# OpenBuildingControl
—
# Team meeting

Michael Wetter, Philip Haves,
Jianjun Hu, Milica Grahovac, Lisa Rivalin,
Kun Zhang, Antoine Gautier

March 27, 2019

BERKELEY LAB Lawrence Berkeley National Laboratory

# Content

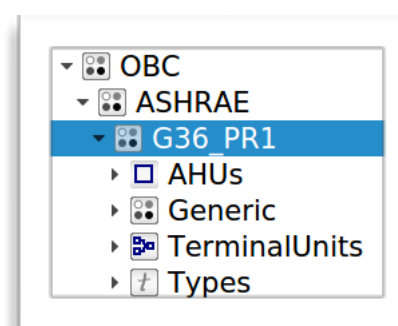
- Primary sequence implementation
- 2nd case study
- Sequence translation
- Verification
- Commercialization plan
- Project status
- Milestones

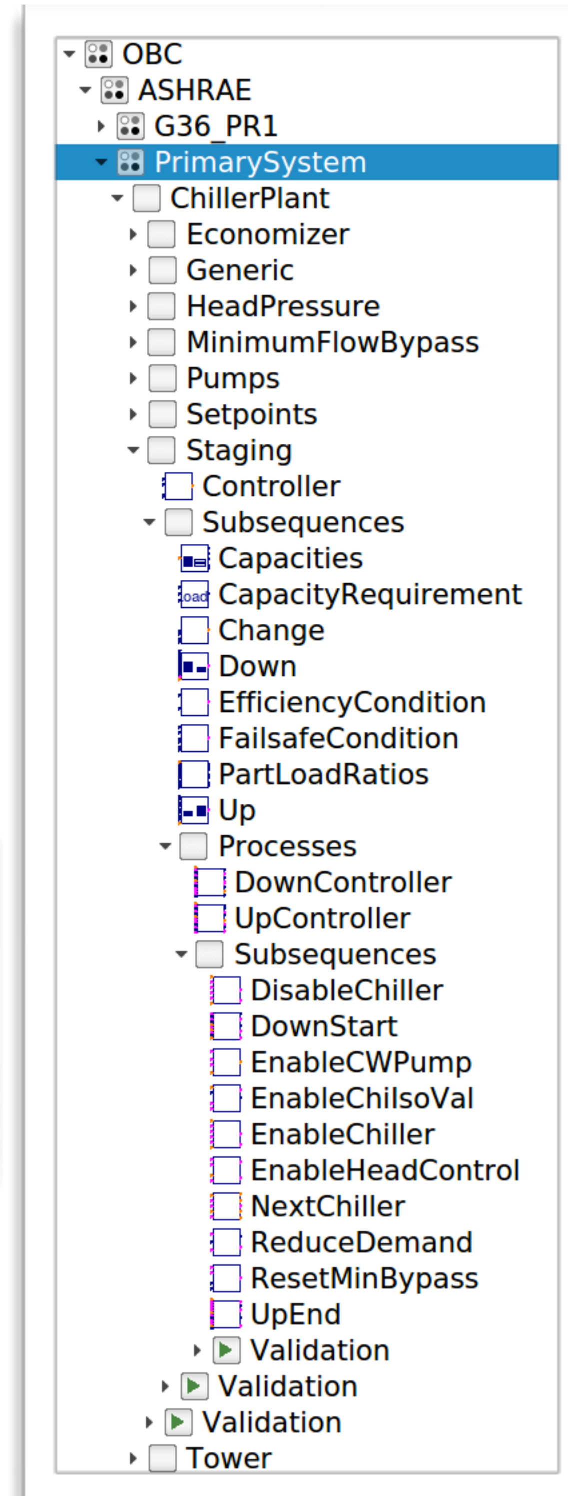
# Primary sequence implementation

Development according to ASHRAE RP-1711
(draft 4, **01/07/2019**) .

When comparing with ASHRAE G36, it has:

- more subsequences:
  - RP-1711 (**150 pages**) vs. G36 (104 pages)
  - `Buildings.Controls.OBC.ASHRAE.G36_PR1`: 26 sequences
  - `Buildings.Controls.OBC.ASHRAE.PrimarySystem`: **>60** sequences
- more combinations: [blue, implemented; grey, not implemented]
  - primary-only / primary-secondary
  - headered / dedicated pumps
  - constant / variable speed pumps
  - with / without pony chiller
  - parallel / series chiller installation
  - with / without waterside economizer



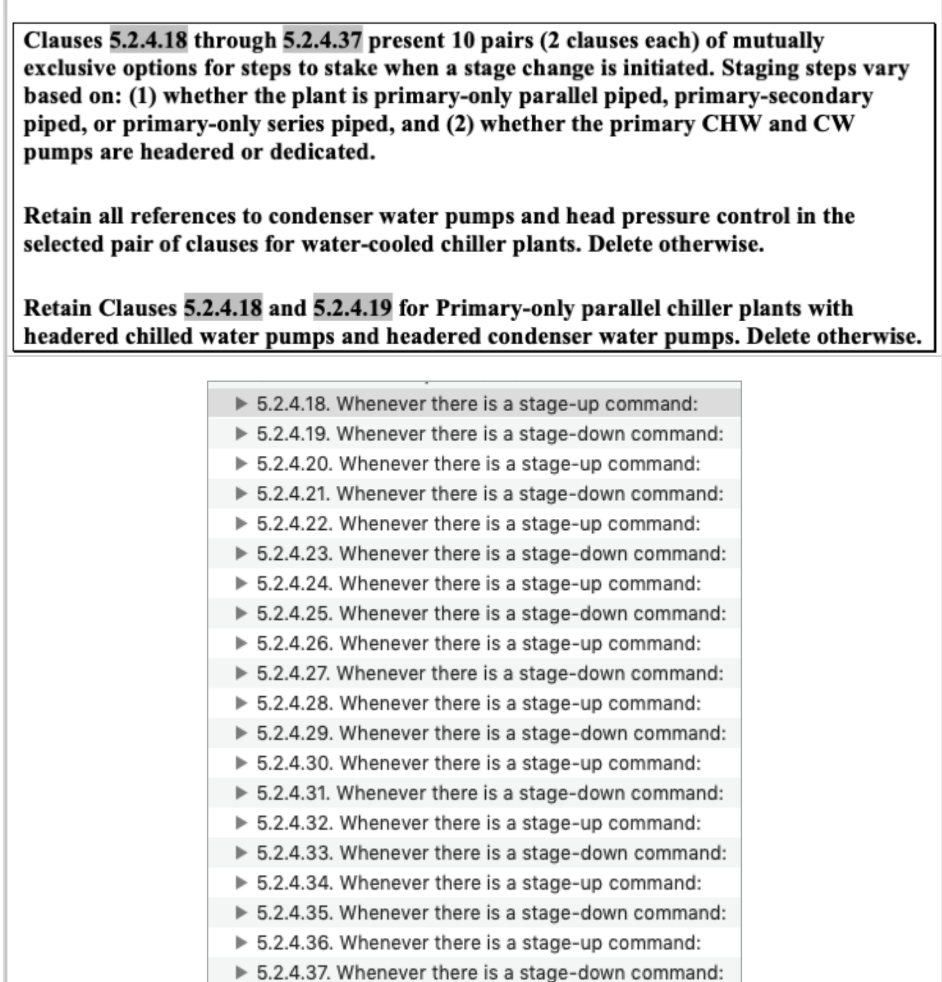



# Primary sequence implementation

Restructure RP-1711 sequences, from type-by-type to modular, object-oriented:

- needed for modular, configurable control implementation
- reduces number of sequences

*structured based on system type*



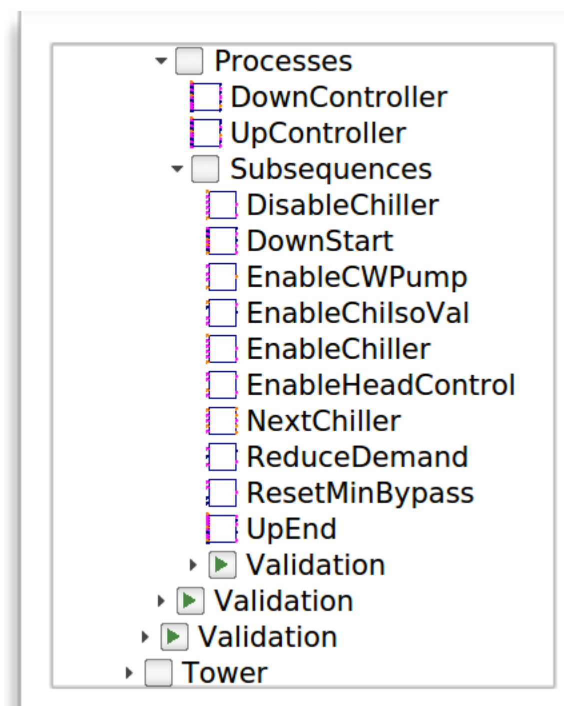
**Clauses 5.2.4.18 through 5.2.4.37 present 10 pairs (2 clauses each) of mutually exclusive options for steps to stake when a stage change is initiated. Staging steps vary based on: (1) whether the plant is primary-only parallel piped, primary-secondary piped, or primary-only series piped, and (2) whether the primary CHW and CW pumps are headered or dedicated.**

