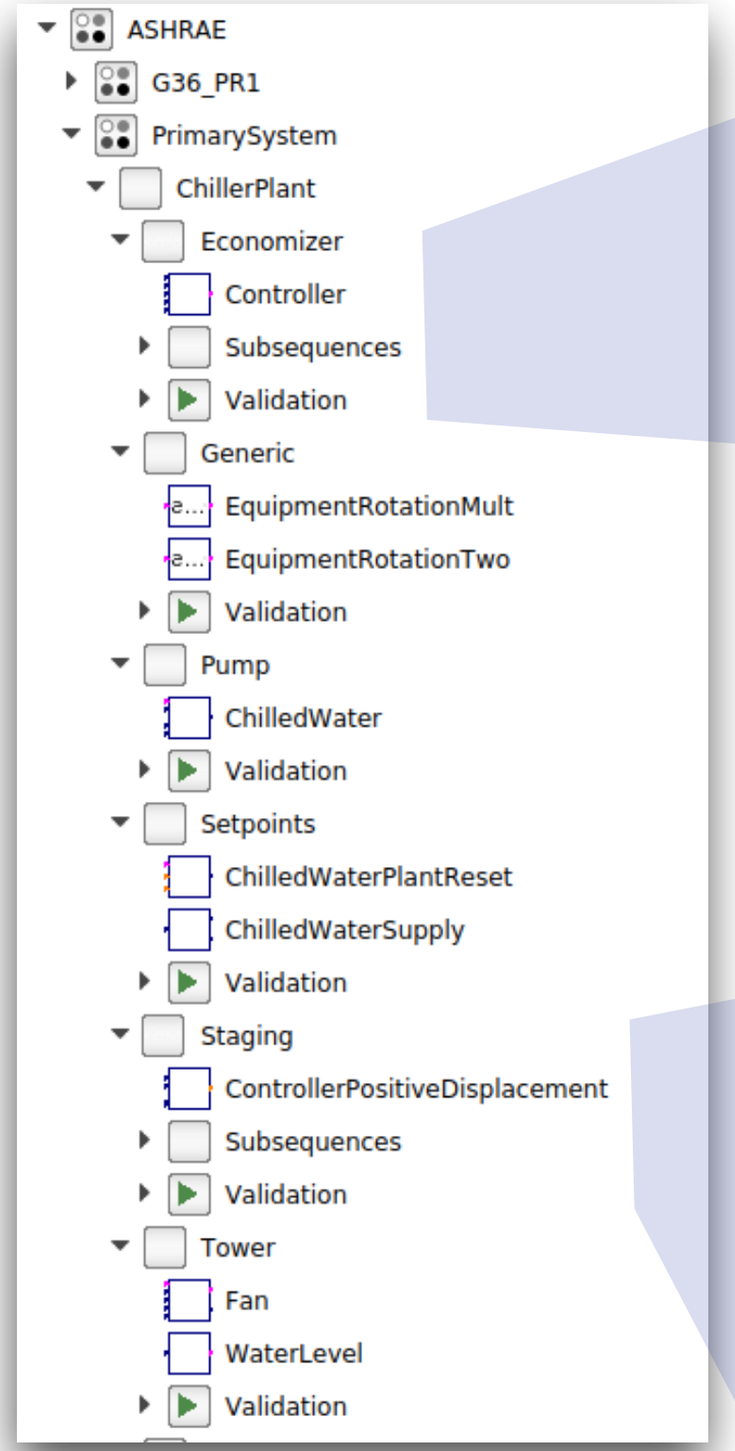
Primary sequence implementation I

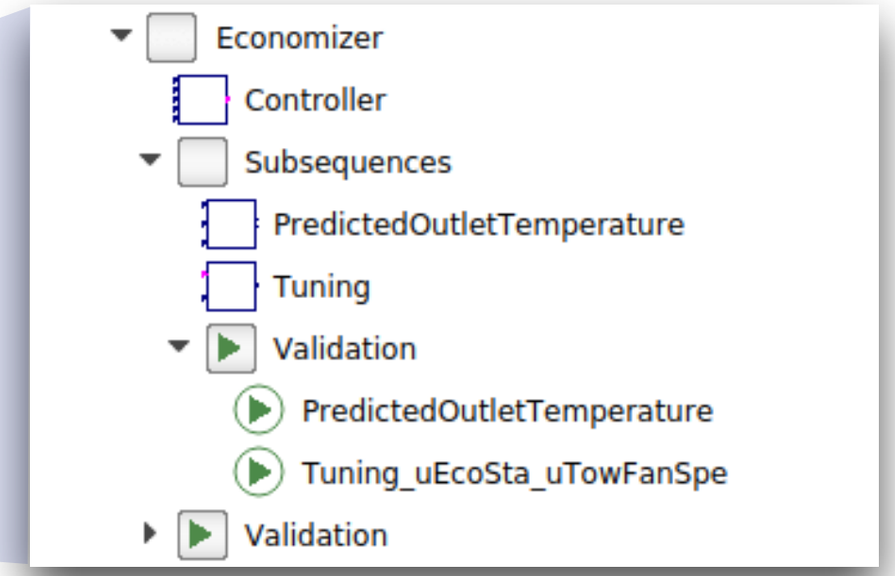
- Developed a typical plant control schematics and sequences based on “ASHRAE Fundamentals of Chilled Water Plant Design and Control SDL, Chapter 7. Controls”:
  - 2 chillers, 2 CT, 2 CHWP, 2 CWP



- Included water side economizer in the package

