

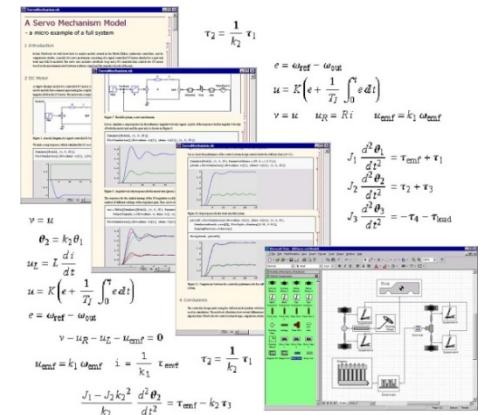
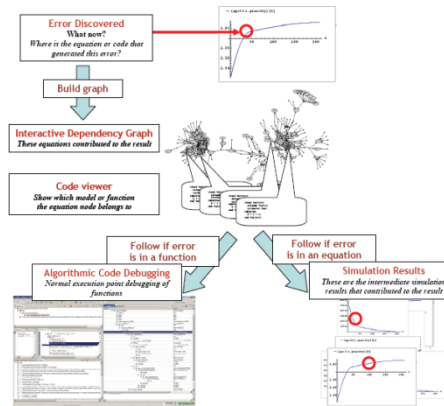
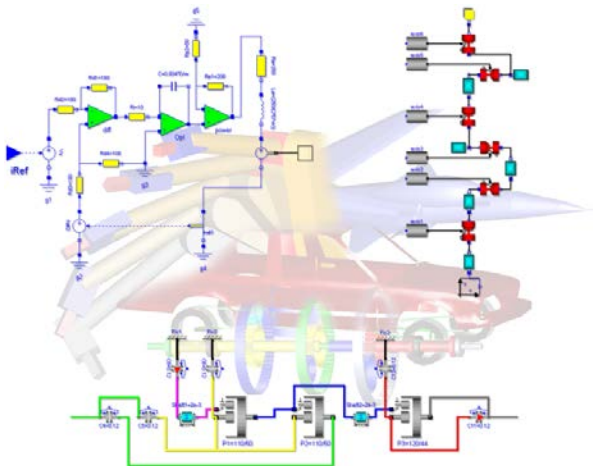
OpenModelica.org

Presentation, Status and Future Developments

Adrian.Pop@liu.se

2019-02-04

Open Source Modelica Consortium
Programming Environment Laboratory
Department of Computer and Information Science
Linköping University



www.OpenModelica.org

- **OpenModelica**
 - What is OpenModelica?
 - The past
- **OpenModelica Technical Overview**
 - OMC, OMShell, OMNotebook, OMEdit, ModelicaML, OMSimulator, OMPython, OMJulia, OMMatlab
- **OpenModelica Development Environment**
 - MetaModelica
 - The Eclipse Environment (MDT)
- **OpenModelica Latest Developments (2018-2019)**