**Retain all references to condenser water pumps and head pressure control in the selected pair of clauses for water-cooled chiller plants. Delete otherwise.**

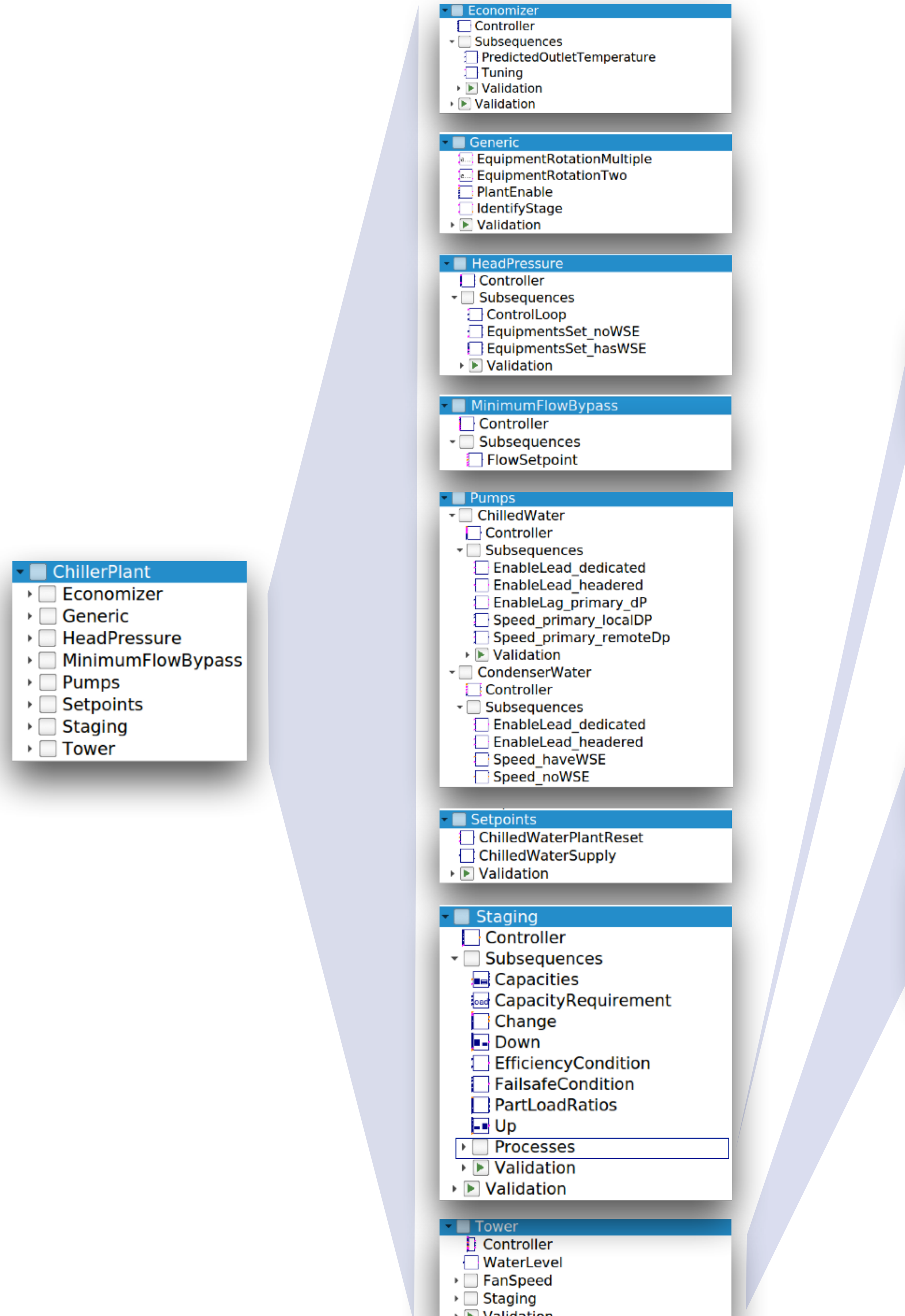
**Retain Clauses 5.2.4.18 and 5.2.4.19 for Primary-only parallel chiller plants with headered chilled water pumps and headered condenser water pumps. Delete otherwise.**

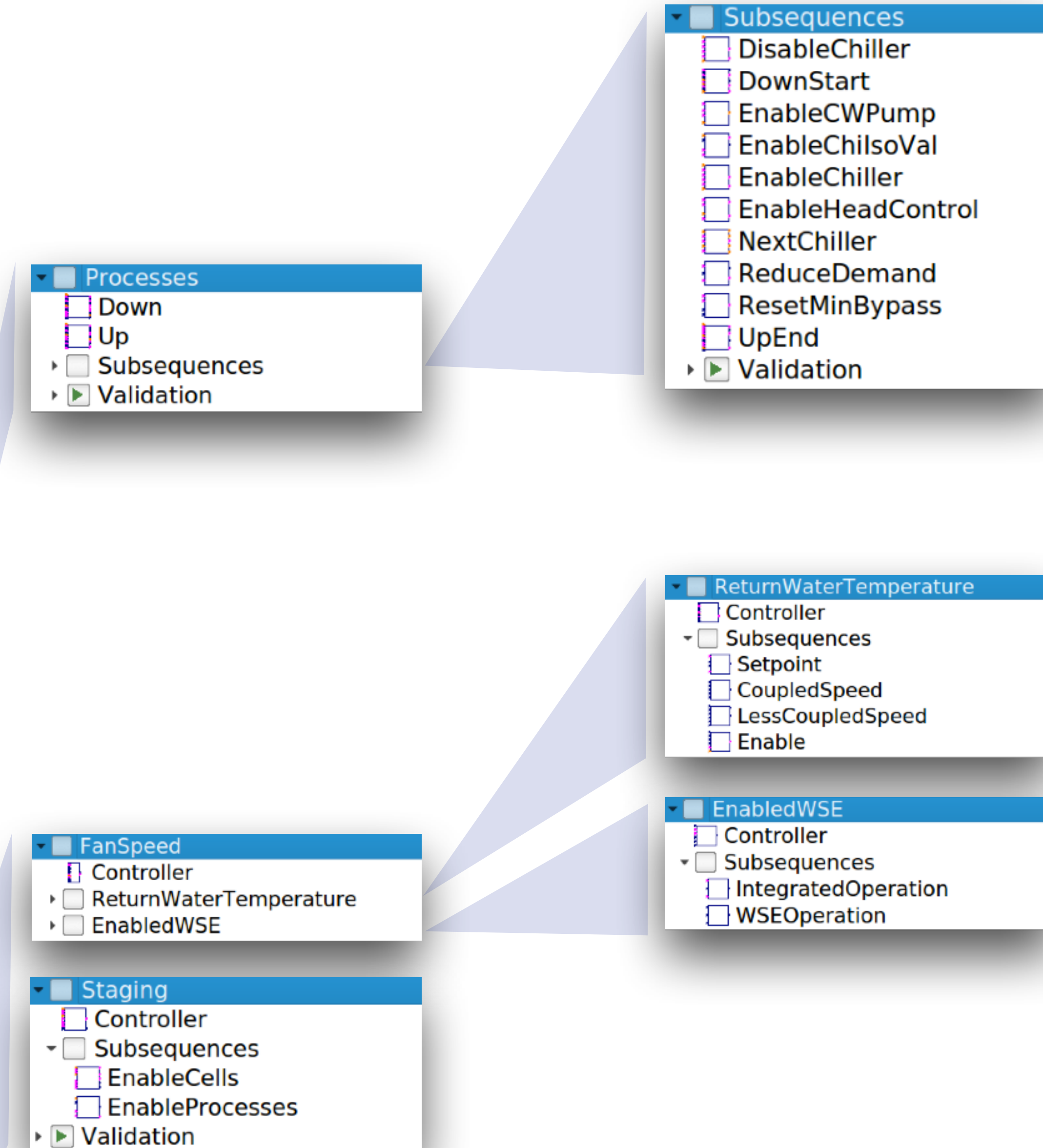
- 5.2.4.18. Whenever there is a stage-up command:
- 5.2.4.19. Whenever there is a stage-down command:
- 5.2.4.20. Whenever there is a stage-up command:
- 5.2.4.21. Whenever there is a stage-down command:
- 5.2.4.22. Whenever there is a stage-up command:
- 5.2.4.23. Whenever there is a stage-down command:
- 5.2.4.24. Whenever there is a stage-up command:
- 5.2.4.25. Whenever there is a stage-down command:
- 5.2.4.26. Whenever there is a stage-up command:
- 5.2.4.27. Whenever there is a stage-down command:
- 5.2.4.28. Whenever there is a stage-up command:
- 5.2.4.29. Whenever there is a stage-down command:
- 5.2.4.30. Whenever there is a stage-up command:
- 5.2.4.31. Whenever there is a stage-down command:
- 5.2.4.32. Whenever there is a stage-up command:
- 5.2.4.33. Whenever there is a stage-down command:
- 5.2.4.34. Whenever there is a stage-up command:
- 5.2.4.35. Whenever there is a stage-down command:
- 5.2.4.36. Whenever there is a stage-up command:
- 5.2.4.37. Whenever there is a stage-down command:

*restructured based on functionality*



# Primary sequence implementation




## Ongoing:

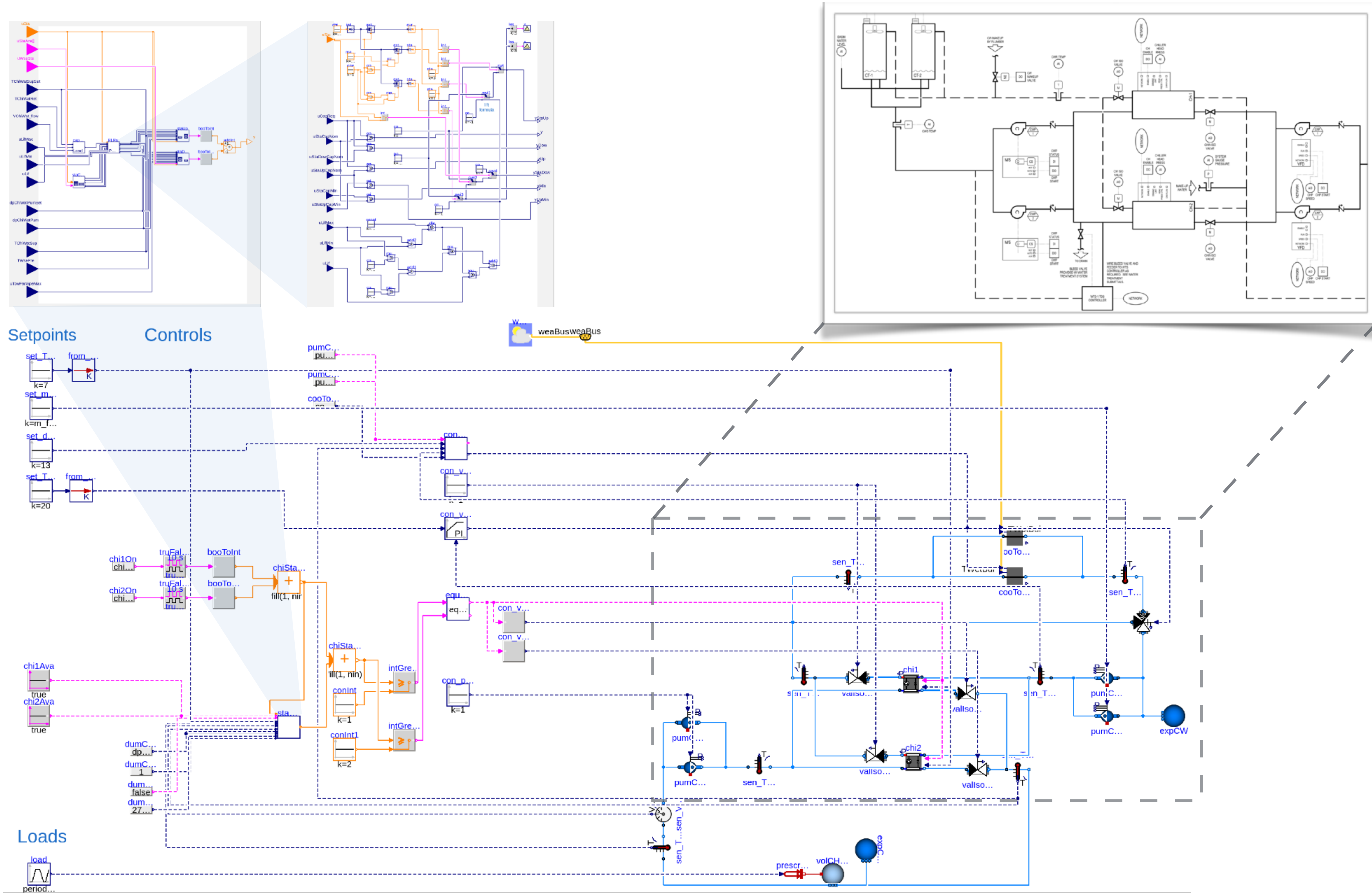
- reviewing subsequences

  https://github.com/lbl-srg/modelica-buildings/tree/issue1378_staging_primarySequences

- integration in top-level controller
- closed loop testing

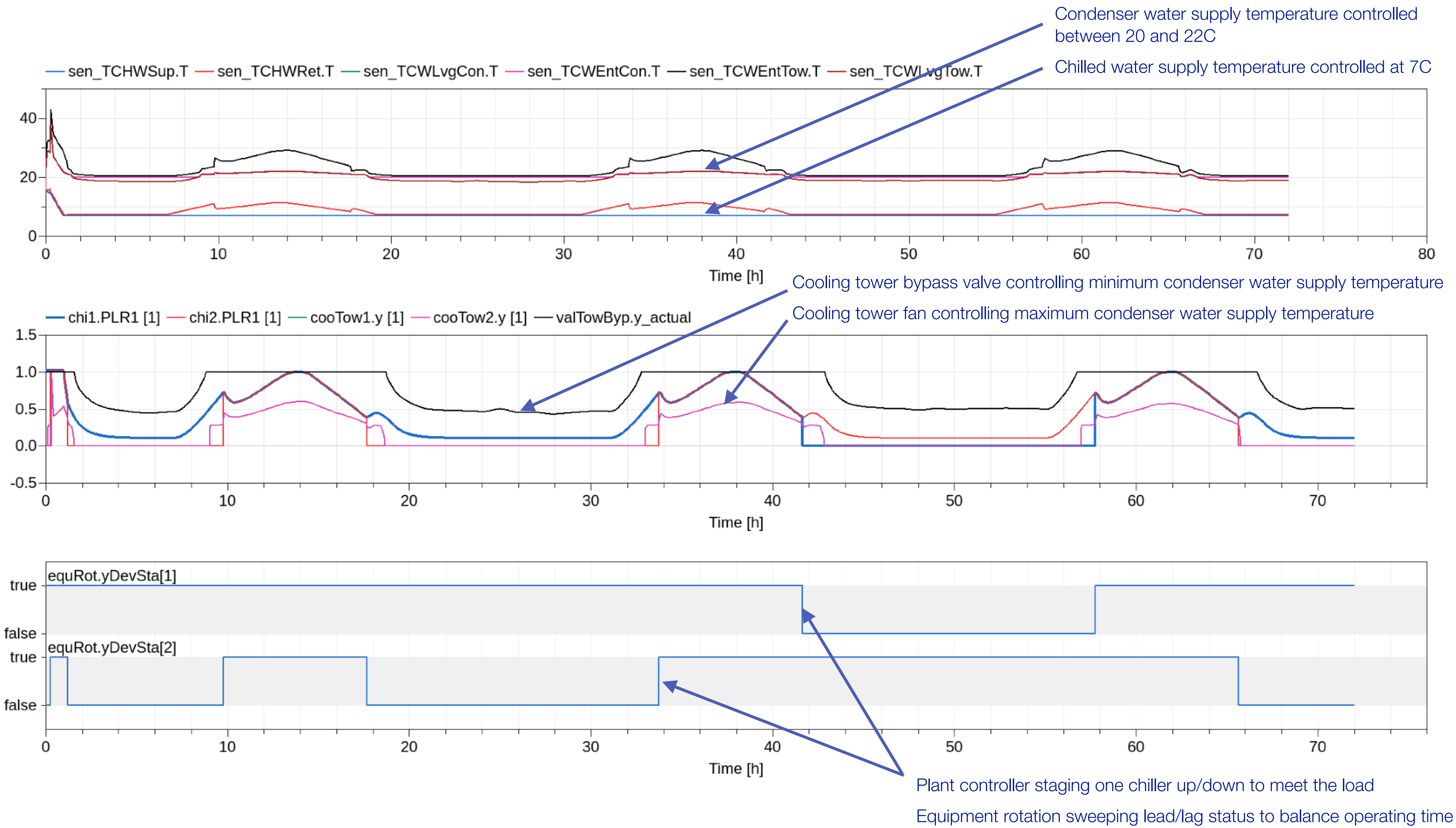
# Validation on closed loop model has started