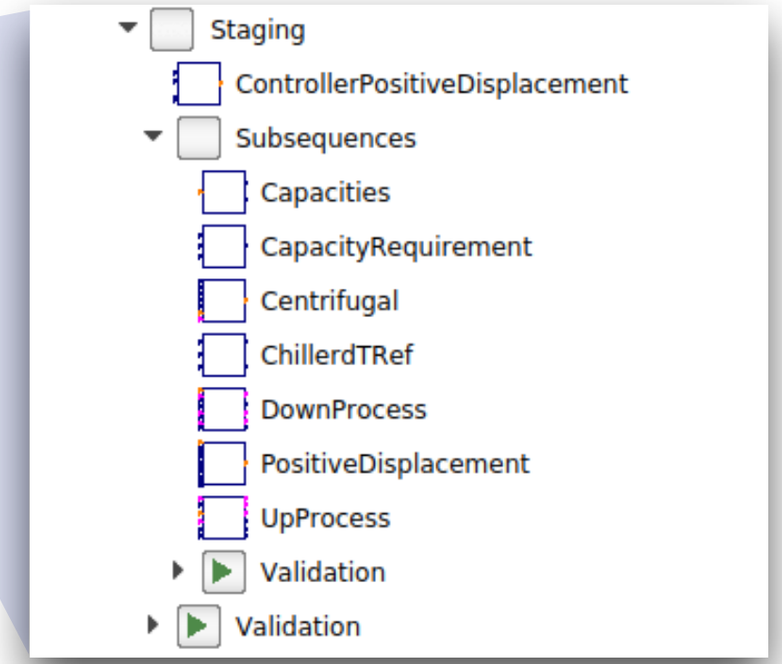
Primary sequence implementation II





Next steps:

- Comply with latest ASHRAE RP-1711 primary sequences document
- Create top level user facing controllers
- Review the package and include in the library



Case study I

Chiller plant in a commercial office building in Hacienda Business Park in Pleasanton, California

