D5.4	Benchmark building and energy system models
Access[1]:	**PU**
Type[2]:	**Report**
Version:	**1.9**
Due Dates[3]:	**M24**

Open Cyber-Physical System Model-Driven Certified Development

Executive summary[4]:

The purpose of this work is to develop a suite of benchmark test cases that capture essential performance differences between Modelica and FMU tools with respect to typical problem categories in building performance simulation. A brief background of the field is given, along with an account of different modelling and simulation approaches.

Each case has been simulated with more than one tool. EQUA's IDA SE has been compared with Simulink, Dymola and OpenModelica. For this set of test cases, IDA SE is showing radically better scalability than the other tools.

[1] Access classification as per definitions in PCA; PU = Public, CO = Confidential. Access classification per deliverable stated in FPP.
[2] Deliverable type according to FPP, note that all non-report deliverables must be accompanied by a deliverable report.
[3] Due month(s) according to FPP.
[4] It is mandatory to provide an executive summary for each deliverable.

Deliverable Contributors:

	Name	Organisation	Primary role in project	Main Author(s)[5]
Deliverable Leader[6]	EQUA	EQUA Simulation AB	T5.4 lead	
Contributing Author(s)[7]	Per Sahlin	EQUA Simulation AB		X
	Alexey Lebedev	EQUA Simulation AB		
Internal Reviewer(s)[8]				
	Peter Fritzson	Linköping University		

Document History:

Version	Date	Reason for Change	Status[9]
0.2	29/11/2016	First Draft Version	Draft
1.0	30/11/2016	First Issue	Released
1.9	06/04/2018	Draft	

5 Indicate Main Author(s) with an “X” in this column.
6 Deliverable leader according to FPP, role definition in PCA.
7 Person(s) from contributing partners for the deliverable, expected contributing partners stated in FPP.
8 Typically person(s) with appropriate expertise to assess deliverable structure and quality.
9 Status = “Draft”, “In Review”, “Released”.

CONTENTS

ABBREVIATIONS

List of abbreviations/acronyms used in document:

Abbreviation	Definition
BPS	Building performance simulation
DACH	Germany, Austria and Switzerland
HVAC	Heating ventilation and air conditioning
Hybrid DAE	Differential-Algebraic Equations with discrete events and hysteresis
IDA ICE	IDA Indoor Climate and Energy
IDA SE	IDA Simulation Environment
NMF	Neutral Model Format – a simple modelling language for Hybrid DAEs

1 INTRODUCTION

1.1 Background

"Building performance simulation" (BPS) is the term used for simulation of what goes on in a building or a district. The field is relatively large, with for example an international organization (International Building Performance Simulation Association), several international and national conference series, and hundreds of simulation tools (http://www.buildingenergysoftwaretools.com/). Basically, all processes in a building can be simulated at the planning, pre-design, design, commissioning, and operation phases at a wide range of meaningful resolutions. Energy use, indoor comfort, air quality, and light are often key study objects. Normally, the emphasis lies on whole-building simulation – all rooms and systems– over whole years of operation. In order to capture energy performance, a whole year of operation must be simulated.

Some hundreds of quantities are typically modelled for each room. Key variables are: air and surface temperatures, direct and diffuse long and short wave radiation, air moisture, CO2 etc. HVAC and control system models are similarly complex, and must normally also be simulated over full years of operation. Altogether, a model of a large building may have 10^6-10^7 variables and each model run will encompass 10^4-10^5 time steps.

These models are rarely exercised by simulation experts, knowledgeable in scientific computing and numerical methods, but by HVAC and energy engineers in building design offices. The person-time allotted in a typical project for performance simulation is counted in days rather than in weeks or months. Powerful modelling tools are a necessity to meet project time-lines.

The field is currently dominated by special purpose tools, where models and solution methods are intertwined. Tailored solution techniques are used for special sub-problems. The two leading tools in this class: IES <VE> and EnergyPlus have evolved organically out of methods that were first developed in the seventies and eighties.

The idea of using Hybrid DAEs with variable time step solvers for building performance simulation has been discussed and tried since the late eighties. A first DAE modelling language for the field, NMF, was first proposed in 1989 (Sahlin and Sowell 1989). Leading researchers largely agree, since the early nineties, about the principal advantages of using Hybrid DAEs for model definition and long-term model maintenance. However, only a single end-user tool that is based on this technology has so far become useful in everyday project work: EQUA's IDA Indoor Climate and Energy. IDA ICE is currently the leading tool in the Nordic and DACH countries.

1.2 EQUA's tools for Hybrid DAE's

IDA ICE is built as an application for EQUA's general-purpose simulation environment, IDA SE (www.equa.se). IDA SE is based on a numerical method for efficient solution of pre-compiled Hybrid DAEs. Pre-compiled components and subsystems – similar in concept to FMU for model exchange – are *acausally* interconnected into large system models. In IDA SE, a strict separation is made between the *component-maker*, who formulates, tests and compiles component models, and the *model end-user*, normally a design engineer with limited mathematical and numerical understanding. Some IDA SE applications have thousands of end-users who are completely unaware of the intricacies of Hybrid DAE modelling.

Component models for IDA SE can be written in NMF or in IDA Modelica. IDA Modelica is a flavour of Modelica that is suited to pre-compiled sub-model usage. An early version of IDA Modelica is described in (Sahlin and Grozman 2003). IDA Modelica is mostly a subset of full Modelica, but it also relies on a few language extensions. Most importantly, the Pre operator is defined also for continuous variables, where it delivers the value of the variable at the last accepted timestep. This is used in many models to remember model state (for, e.g, modelling of hysteresis), and for inline integration along with several other purposes. Although NMF can be fully and automatically translated to IDA Modelica, these extensions presently make it problematic to port full IDA models into other Modelica simulation environments.

IDA's native pre-compiled component format is conceptually similar to FMU for model exchange. However, some key features that are missing from FMI 2.0 allow the formulation of highly re-usable libraries of pre-compiled components for physical systems:

Non-expanded arrays	Sizes of arrays and tensors are parameters that can be varied in compiled code. For many applications, especially in BPS and related domains, most components have arrays, the size-variation of which are essential to practical configuration of model libraries. (For example, a wall or pipe model with a fixed number of cells would be virtually useless. A zone model or tank with a fixed number of connectors in each category would also be extremely limiting and combinatorial explosion would quickly ensue in trying to formulate re-usable component libraries.)
Physical connectors	Groups of variables that form physical connectors (for pipes, ducts, thermal and electrical connections etc.) can be defined.
DAE access	A solver can evaluate model equations and Jacobians.

Similarly to FMI, IDA Modelica component models have declared inputs and outputs. However, IDA SE interprets these only as *preferred causality*, i.e. a model end-user is free to connect inputs with inputs and outputs with outputs, enabling efficient modelling of physical systems in a fashion similar to most Modelica tools. In IDA SE, the information on preferred causality is used to ensure that each component model is well-posed and solvable. For debugging large system models this is useful, since it facilitates signalling of which individual component instance it is that gives rise to a singularity.

1.3 Modelica and FMI for Building Performance Simulation

Several academic groups develop Modelica libraries for BPS. A majority of these have during the last few years collaborated in the development of a common framework for Modelica BPS models (http://www.iea-annex60.org/). A range of useful case studies have been documented, showing for example distinct advantages when it comes to solving optimal control problems as well as the development of more flexible HVAC system model libraries.