Base case: parallel chillers, constant primary chilled water loop loaded with prescribed heat flow rate, constant condenser water loop with cooling towers



# Validation on closed loop model

## Simulation of staging and rotating sequences



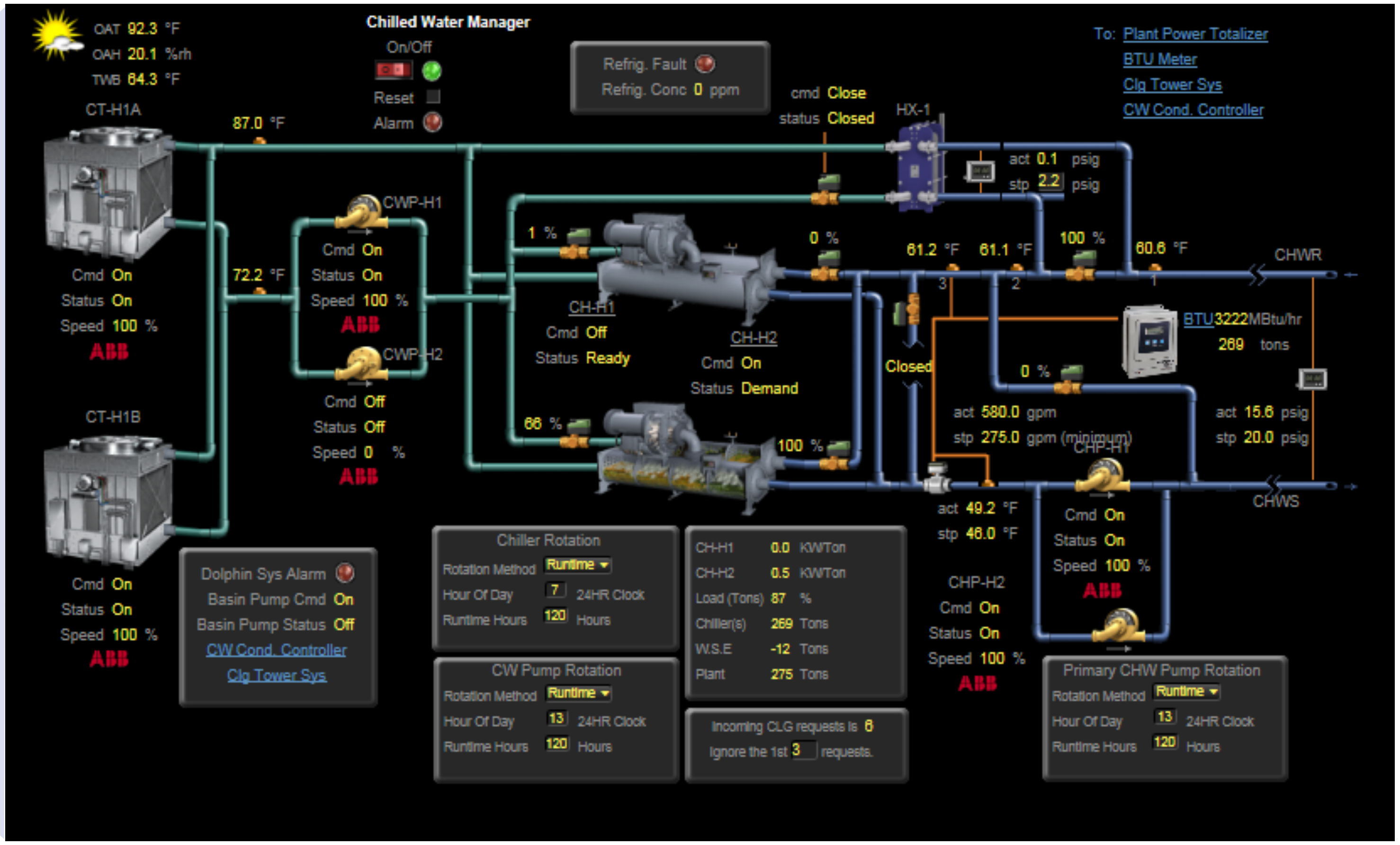
# Case study primary plant

Plant installed 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



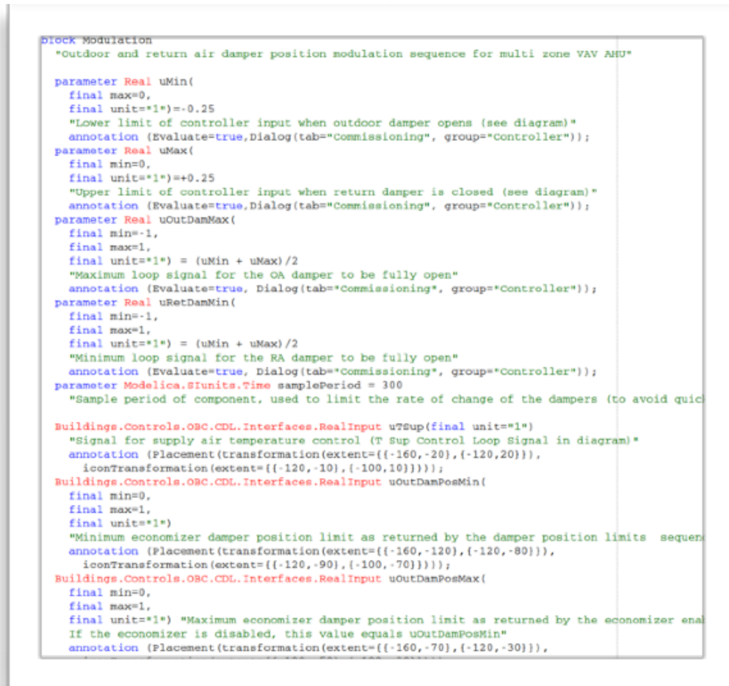

Eikon equipment view

## 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

Update: On hold as it requires the chiller sequences that are now being implemented.

# Sequence translation tool: modelica-json parser

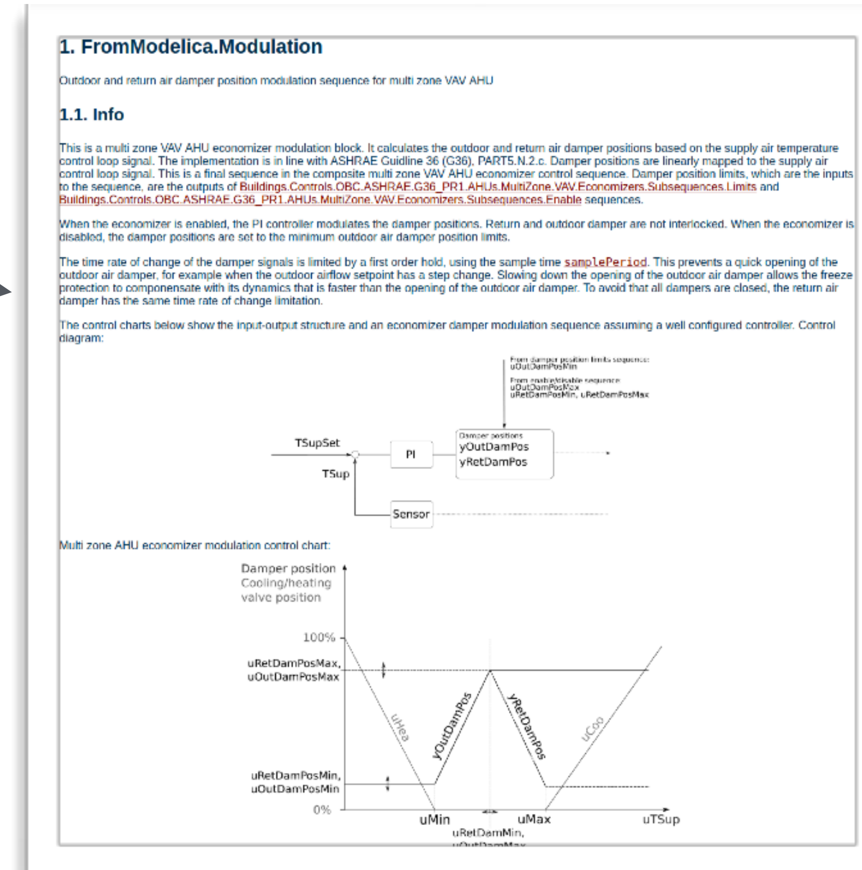


```
block Modulation
  "Outdoor and return air damper position modulation sequence for multi zone VAV AHU"

  parameter Real uMin(
    final max=0,
    final unit="1")=-0.25
    "Lower limit of controller input when outdoor damper opens (see diagram)"
    annotation (Evaluate=true,Dialog(tab="Commissioning", group="Controller"));
  parameter Real uMax(
    final min=0,
    final unit="1")=+0.25
    "Upper limit of controller input when return damper is closed (see diagram)"
    annotation (Evaluate=true,Dialog(tab="Commissioning", group="Controller"));
  parameter Real uOutDamMax(
    final min=-1,
    final max=1,
    final unit="1") = (uMin + uMax)/2
    "Maximum loop signal for the OA damper to be fully open"
    annotation (Evaluate=true, Dialog(tab="Commissioning", group="Controller"));
  parameter Real uRetDamMin(
    final min=-1,
    final max=1,
    final unit="1") = (uMin + uMax)/2
    "Minimum loop signal for the RA damper to be fully open"
    annotation (Evaluate=true, Dialog(tab="Commissioning", group="Controller"));
  parameter Modelica.SIunits.Time samplePeriod = 300
    "Sample period of component, used to limit the rate of change of the dampers (to avoid quic

  Buildings.Controls.OBC.CDL.Interfaces.RealInput uTSup(final unit="1")
    "Signal for supply air temperature control (T Sup Control Loop Signal in diagram)"
    annotation (Placement(transformation(extent={{-160,-20},{-120,20}}),
      iconTransformation(extent={{-120,-10},{-100,10}})));
  Buildings.Controls.OBC.CDL.Interfaces.RealInput uOutDamPosMin(
    final min=0,
    final max=1,
    final unit="1")
    "Minimum economizer damper position limit as returned by the damper position limits  sequen
    annotation (Placement(transformation(extent={{-160,-120},{-120,-80}}),
      iconTransformation(extent={{-120,-90},{-100,-70}})));
  Buildings.Controls.OBC.CDL.Interfaces.RealInput uOutDamPosMax(
    final min=0,
    final max=1,
    final unit="1") "Maximum economizer damper position limit as returned by the economizer ena
    If the economizer is disabled, this value equals uOutDamPosMin"
    annotation (Placement(transformation(extent={{-160,-70},{-120,-30}}),
```

```
[
 {
  "modelicaFile": "./Modulation.mo",
  "within": "FromModelica",
  "topClassName": "FromModelica.Modulation",
  "comment": "Outdoor and return air damper position modulation sequence for multi zone VAV AHU",
  "public": {
   "parameters": [
    {
     "className": "Real",
     "name": "uMin",
     "value": "-0.25",
     "comment": "Lower limit of controller input when outdoor damper opens (see diagram)",
     "unit": {
      "prefix": "final",
      "value": "\"1\""
     },
     "max": {
      "prefix": "final",
      "value": "0"
     },
     "modifications": [
      {
       "value": "0",
       "isFinal": true
      },
      {
       "value": "\"1\"",
       "isFinal": true
      }
     ],
```