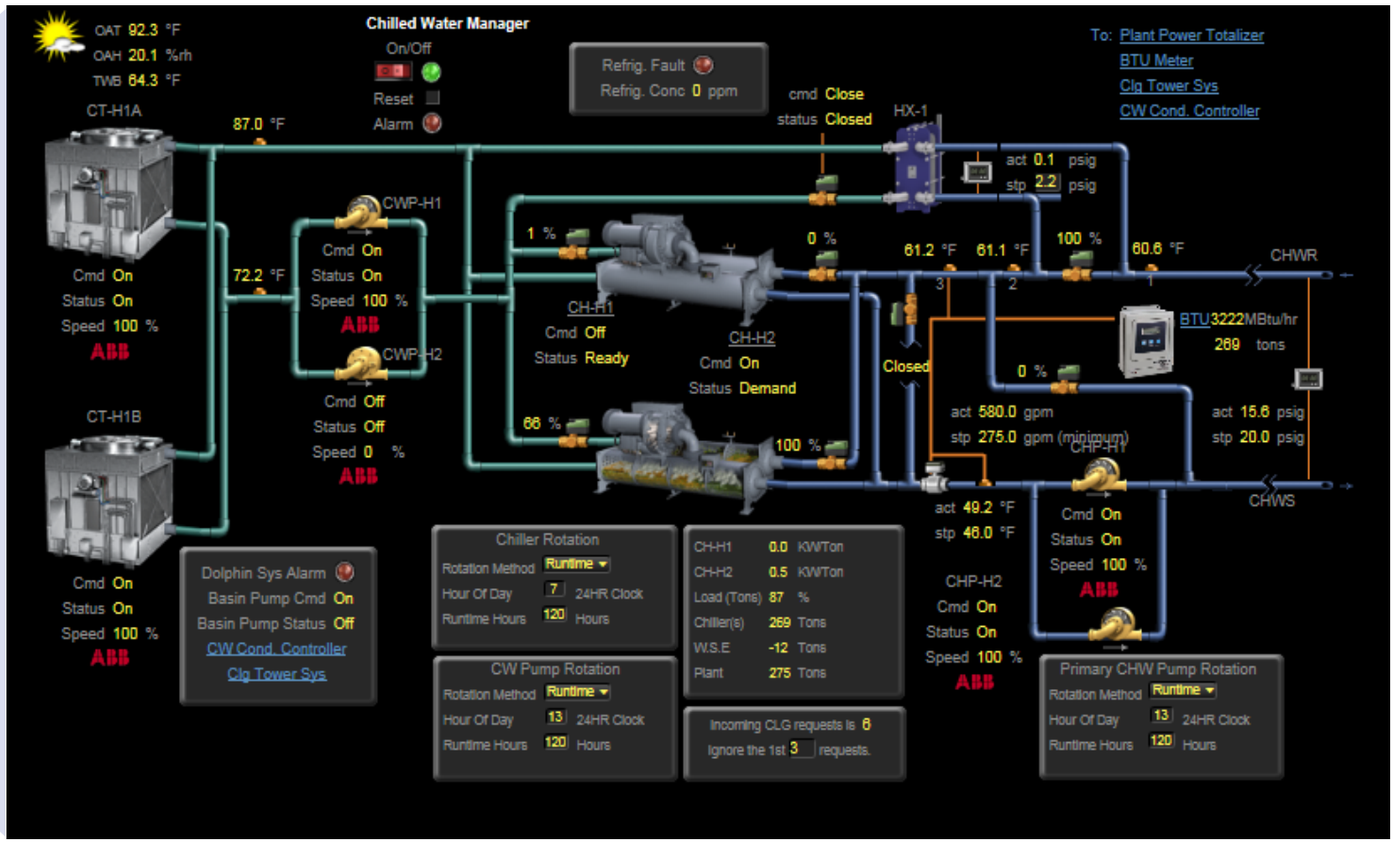
Control design by Taylor Engineering

Plant consists of:

- 2 x 310 ton screw chillers
- 2 x CWP, CHP
- 2 x CT
- 1 x WS economizer HE

Chiller plant specifics:

2 identical scroll chillers, cooling towers and chilled water pumps
Water side economizer