However, full-scale Modelica BPS simulation models are still not sufficiently fast for practical application. For example, performance measures with Dymola for a 32 zone (room) model of an office building are reported in (Jorissen et al. 2015). The model contains about 30 000 variables, which is of the same order of magnitude as a comparable IDA ICE model would have. (IDA ICE will typically have about 2 000 variables per zone.) Special care has been taken in the test model to avoid short time constants; the shortest time constants are about 30s. The best reported performance is obtained with explicit integration methods while implicit methods show simulation times approaching real time. The best explicit method executes the model for a year in about 18h.

Unfortunately, these results are not very encouraging. Reported run times are too long for practical project work. Furthermore, and worse, explicit methods are very impractical for a realistic commercial implementation, since the modeller would have to take special care to avoid any short time constants in the model – an imposition that would be close to prohibitive in the marketplace. Most BPS modellers are unaware of the time constants they create in a model. Automatically detecting and adjusting them would be a major nuisance for a practical modelling tool.

Furthermore, the current practice in the Modelica world of global compilation each time a topological change has been made is impractical for the large scale models that are required for BPS. Although many simulations encompass a large number of timesteps, shorter runs are also common for, e.g., system sizing. For such runs, compilation time would be significant. Pre-compilation is therefore an attractive alternative for BPS.

In view of results such as these, researchers are looking for alternatives. Drastically different solver technologies are being pursued (Wetter et al. 2015) and hope is being placed in the evolution of FMI to better encompass physical component models.

1.4 Modelica for pre-compiled component models

The authors of this report hold the view that pre-compiled sub-models is a requirement of any viable Hybrid DAE approach to BPS. A further requirement would be to feature pre-compiled sub-models that exhibit the three listed characteristics of the IDA compiled model format: non-expanded arrays, physical connectors, and solver DAE access. Fortunately, FMI working groups are in place to include these features into the FMI standard. However, it is unclear when this work can be expected to deliver useful results.

Unfortunately, automatically converting current Modelica 3.3 library models to pre-compiled units with these required characteristics have proven exceedingly difficult. The main problem lies with parameter-controlled changes to the structure of models. Since most Modelica tools expand arrays into scalars at an early stage of the symbolic processing and the language in itself allows virtually any complex alteration of also individual elements of an array, robust and sufficiently general methods to recreate these transformations in compiled code have proven difficult to achieve. It is unclear if such tools will be developed within the foreseeable future.

On the other hand, IDA Modelica, has been defined with the specific purpose of generation of pre-compiled units that can readily be translated to pre-compiled units with non-expanded arrays. Therefore, although limiting, it is natural to base the benchmark studies on IDA

Modelica. However, since the purpose of the test suite is to compare performance of different Modelica (and FMU) tools, we have to limit the tests to IDA Modelica without the language extension features. Unfortunately, the extensions are very widely used in standard IDA applications and it is, therefore, not trivial to generate large test cases from IDA.

1.5 Purpose and scope of the suggested test cases

The purpose of the proposed cases is to capture essential performance differences between Modelica and FMU tools with respect to typical BPS problem categories. Since scaling properties with respect to problem size is of paramount importance for practical application, focus will be placed on models that can easily be scaled. Within OPENCPS, the main use for these test cases will be to test numerical performance of OpenModelica and OPENCPS FMI co-simulation master algorithms. However, in this report, primary focus is not placed on OpenModelica performance, but rather on presenting performance of other systems.

Although the main interest is on full scale, whole-building, whole-year studies, the incompatibilities listed above prevent us from generating such Modelica models within the scope of this study. Truly large-scale IDA ICE problems today (10^6 - 10^7 variables) are decoupled into weakly interacting modules which are co-simulated in parallel with a special method. This type of simulation is represented here with an FMU model for co-simulation.

Table 1 Overview of test cases

Test case	*Reference results*	*Single/ Multi core*	*Comment*
GrundA	Simulink-IDA-Dymola (Modelica)	S	A small dedicated BPS test model for all three environments
GrundC	IDA-Dymola (Modelica)	S	An elaboration on GrundA for somewhat larger models
DHCPipingS	IDA-Dymola (Modelica)	S	Scalable models for district heating/cooling networks that have been adapted from IEA Annex 60.
HCPlant	IDA-OpenModelica (Modelica)	S	An energy conversion plant model.
SotABuilding	none	M	A state of the art de-coupled and co-simulated full scale building model using *FMI for co-simulation with tool coupling*.

2 TESTING REPORT

2.1 The GrundA and GrundC cases

This case was originally constructed in order to compare pros and cons of Simulink vs. Hybrid DAE modelling for physical BPS modelling. With a minimum of complexity, the model covers some typical phenomena that are common in BPS models:

- Non-linear quasi-static pressure-flow networks (algebraic loops)
- Representative stiffness
- Representative ratio between state and total variables
- Representative number of events

The equations and physics of the original Simulink model (Figure 1) are described in Appendix I.

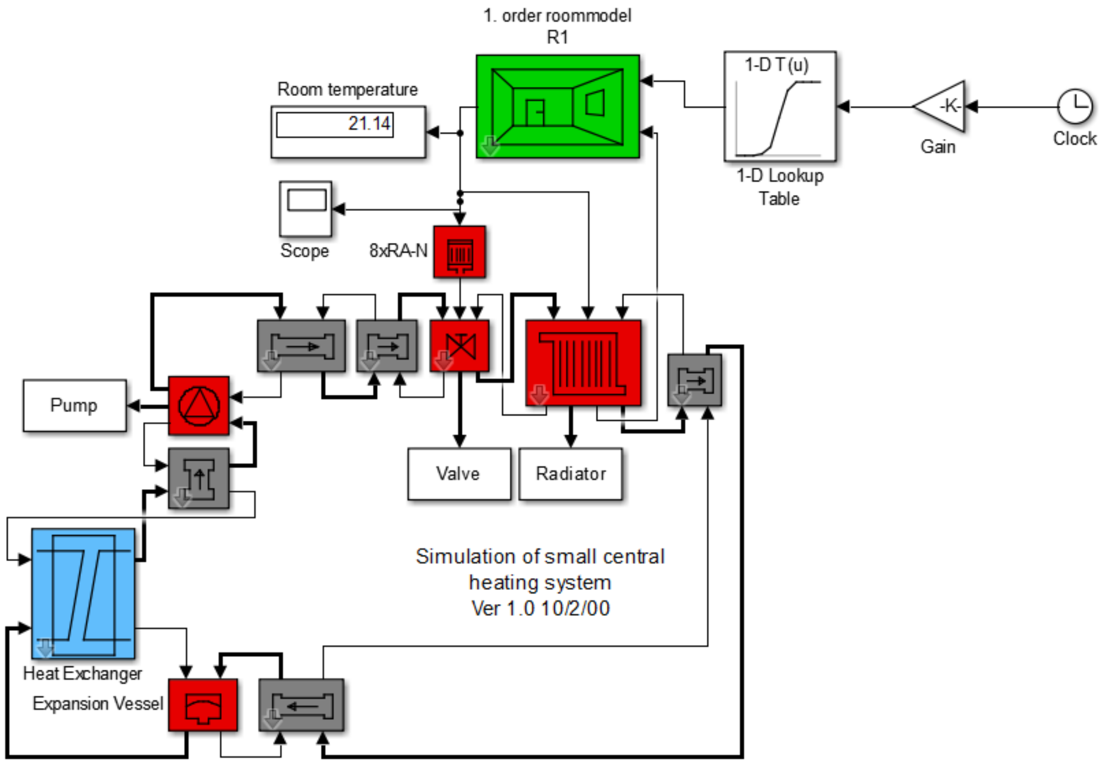


Figure 1. The GrundA Simulink model

The underlying equations were first implemented in NMF and then automatically translated into IDA Modelica without using any of the language extensions. In IDA SE, the NMF models performed nearly identically as IDA Modelica, so this distinction is not further investigated here.