## 1. FromModelica.Modulation

Outdoor and return air damper position modulation sequence for multi zone VAV AHU

### 1.1. Info

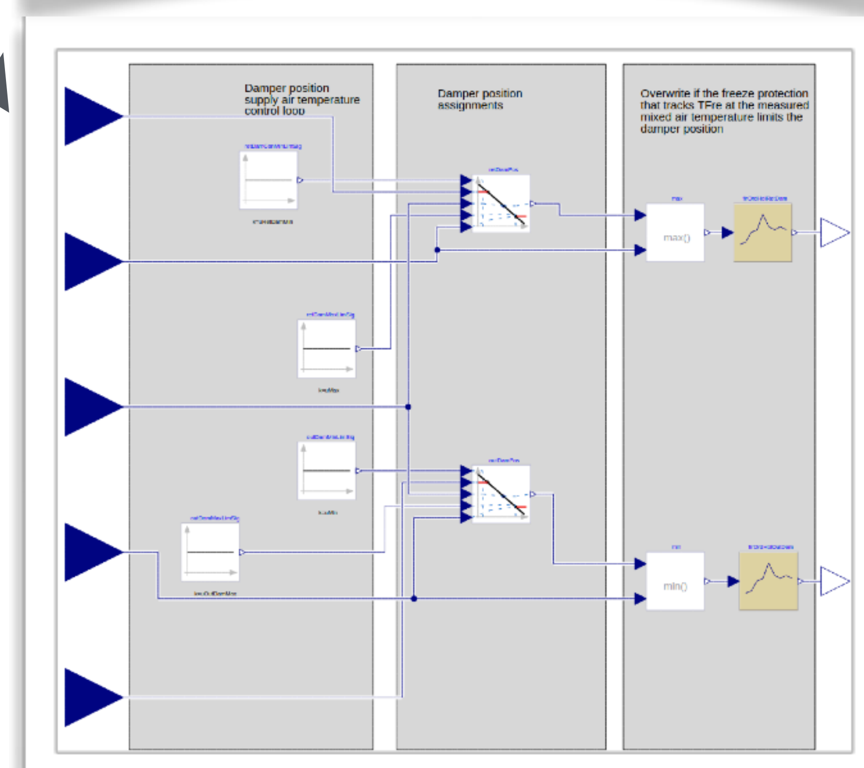
This is a multi zone VAV AHU economizer modulation block. It calculates the outdoor and return air damper positions based on the supply air temperature control loop signal. The implementation is in line with ASHRAE Guidline 36 (G36), PART5.N.2.c. Damper positions are linearly mapped to the supply air control loop signal. This is a final sequence in the composite multi zone VAV AHU economizer control sequence. Damper position limits, which are the inputs to the sequence, are the outputs of Buildings.Controls.OBC.ASHRAE.G36_PR1.AHUs.MultiZone.VAV.Economizers.Subsequences.Limits and Buildings.Controls.OBC.ASHRAE.G36_PR1.AHUs.MultiZone.VAV.Economizers.Subsequences.Enable sequences.

When the economizer is enabled, the PI controller modulates the damper positions. Return and outdoor damper are not interlocked. When the economizer is disabled, the damper positions are set to the minimum outdoor air damper position limits.

The time rate of change of the damper signals is limited by a first order hold, using the sample time samplePeriod. This prevents a quick opening of the outdoor air damper, for example when the outdoor airflow setpoint has a step change. Slowing down the opening of the outdoor air damper allows the freeze protection to componensate with its dynamics that is faster than the opening of the outdoor air damper. To avoid that all dampers are closed, the return air damper has the same time rate of change limitation.

The control charts below show the input-output structure and an economizer damper modulation sequence assuming a well configured controller. Control diagram:






```
{
 "$schema": "http://json-schema.org/draft-07/schema#",
 "$id": "schema-CDL.json",
 "type": "array",
 "title": "JSON parsed Modelica file",
 "description": "Json representation of a Modelica model",
 "definitions": {
  "coordinates": {
   "description": "Coordinates on a 2d plan",
   "type": "array",
   "additionalItems": false,
   "items": {
    "type": "object",
    "required": [
     "x1",
     "x2"
    ],
    "additionalProperties": false,
    "properties": {
     "x1": {
      "type": "number"
     },
     "x2": {
      "type": "number"
     }
    }
   }
  },
  "rgbcolors": {
   "type": "string",
   "oneOf": [
    {
     "pattern": "^(rgb[(]|[0-9]{1,3},[0-9]{1,3},[0-9]{1,3}[)]$"
    },
    {
```

Parse modelica models and/or control sequences

Output formats:

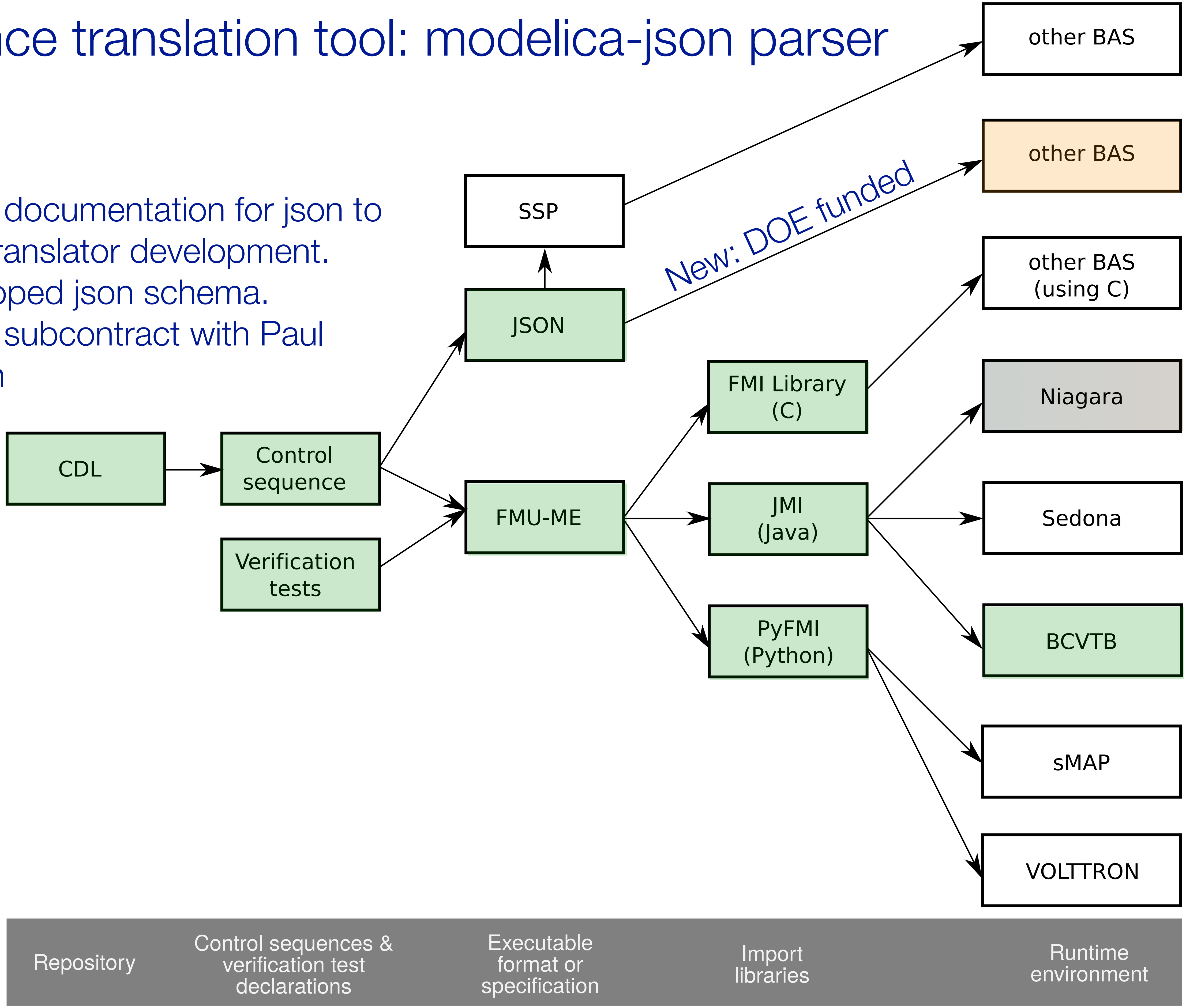
- json intermediate format
- validate format by JSON schema
- svg for graphics
- html & svg for documentation
- 2019/20: prototype translator to a commercial control product line
- 2020+: Basis for HVAC & control design tool

https://github.com/lbl-srg/modelica-json

# Sequence translation tool: modelica-json parser

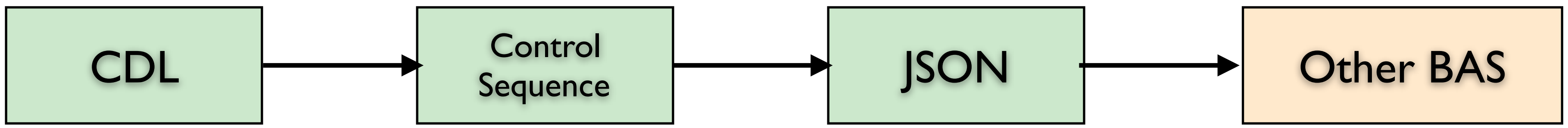
Update:

- Wrote documentation for json to BAS translator development.
- Developed json schema.
- Setup subcontract with Paul Ehrlich




http://obc.lbl.gov/specification/codeGeneration.html#use-of-control-sequences-or-verification-tests-in-realtime-applications

# From JSON to a control product line…




Modelica-json package offers :

- A JSON Schema for JSON representation of CDL sequences  that describes
  - ➡ The data structure
  - ➡ Required or optional properties
  - ➡ Constraints on values and expected patterns
- A Validation script to test the compliance of a JSON file to the Schema

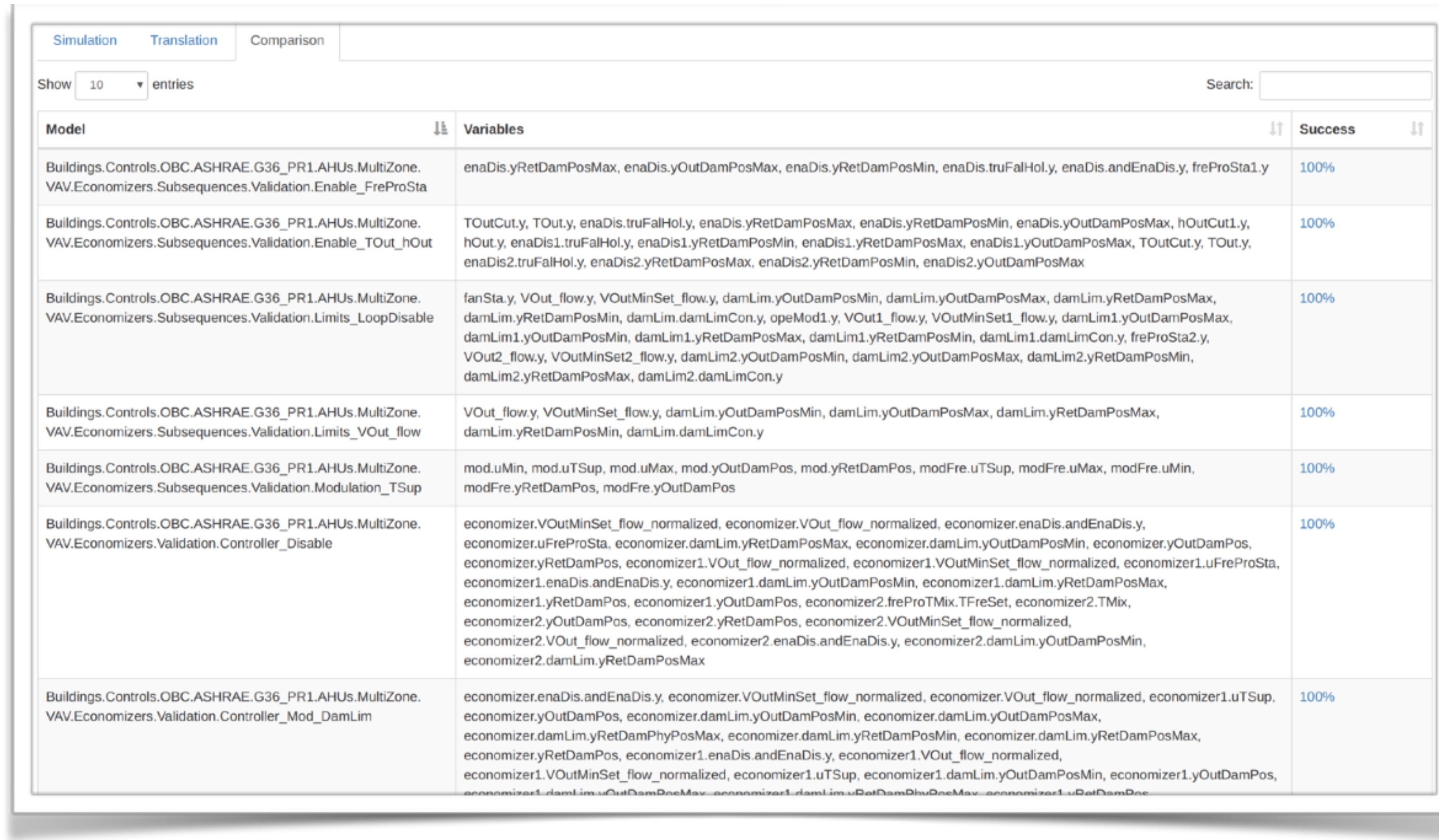
- Contractors can use the JSON Schema as a specification to develop a translator to a control product line.
- If JSON Files are the starting point, then they must validate against the schema.
- The detailed schema can be found here

# Control verification tool: `funnel`

A cross-platform C-based software for comparing two (x, y) data sets given tolerances along x and y directions

- Validation of control sequences by comparing time series from real operation vs simulation
- Main principles and features:
  - Available as a Python module with HTML interactive plot for enhanced error analysis
  - To be released: multiple plots and HTML summary report