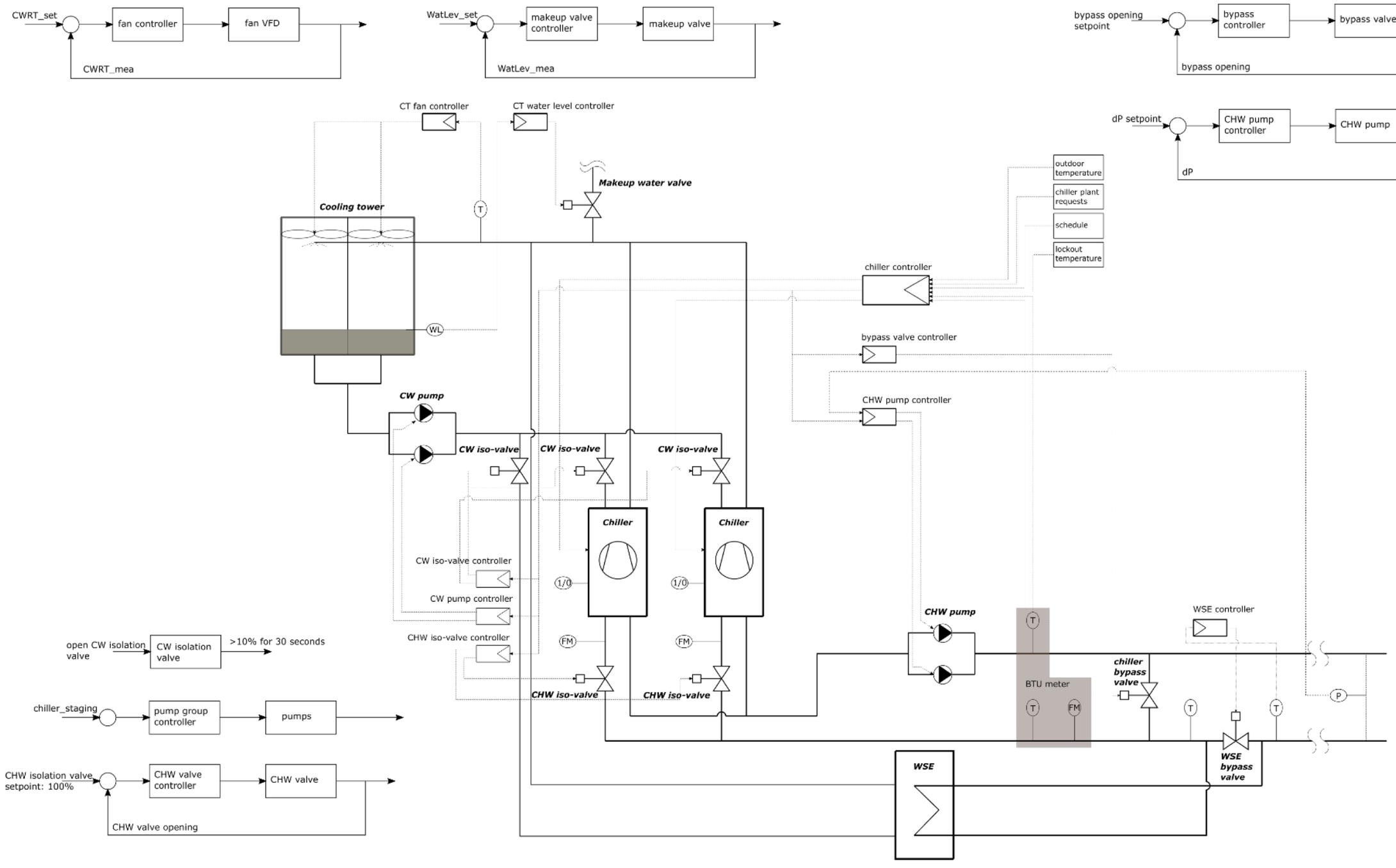
Eikon equipment view

Trend data specs

- ~50 data points
- 1 minute interval data for Jun 22 - July 10 2018
- 5 minute interval data for Mar 11 - Jun 2 2018
- multiple operation stages