Figure 2 shows the corresponding system in IDA. The acausal physical connections make it much easier to understand the diagram (and to work with it).

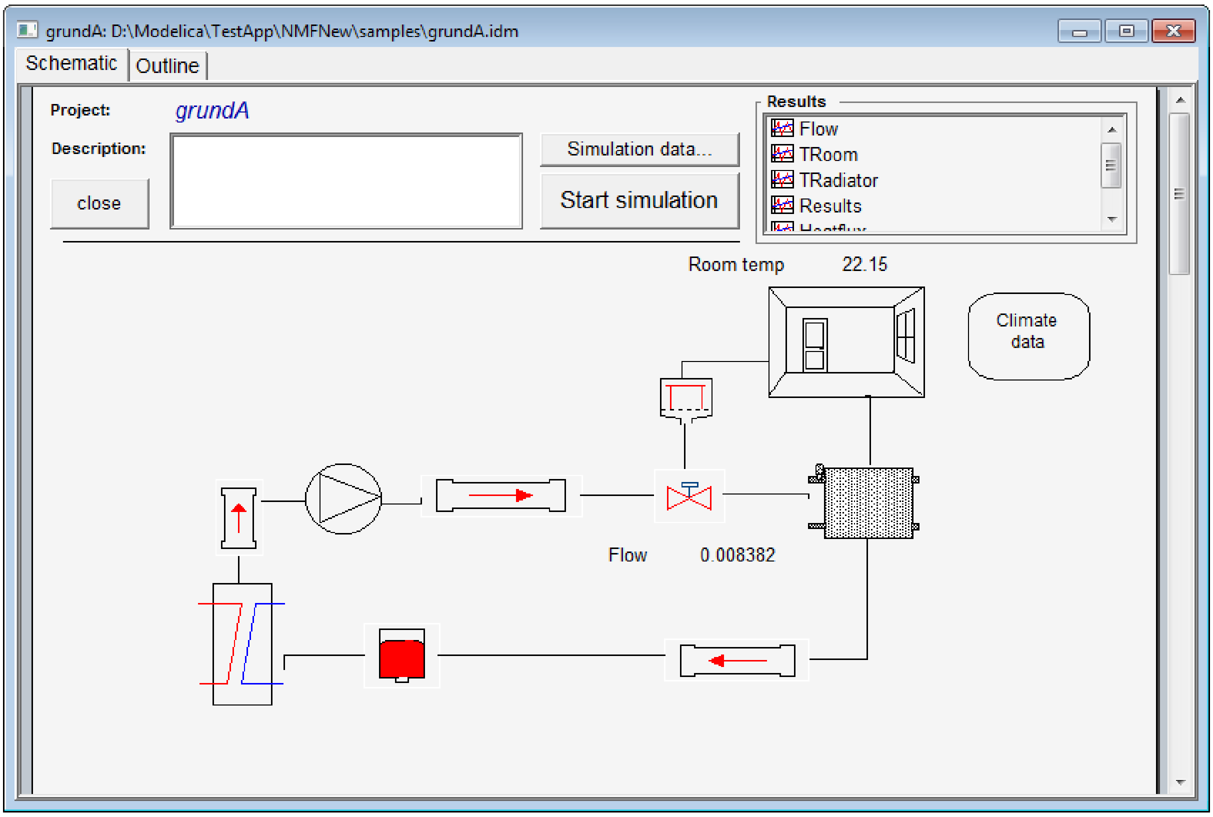


Figure 2. GrundA in IDA ICE

The single room GrundA model has a total of just 59 variables[10] and 4 states and in order to make it slightly larger, water split and merge components were added to the Modelica model so that a structure of multiple rooms and floors could be added. This variant (Figures 3.1—3.3) is called GrundC. With this extension, the model has a total of 821 variables and 32 states, still very small, but sufficient to investigate some differences in scaling.

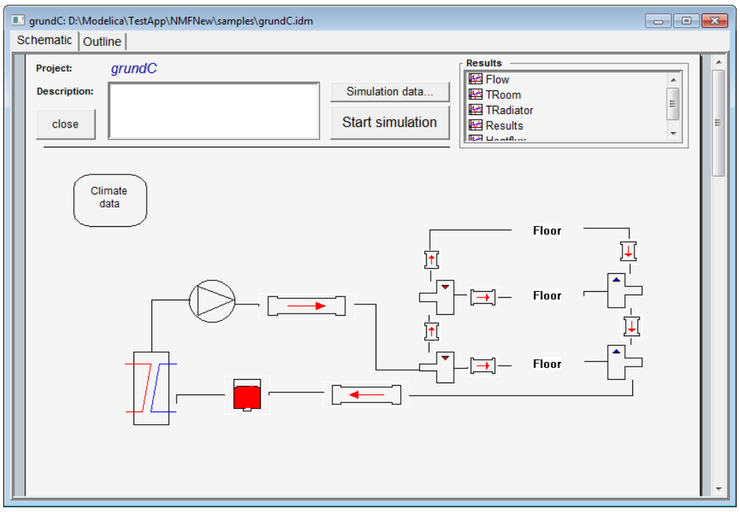


Figure 3.1. GrundA extended to three floors with 5 zones per floor.

10 In IDA's definition "Total no. of vars." – approximately equal to "nontrivial equations" in Dymola.

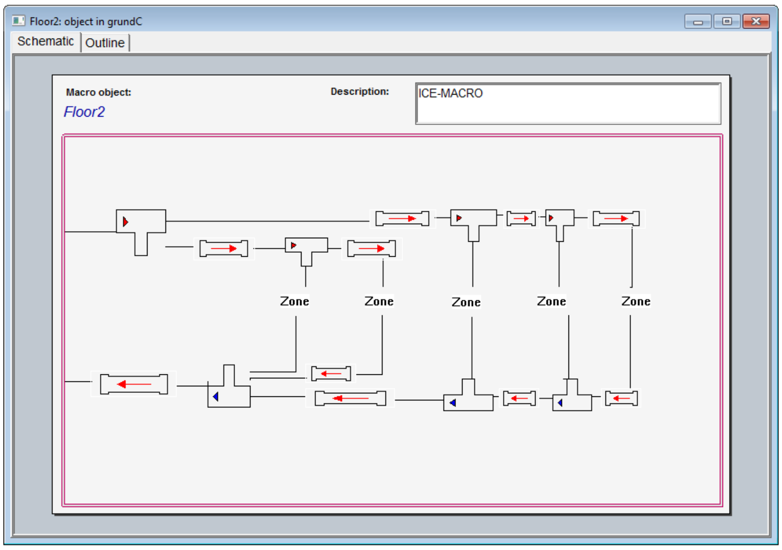


Figure 3.2. A floor in grundC.