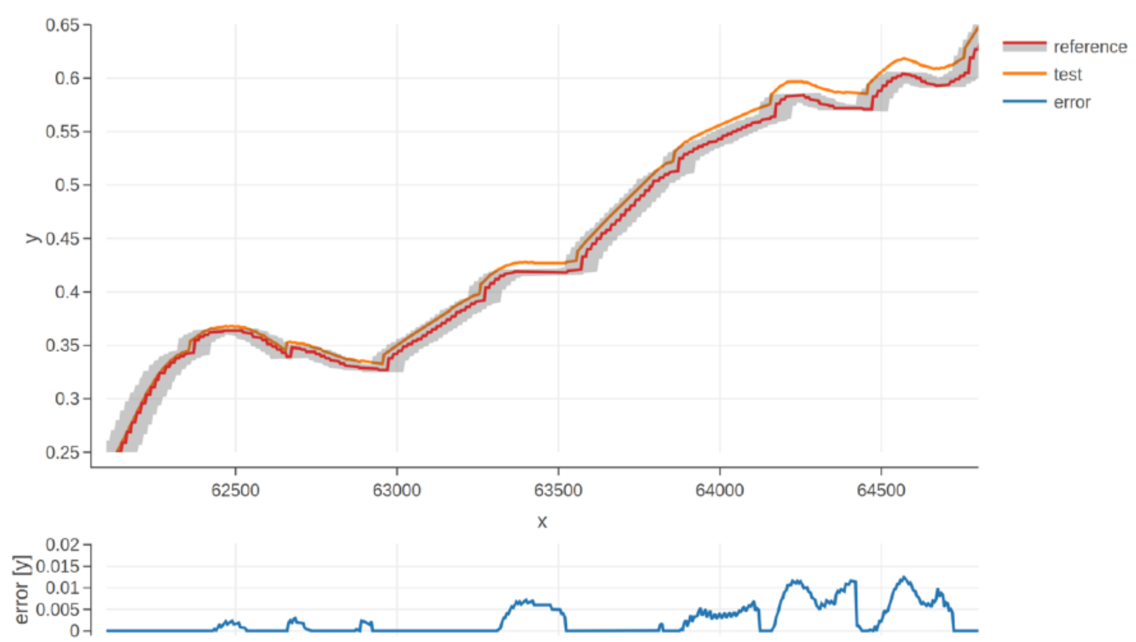
(https://github.com/lbl-srg/funnel)

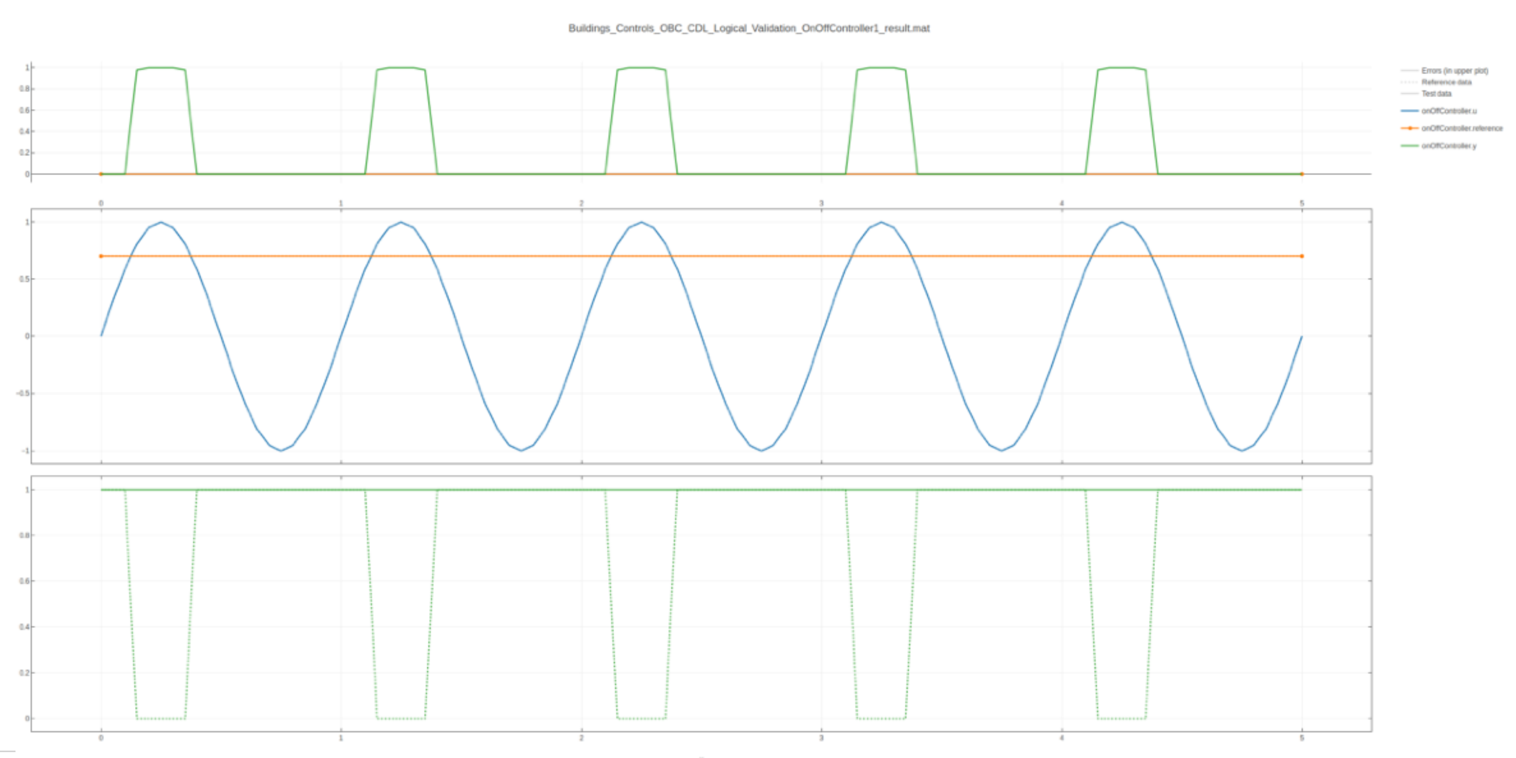


Simulation Translation Comparison

Show 10 entries Search:

| Model | Variables | Success |
|---|---|---|
| Buildings.Controls.OBC.ASHRAE.G36_PR1.AHUs.MultiZone.VAV.Economizers.Subsequences.Validation.Enable_FreProSta | enaDis.yRetDamPosMax, enaDis.yOutDamPosMax, enaDis.yRetDamPosMin, enaDis.truFalHol.y, enaDis.andEnaDis.y, freProSta1.y | 100% |
| Buildings.Controls.OBC.ASHRAE.G36_PR1.AHUs.MultiZone.VAV.Economizers.Subsequences.Validation.Enable_TOut_hOut | TOutCut.y, TOut.y, enaDis.truFalHol.y, enaDis.yRetDamPosMax, enaDis.yRetDamPosMin, enaDis.yOutDamPosMax, hOutCut1.y, hOut.y, enaDis1.truFalHol.y, enaDis1.yRetDamPosMin, enaDis1.yRetDamPosMax, enaDis1.yOutDamPosMax, TOutCut.y, TOut.y, enaDis2.truFalHol.y, enaDis2.yRetDamPosMax, enaDis2.yRetDamPosMin, enaDis2.yOutDamPosMax | 100% |
| Buildings.Controls.OBC.ASHRAE.G36_PR1.AHUs.MultiZone.VAV.Economizers.Subsequences.Validation.Limits_LoopDisable | fanSta.y, VOut_flow.y, VOutMinSet_flow.y, damLim.yOutDamPosMin, damLim.yOutDamPosMax, damLim.yRetDamPosMax, damLim.yRetDamPosMin, damLim.damLimCon.y, opeMod1.y, VOut1_flow.y, VOutMinSet1_flow.y, damLim1.yOutDamPosMax, damLim1.yOutDamPosMin, damLim1.yRetDamPosMax, damLim1.yRetDamPosMin, damLim1.damLimCon.y, freProSta2.y, VOut2_flow.y, VOutMinSet2_flow.y, damLim2.yOutDamPosMin, damLim2.yOutDamPosMax, damLim2.yRetDamPosMin, damLim2.yRetDamPosMax, damLim2.damLimCon.y | 100% |
| Buildings.Controls.OBC.ASHRAE.G36_PR1.AHUs.MultiZone.VAV.Economizers.Subsequences.Validation.Limits_VOut_flow | VOut_flow.y, VOutMinSet_flow.y, damLim.yOutDamPosMin, damLim.yOutDamPosMax, damLim.yRetDamPosMax, damLim.yRetDamPosMin, damLim.damLimCon.y | 100% |
| Buildings.Controls.OBC.ASHRAE.G36_PR1.AHUs.MultiZone.VAV.Economizers.Subsequences.Validation.Modulation_TSup | mod.uMin, mod.uTSup, mod.uMax, mod.yOutDamPos, mod.yRetDamPos, modFre.uTSup, modFre.uMax, modFre.uMin, modFre.yRetDamPos, modFre.yOutDamPos | 100% |
| Buildings.Controls.OBC.ASHRAE.G36_PR1.AHUs.MultiZone.VAV.Economizers.Validation.Controller_Disable | economizer.VOutMinSet_flow_normalized, economizer.VOut_flow_normalized, economizer.enaDis.andEnaDis.y, economizer.uFreProSta, economizer.damLim.yRetDamPosMax, economizer.damLim.yOutDamPosMin, economizer.yOutDamPos, economizer.yRetDamPos, economizer1.VOut_flow_normalized, economizer1.VOutMinSet_flow_normalized, economizer1.uFreProSta, economizer1.enaDis.andEnaDis.y, economizer1.damLim.yOutDamPosMin, economizer1.damLim.yRetDamPosMax, economizer1.yRetDamPos, economizer1.yOutDamPos, economizer2.freProTMix.TFreSet, economizer2.TMix, economizer2.yOutDamPos, economizer2.yRetDamPos, economizer2.VOutMinSet_flow_normalized, economizer2.VOut_flow_normalized, economizer2.enaDis.andEnaDis.y, economizer2.damLim.yOutDamPosMin, economizer2.damLim.yRetDamPosMax | 100% |
| Buildings.Controls.OBC.ASHRAE.G36_PR1.AHUs.MultiZone.VAV.Economizers.Validation.Controller_Mod_DamLim | economizer.enaDis.andEnaDis.y, economizer.VOutMinSet_flow_normalized, economizer.VOut_flow_normalized, economizer1.uTSup, economizer.yOutDamPos, economizer.damLim.yOutDamPosMin, economizer.damLim.yOutDamPosMax, economizer.damLim.yRetDamPhyPosMax, economizer.damLim.yRetDamPosMin, economizer.damLim.yRetDamPosMax, economizer.yRetDamPos, economizer1.enaDis.andEnaDis.y, economizer1.VOut_flow_normalized, economizer1.VOutMinSet_flow_normalized, economizer1.uTSup, economizer1.damLim.yOutDamPosMin, economizer1.yOutDamPos, | 100% |