Case study II

Plant schematics with a WSE



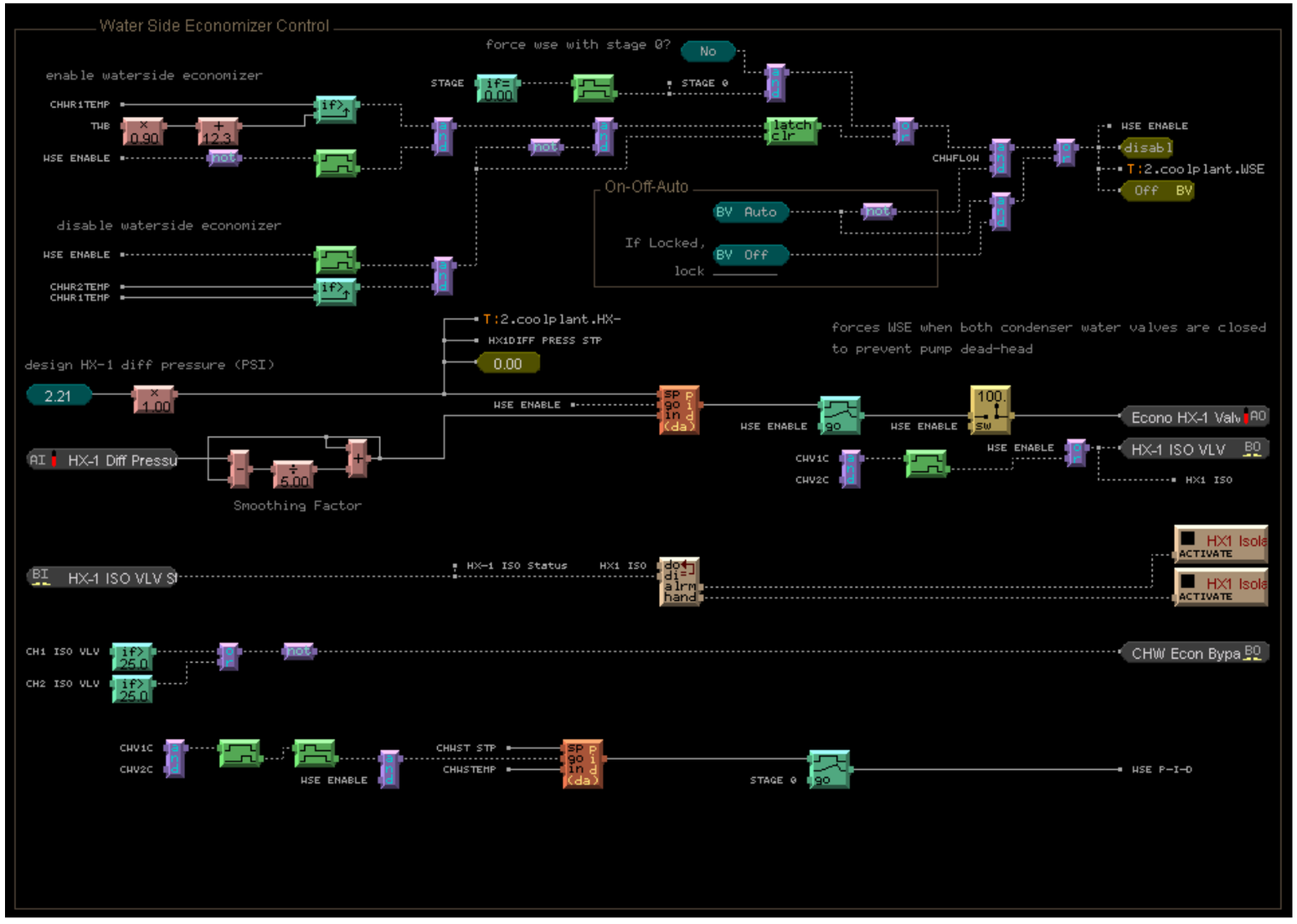
Case study III

Only one sequence or whole plant control?

Approach

- Introduce case study specific edits to primary sequences
- Pick a sequence for conducting the verification test (e.g. WSE control)
- Implement sequence verification with trended data. The method is the same as used for the cooling valve verification example

ALC EIKON implementation of the WSE sequence



Sequence translation tool

"modelica-json": parse control sequences written in Modelica to JSON, and from JSON to other format, such as html, to graphical rendering

- different parsing modes:
  - "cdl": ensure models following cdl syntax
  - "modelica": general modelica syntax