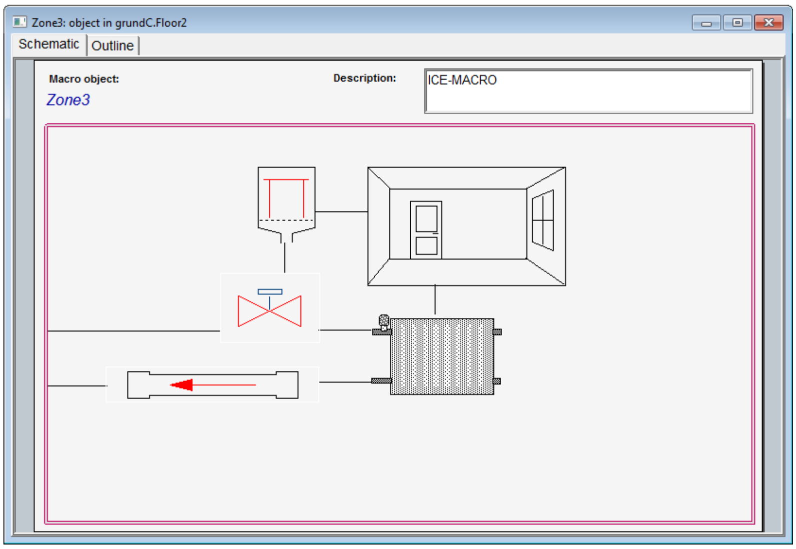


Figure 3.3. A zone in a floor in grundC.

2.1.1 GrundA performance comparisons

In its basic form, and with roughly comparable total numbers of steps, CPU time for a yearly simulation are:

	Simulink	IDA SE	Dymola (DASSL)	Dymola (CVODE)
CPU time	47s	4.1s	3.6s	2.0s

It should be noted that limited care was taken to ensure that overheads in terms of screen updates, writing of output etc. were completely comparable. However, Simulink is clearly at a disadvantage compared to the DAE based tools.

Due to the stiffness of the system, explicit methods performed radically worse on this problem.

2.1.2 GrundA result comparisons

Figure 4 shows the resulting room temperature over the year in Simulink and Dymola. Differences are negligible.

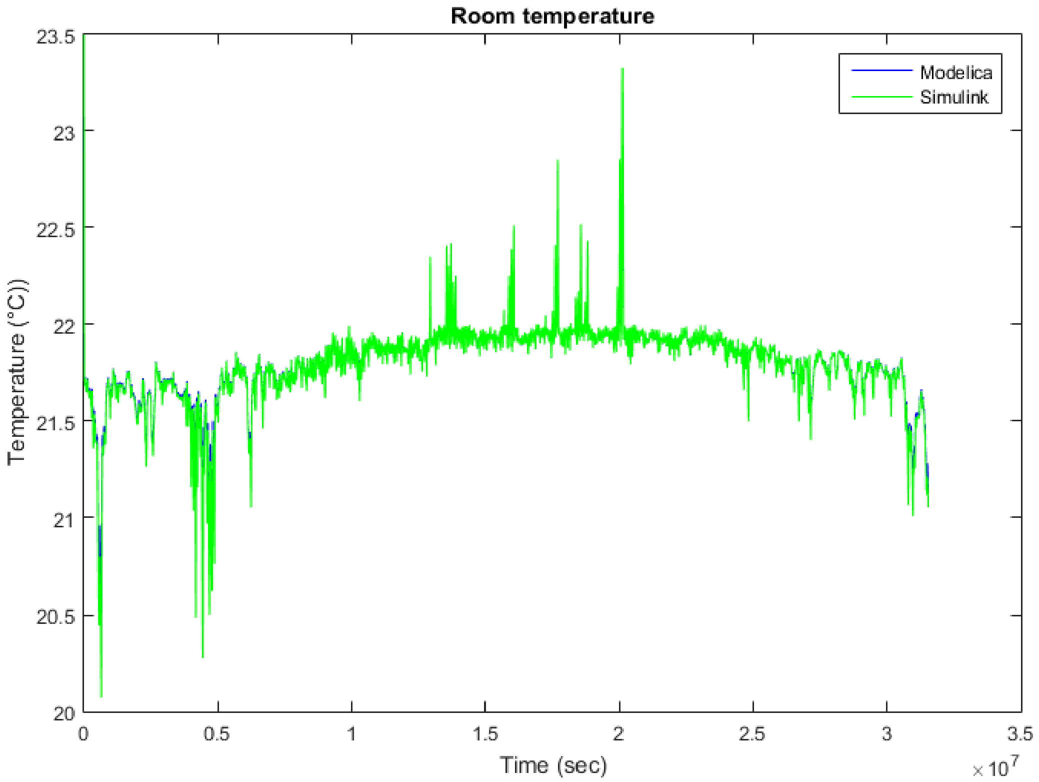


Figure 4. GrundA room temperatures

2.1.3 GrundC performance comparisons

Comparisons were made of how much CPU time and how many time steps it takes to simulate the Modelica GrundC model for one year with different tolerances. Here are the results:

IDA SE:

Tolerance	0.005	0.01	0.02	0.03	0.04	0.05	0.1	0.3	1.0
Time (s)	6.6	5.5	4.9	4.6	5.0	5.2	4.9	5.0	5.5
No. of steps	12017	9789	8602	8206	8521	8716	8358	8585	9600

Dymola:

Solver	Cvode	DASSL	DASSL	DASSL	DASSL	DASSL	DASSL
Tolerance	0.001	0.005	0.01	0.02	0.03	0.04	0.05
Time (s)	124	293	277	241	232	222	230
No. of steps	2829	13143	10146	8388	7571	7009	7211

IDA and DASSL solvers show best results at similar values of tolerance: 0.03 for IDA and 0.04 for DASSL. The number of steps the solvers do are similar, but the times they take are vastly different. The simulations with IDA ICE and Dymola have been run on separate computers with possibly slightly different performance. However, the significant difference in simulation time does not motivate a further investigation into precise computer performance.

CVODE performs roughly twice as well in terms of time, with fewer number of steps.

2.1.4 GrundC result comparisons

The following graph shows the results of the first 2000 hours (about 83 days) of simulation for IDA with tolerance 0.03 and Dymola (DASSL) with tolerance 0.01. The temperature shown is for the largest room of the top floor (represented as Floor2.Zone3 in the model). This room was chosen since the variation of its temperature is maximal compared to other rooms.