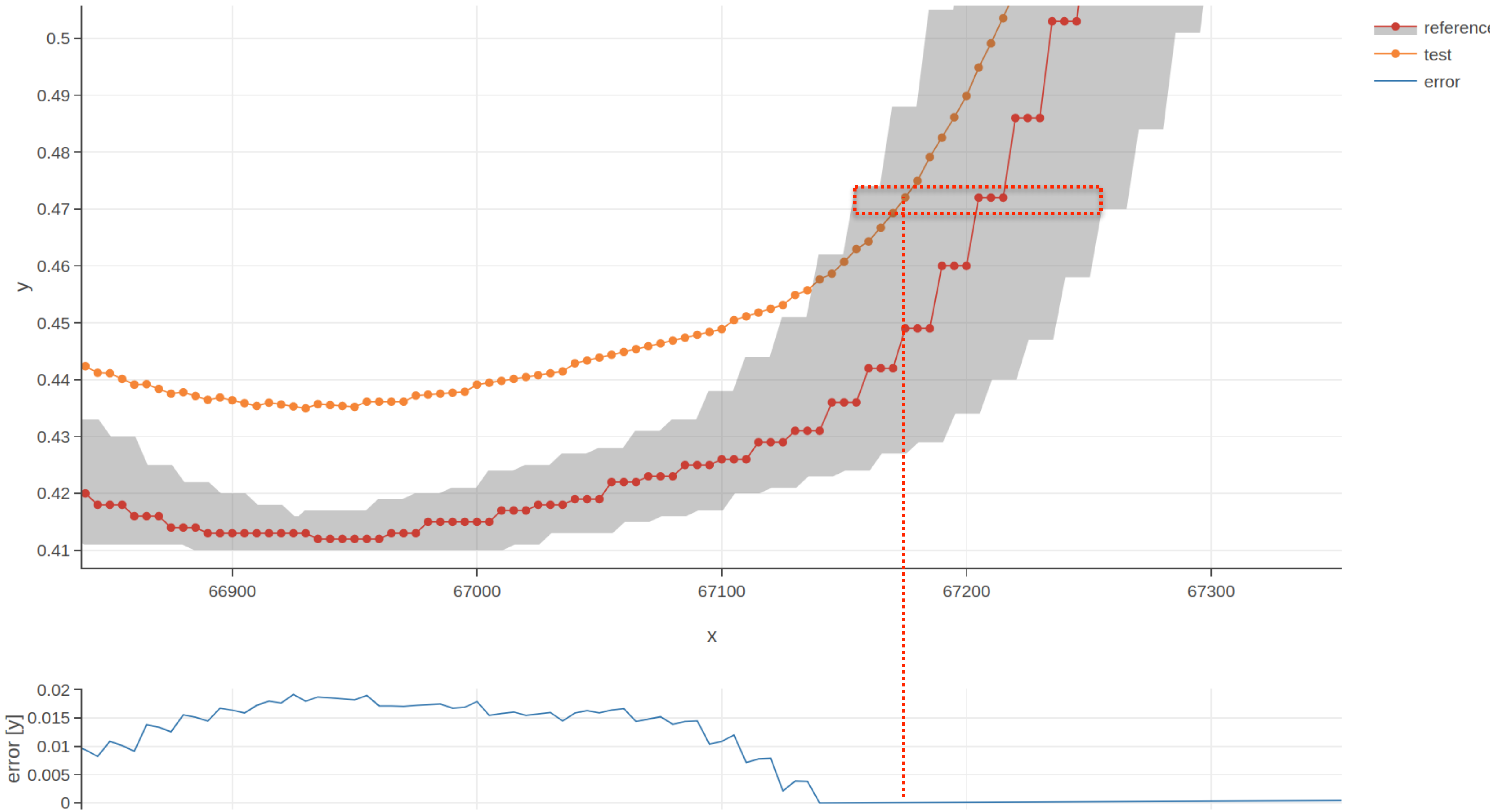


# Control verification tool: `funnel`

## Detailed principles

- L1-norm based comparison
- Trajectory comparison (as opposed to point-to-point): handles time events & different time scales



# Update CDL

- Created block: `Utilities.SunRiseSet`
- Working on new block: `Utilities.OptimalStart`

**Commercial controller (Carrier, Trane, Honeywell)**

- Temperature gradient Tg
  - measurable, unit = C/hr, F/hr

$$t_{opt} = \left| \frac{T_{setpoint} - T_{zone}}{T_g} \right|$$

  - separate Tg for preheat and precool
  - Preheat: temperature increase rate
  - Precool: temperature decrease rate
- Record and update
  - calculate Tg based on simulation results for each day
  - update Tg based on results from previous days

$$T_{g,i} = mean(T_{g,i-1}, T_{g,i-2}, ...)$$

**ASHRAE handbook**

- Regression
- Precool

$$t_{opt} = a_0 + a_1 T_{zone} + a_2 T_{zone}^2$$

- Preheat

$$t_{opt} = b_0 + (1 - \omega)(b_1 T_{zone} + b_2 T_{zone}^2 + \omega a_3 T_{out})$$

$$\omega = 1000^{-(T_{zone} - T_{sp,occ})/(T_{sp,occ} - T_{sp,unocc})}$$

- Update regression coefficients (a0, b0,...) based on simulation results from previous days

Additional option: optimal stop

# Commercialization plan

- Benefits for stakeholders ranging from owners to control providers, mechanical designers and utilities.
- Future work
  - Market transformation, inform industry
  - Sequence selection tool
  - System design tool
  - Library expansion
  - Control vendor support
  - Standardization (such as ASHRAE/ANSI)
  - Semantic tagging

Update:

- Finished and submitted to DOE and CEC draft commercialization plan.

# Project status

Submitted to CEC change of scope. Currently in CEC management approval process.

Extended DOE period of performance to 9/31/2020 due to delay in CEC cost-share and staffing shortage. (Phone call Feb. 12, 2019).

CEC Agreement Term is until 12/30/2020 (remains unchanged).

- Move draft final report from 10/18/19 to 5/1/20
- Move final report from 11/2/2019 to 8/1/20

Past spending was conservative, assuming CEC cost-share does not materialize.

Measured by spending, we are now at deliverables of August 2018 (e.g., Q8)

# Milestone and progress

**Task 2.3**: By Q7, release a version of the control library for primary systems, facade and lighting.
— Primary system is ongoing, facade is completed, lighting perceived as not important by TAG. [Note: For Spawn, we will add an example for lighting control.]

**Task 2**: By Q7, coordinate with NREL to architect the control design tool as part of the OpenStudio framework.
— Coordination done, design specification in development.

**Task 2**: By Q8, demonstrate importing and exporting CDL in the control design tool.
— Completed (through Dymola and OpenModelica, and export through modelica-json)

**Task 4**: By Q8, write case study report as above, but for primary HVAC system.
— Delayed, data collected, implementation of control ongoing; expect completion in June.

**Task 5**: By Q8, write first version of commercialization and market transformation plan.
— Completed.

**Task 5:** By Q9, hold an ASHRAE seminar or forum about controls design and verification to aid dissemination and get feedback from the ASHRAE community.
— Seminar at BS'19, Haystack Connect. Presentation given to ASHRAE Guideline 36 Committee in Jan. 2019