- graphical annotation
  - provide graphical layout for display in block diagram editors (Modelica or actual control platforms)
  - generate graphical diagram for inclusion in documentation (in svg format)
  - render both icon and diagram layer

```
block CustomPWithLimiter
  "Custom implementation of a P controller with variable output limiter"

  parameter Real k = 2 "Constant gain";

  Buildings.Controls.OBC.CDL.Interfaces.RealInput yMax "Maximum value of outp
    annotation (Placement(transformation(extent={{-140,20},{-100,60}})));

  Buildings.Controls.OBC.CDL.Interfaces.RealInput e "Control error"
    annotation (Placement(transformation(extent={{-140,-60},{-100,-20}})));

  Buildings.Controls.OBC.CDL.Interfaces.RealOutput y "Control signal"
    annotation (Placement(transformation(extent={{100,-10},{120,10}})));

  Buildings.Controls.OBC.CDL.Continuous.Gain gain(final k=k) "Constant gain"
    annotation (Placement(transformation(extent={{-60,-50},{-40,-30}})));

  Buildings.Controls.OBC.CDL.Continuous.Min minValue "Outputs the minimum of
    annotation (Placement(transformation(extent={{20,-10},{40,10}})));
equation
  connect(yMax, minValue.u1) annotation (
    Line(points={{-120,40},{-120,40},{-20,40},{-20, 6},{18,6}},
         color={0,0,127}));
  connect(e, gain.u) annotation (
    Line(points={{-120,-40},{-92,-40},{-62,-40}},
         color={0,0,127}));
  connect(gain.y, minValue.u2) annotation (
    Line(points={{-39,-40},{-20,-40},{-20,-6}, {18,-6}},
         color={0,0,127}));
  connect(minValue.y, y) annotation (
     Line(points={{41,0},{110,0}},
          color={0,0,127}));

  annotation (Documentation(info="<html>
<p>
Block that outputs <code>y = min(yMax, k*e)</code>,
where
<code>yMax</code> and <code>e</code> are real-valued input signals and
```

1. FromModelica.CustomPWithLimiter

Custom implementation of a P controller with variable output limiter

1.1. Info

Block that outputs `y = min(yMax, k*e)`, where yMax and e are real-valued inp

1.2. Parameters

It has the following parameters:

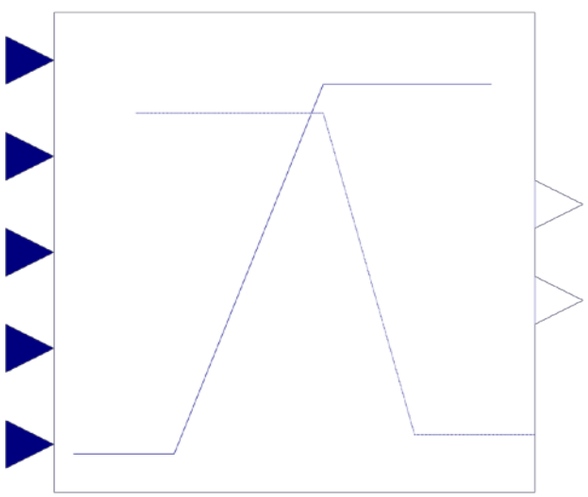
Name	Description	Value	Unit	Display unit	Type	m
General						
Parameters						
k	Constant gain	2			Real	

1.3. Inputs

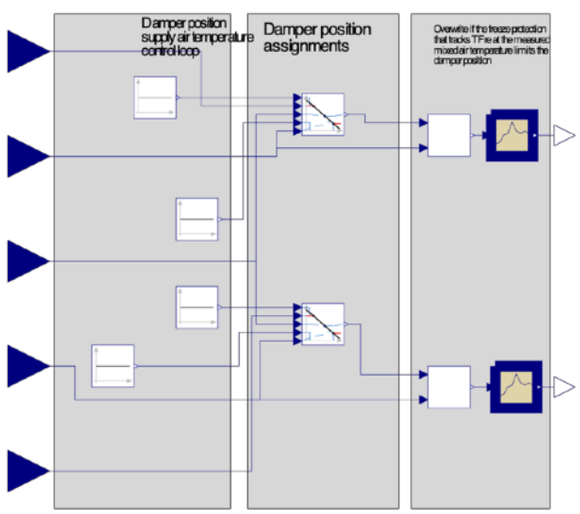
It has the following inputs:

Type	Name	Description	min	max	Uni
Real	yMax	Maximum value of output signal			
Real	e	Control error			

Icon



Diagram



Update about commercialization plan (separate slides)