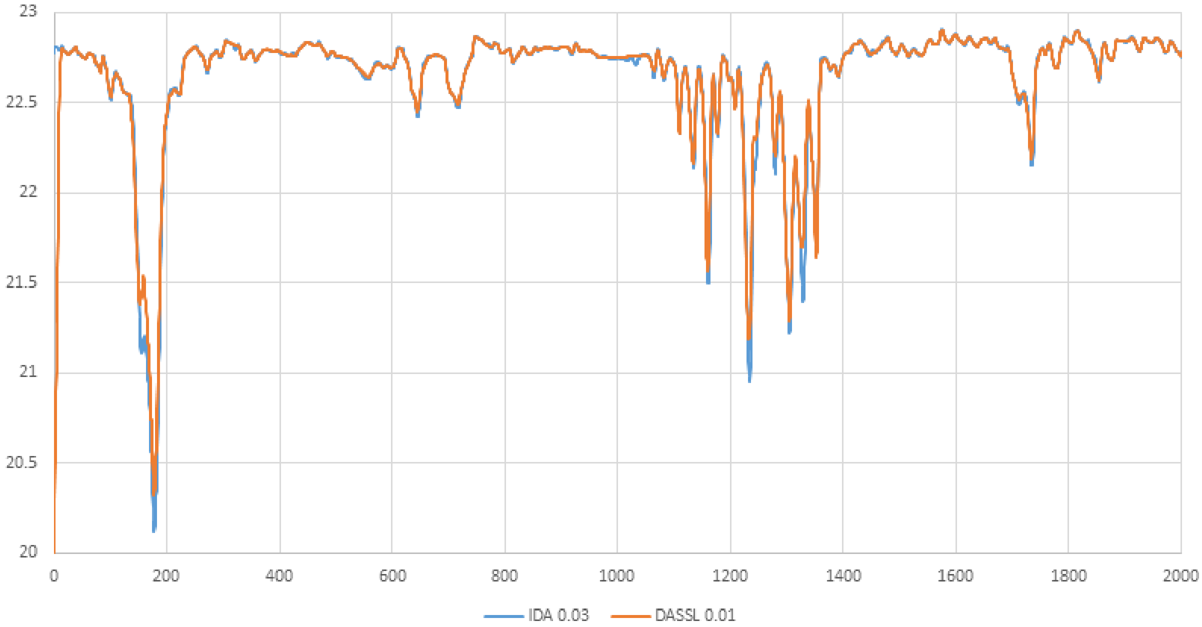


Figure 5. GrundC room temperatures

Another way to compare the results is to look at integral characteristics, for example, the time a certain variable (e.g., some temperature) is under a given value. The following graph shows such a comparison for the temperature of the above-mentioned room. Its x-axis shows time in hours, and its y-axis shows temperature such that the room's temperature spends a given time at this or lower values.

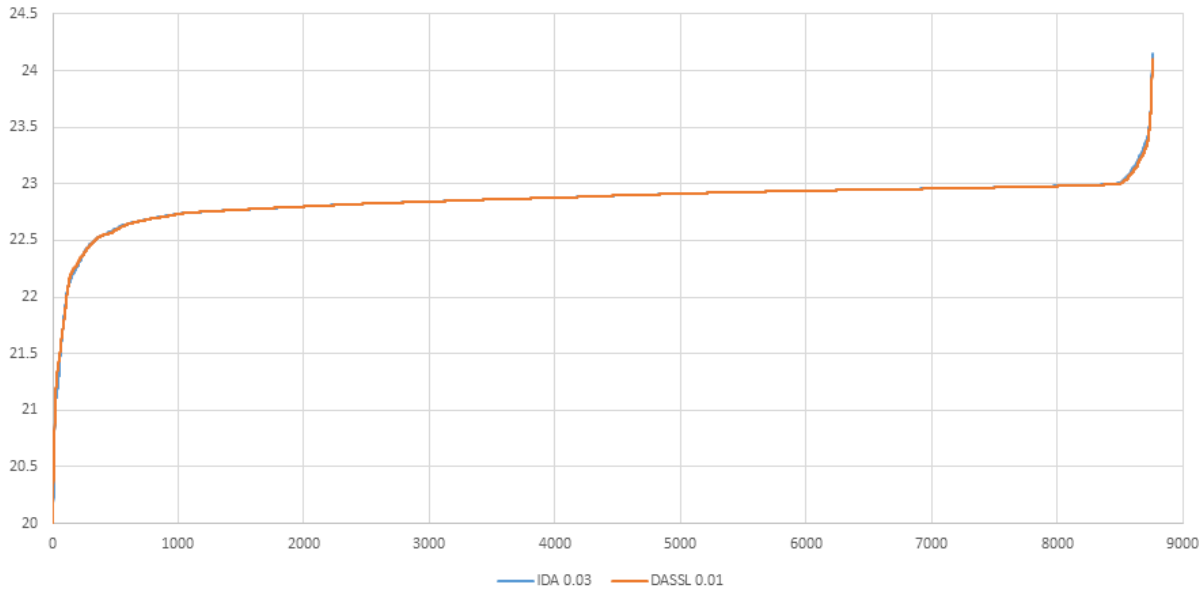


Figure 6. GrundC room temperature duration comparison

One can see from this graph that in both simulation runs, the temperature spends most of the time around the set value of 23 °C. There are some differences between the solvers at the ends of the graph.

2.2 The DHCPipingS case

This case was designed to see how well different tools cope when simulating district heating networks. As a base for the test case, data on a small DH network were used. The network contains 205 customer stations and about 900 pipes, counting both supply and return ones. The total number of variables was 81700.

The Modelica model for a customer station (a building with DH substation) was rather simple, with a total of 23 variables. Two different models were used for pipes: pipes longer than 60 meters were represented by a model using a physical delay to keep track of the propagated temperature profile along the pipe, while shorter pipes were represented by a finite difference model, with 1 cell per about 10 meters of pipe's length.

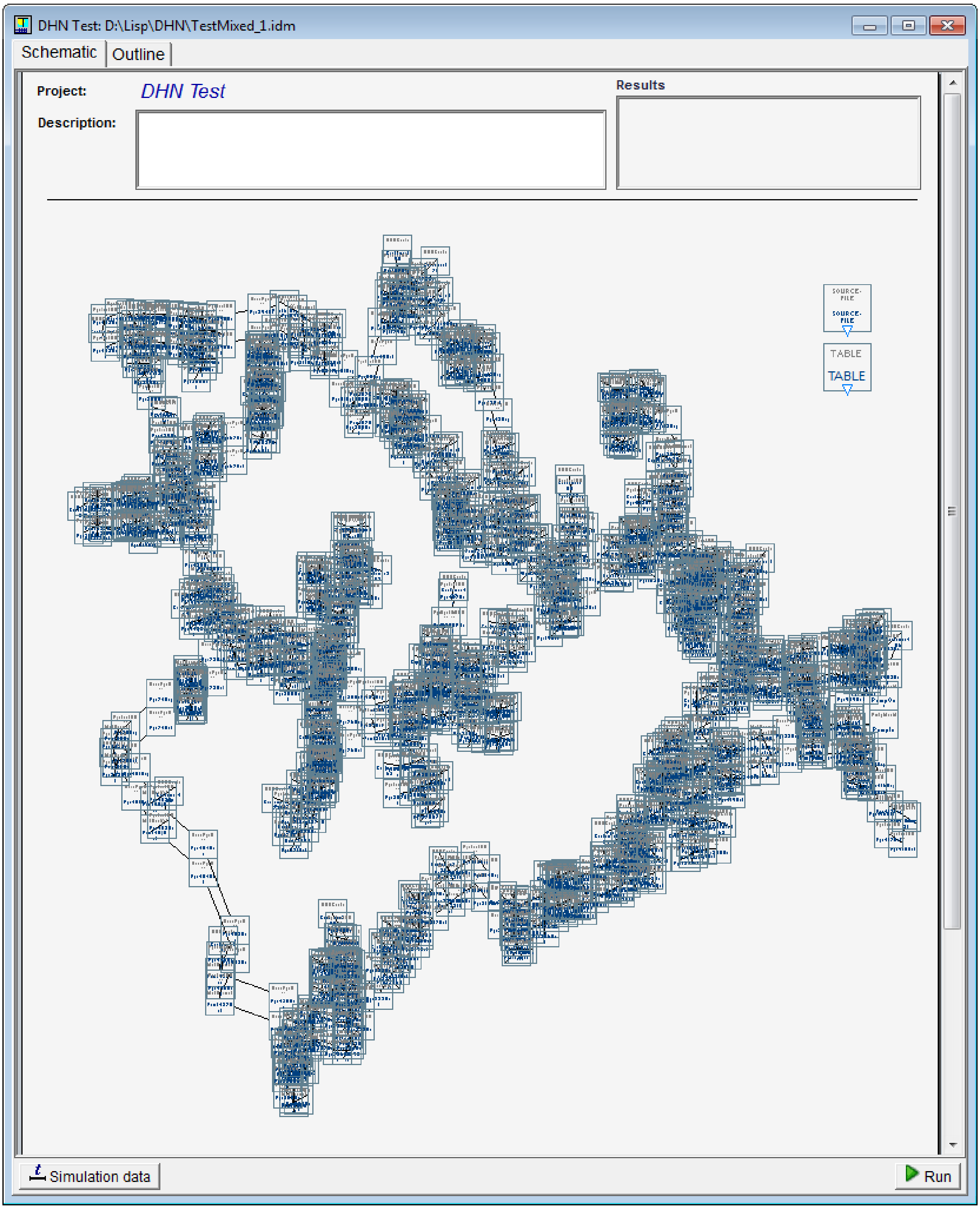
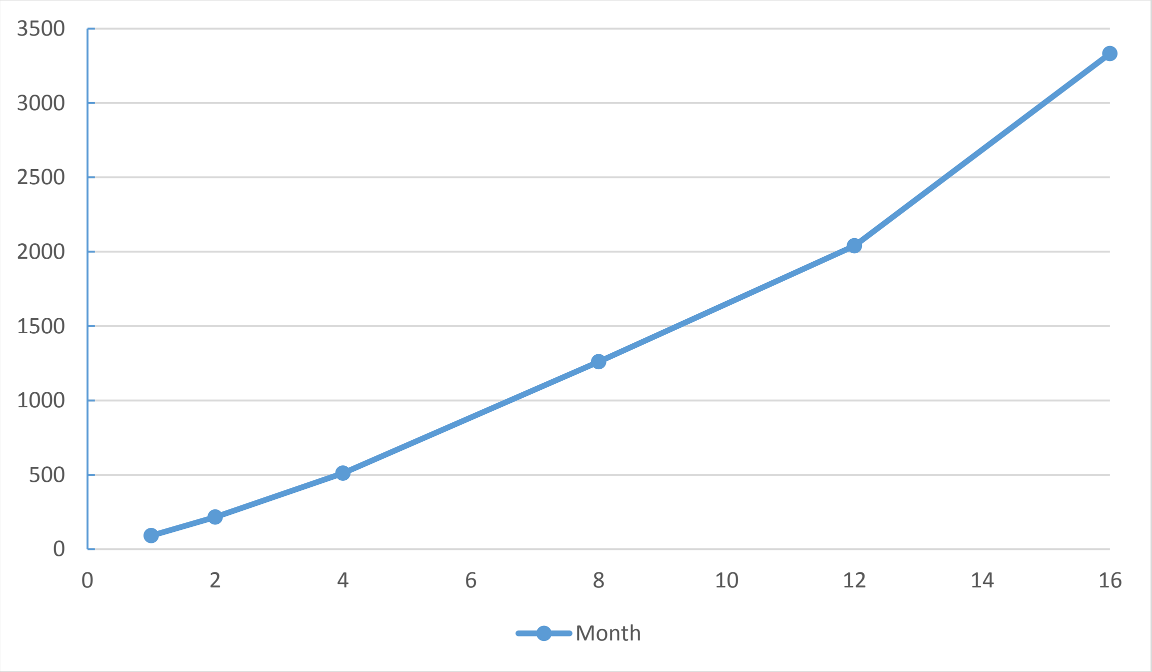


Figure 7. DHN test case in IDA ICE

Simulating this model in IDA ICE for a year took 5746 s, with 238110 steps taken. Simulating an equivalent Modelica model in Dymola *for a day* took 7578 s, with 638 steps taken. While those simulations were run on different machines with similar performance (both on a single core), IDA ICE clearly shows more favourable scalability.

In addition, systems consisting of several such networks connected in parallel to the same pump and heater, with their powers scaled proportionally, were simulated in IDA ICE. (We were unable to simulate larger systems in Dymola on the used machine.) The purpose of these experiments was to see how many copies of the network can be simulated at once and how the simulation time scales with the number of copies. The highest number of copies we have simulated successfully in IDA ICE was 16. The simulation time grows somewhat faster than linear with respect to the number of copies. (The last sample seems to deviate from the pattern, possibly due to memory paging. This phenomenon has not been further investigated, nor has the cause of the crash beyond 16 copies been studied.)

Number of copies	1	2	4	8	12	16
Time for 1 month simulation (January), s	89.1	215.1	510.0	1259.9	2040.3	3331.8



2.3 The HCPlant case

The HCPlant test case is a model of an energy conversion plant to be used for model based control. It contains a heat pump, a CHP unit, two sources of thermal energy and several heat exchangers, tanks and pumps. The IDA model contains 655 variables and 136 non-connection equations. An equivalent Modelica model contains 1055 variables and 1055 total equations, according to OpenModelica. (The difference in the number of variables is due to the fact that IDA solver treats two connected variables as one.) Source A and B have given (different) temperature levels and capacities. The sources may be used as free cooling and as a source/sink for the heat pump/chiller. The question to answer is if the CHP or heat pump shall run and how much heat/cold/electricity to produce depending on energy prices and current load.

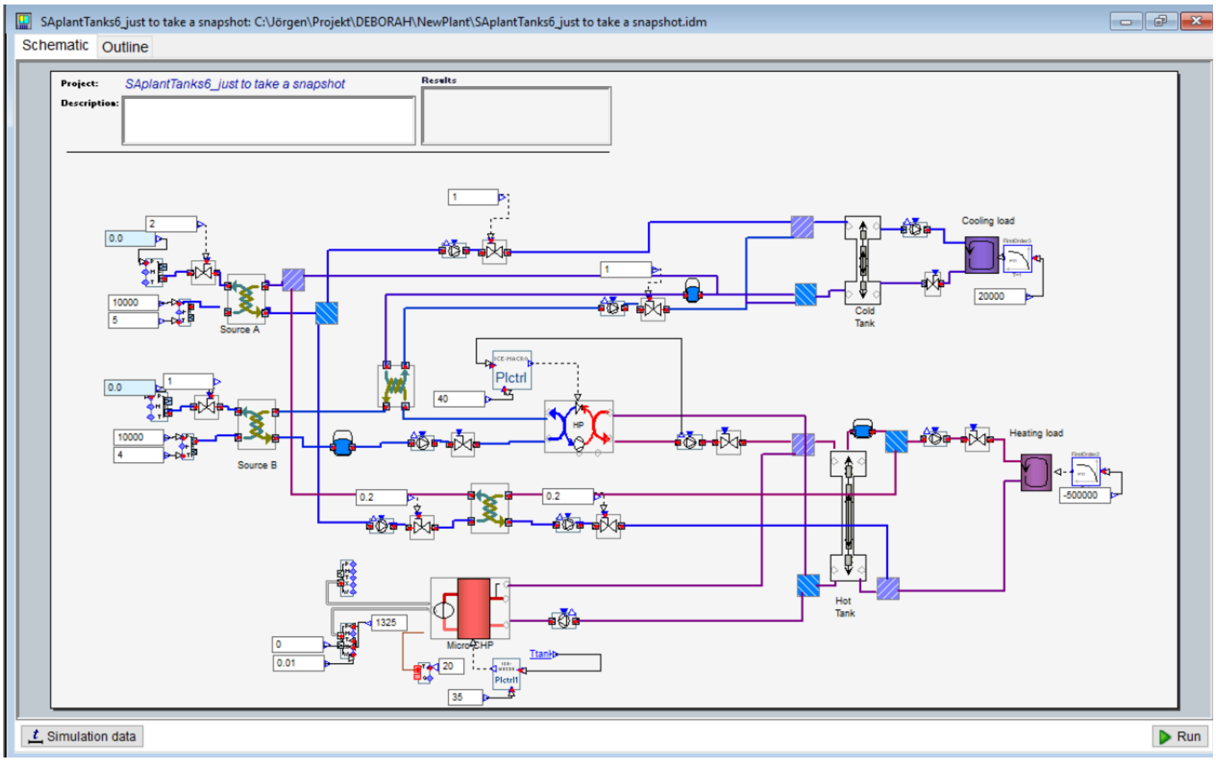


Figure 8. HCPlant test case in IDA ICE

Here is the data on simulating this model in IDA ICE and an equivalent Modelica model in OpenModelica for one year. The tolerance was set to be similar in both environments to achieve similar number of steps (tolerance settings mean different things in IDA ICE and OpenModelica, so tolerance cannot be made exactly the same). The loads were represented by sine functions for the purpose of this simulation.

	IDA ICE	OpenModelica
Time, s	20.95	1627.91
Steps	70779	63826

2.4 The SotABuilding case

The purpose of this test case is to provide an FMU (for co-simulation with tool coupling) for testing of the OPENCPS master algorithm. A complete IDA ICE based simulator is provided.

This model can be used to run several connected IDA ICE building models in parallel. The model is for example suitable to be used as a customer substation in a district. Input variables from the DH network are pressure and temperature on the supply side along with pressure on the return side. The model computes DH massflow and return temperature.

The test case uses a state of the art, full scale model of a single family home (passive house) with 11 rooms. The model takes into account ventilation, water heating, shading inside rooms, air flow between rooms and many other processes.

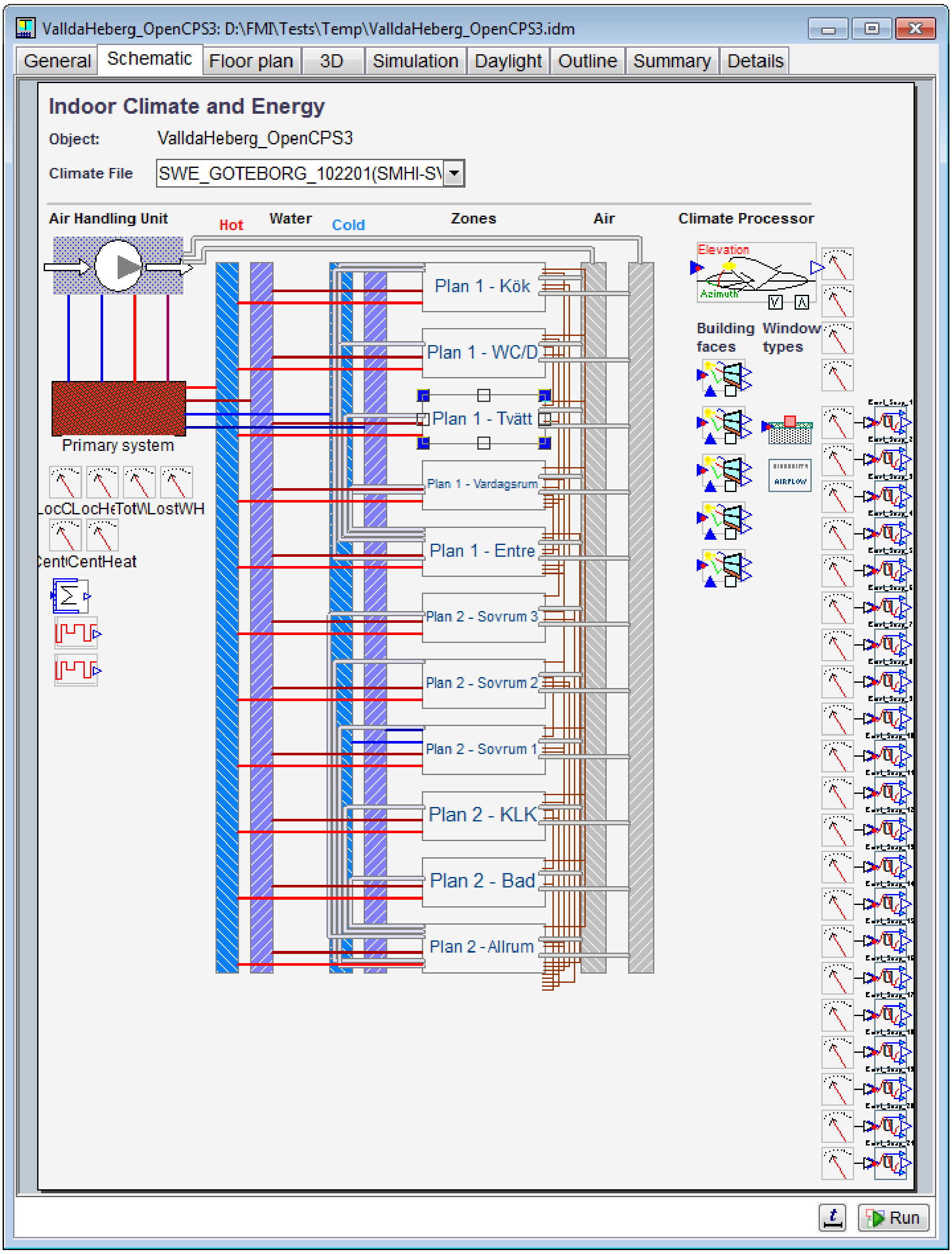


Figure 9. The building model in IDA ICE

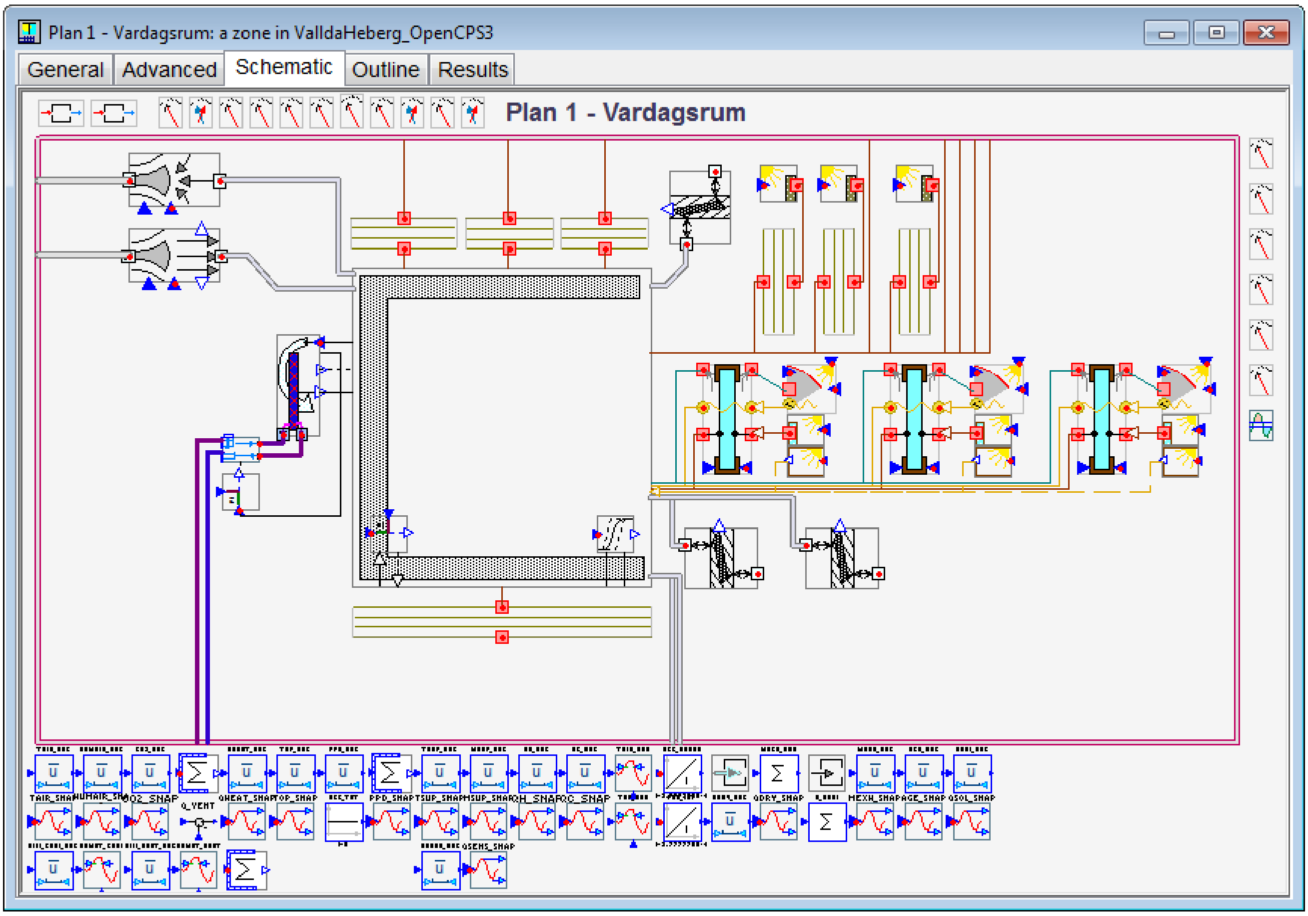


Figure 10. The model of one of the rooms in IDA ICE

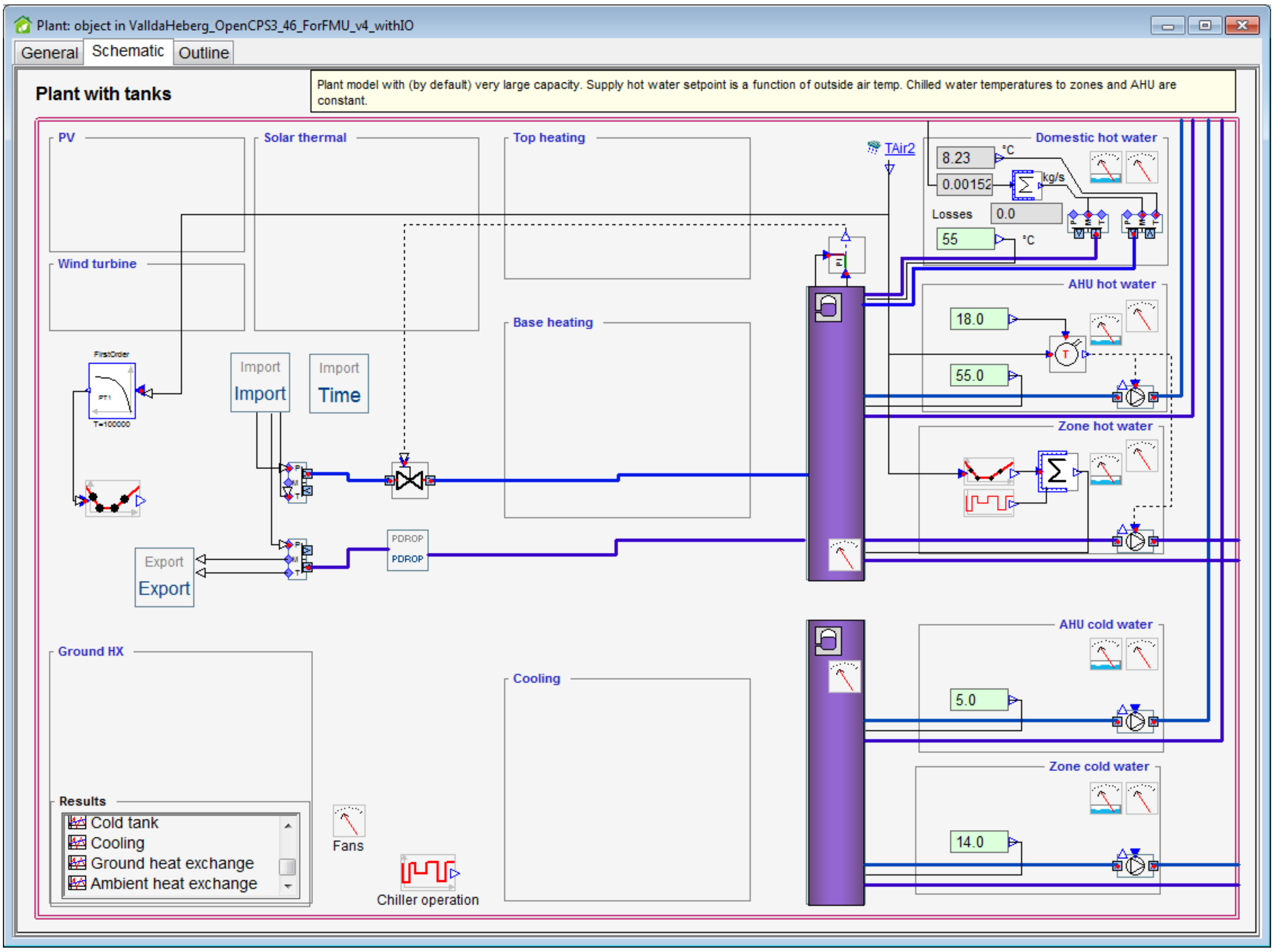


Figure 11. The model's plant, together with Import and Export components used by the FMU for communication

The model has a total of 29302 variables. However, the FMU uses only a few of these for connection to other modules. More specifically, there are 3 input variables, which represent pressures in the supply and return water pipes and the temperature of the supply water, and 2 output variables, which represent the mass flow and the temperature of the return water.

3 CONCLUSIONS AND FURTHER WORK

A range of BPS benchmark tests have been developed for OPENCPS. Some reference results have been computed, using EQUA's IDA Simulation Environment, Dymola and Open-Modelica.

Drastic differences in performance and scalability have been observed between the (sparse, pre-compiled) methods used in IDA and the Modelica tools that apply global symbolic analysis and reduction.

REFERENCES

Jorissen, Filip, Michael Wetter, and Lieve Helsen. "Simulation speed analysis and improvements of Modelica models for building energy simulation." *Proceedings of the 11th International Modelica Conference, Versailles, France, September 21-23, 2015*. No. 118. Linköping University Electronic Press, 2015.

Sahlin, Per, and Edward F. Sowell. "A neutral format for building simulation models." *Proceedings of the IBPSA Building Simulation* 89 (1989).

Sahlin, Per, Pavel Grozman, and Equa Simulation AB. "IDA Simulation Environment-a tool for Modelica based enduser application deployment." *Proceedings of the Third International Modelica Conference*. 2003.

Wetter, Michael, et al. "Prototyping the next generation energyplus simulation engine." *Accepted: 13-th IBPSA Conference. International Building Performance Simulation Association*. 2015.