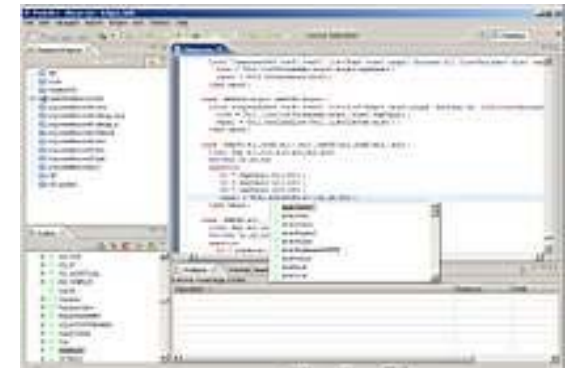
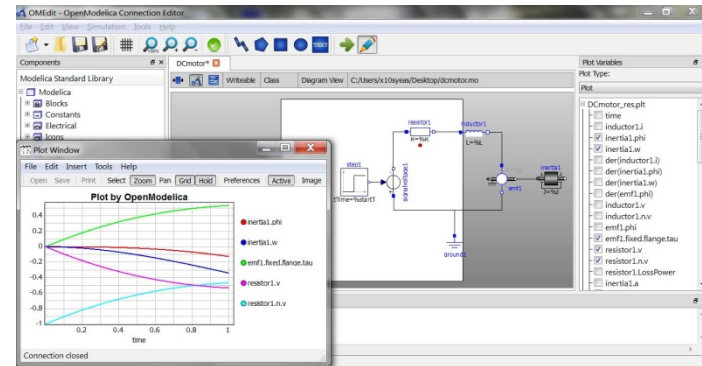
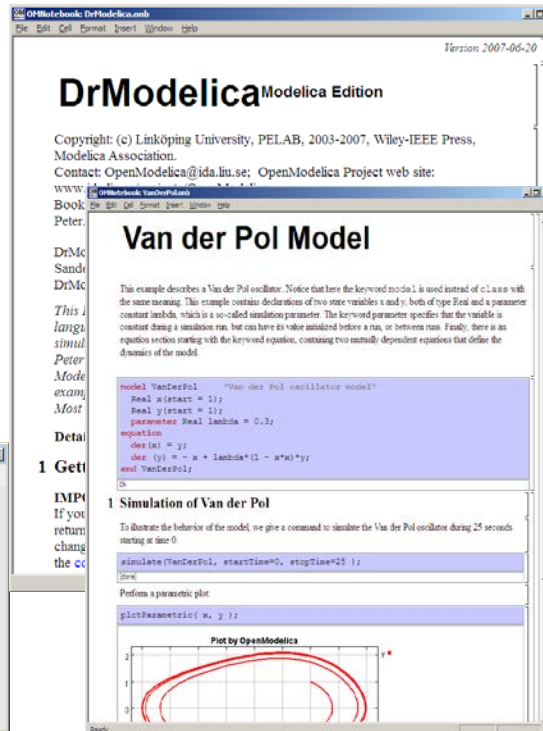
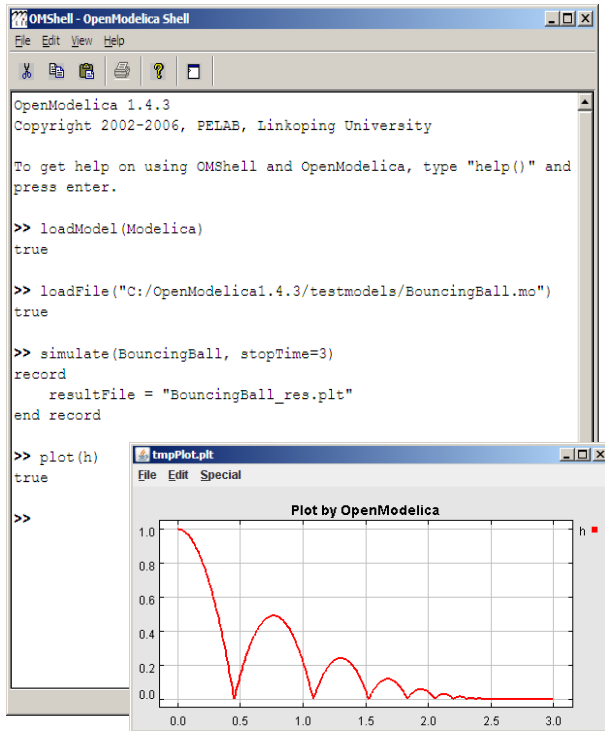
OpenModelica is ... its developers,
testers, bug reporters, contributors
and OSMC members

Thank you!

*asodja, sjoelund.se, sebco011, lochel, wbraun, niklwors, hubert.thieriot,
petar, perost, Frenkel TUD, Unknown, syeas460, adeas31, ppriv, ricli576,
haklu, dietmarw, levsu, mahge930, x05andfe, mohsen, nutaro, x02lucpo,
florosx, x06hener, x07simbj, stebr461, x08joekl, x08kimja, Dongliang Li,
jhare950, x97davka, krsta, edgarlopez, hanke, henjo, wuzhu.chen, fbergero,
harka011, tmttuomas, bjozac, AlexeyLebedev, x06klasj, ankar, kajny,
vasaie_p, niemisto, donida, hkiel, davbr, otto@mathcore.com, Kaie Kubjas,
x06krino, afshe, x06mikbl, leonardo.laguna, petfr, dhedberg, g-karbe,
x06henma, abhinck, azazi, x02danhe, rruusu, x98petro, mater, g-bjoza,
x02kajny, g-pavgr, x05andre, vaden, jansilar, ericmeyers, x05simel, andsa,
leist, choeger, Ariel.Liebman, frisk, vaurich, mwalther, mtiller, ptauber,
casella, vitalij, hkiel, jank, rfranke, mflehmg, crupp2, kbalzereit,
marchartung, adrpo*

What is OpenModelica? (I)

- Advanced Interactive Modelica compiler (OMC)
 - Supports MSL v. 3.2.1/3.2.2/MSL trunk
- Basic and advanced environments for creating models
 - OMShell - an interactive command handler
 - OMNotebook - a literate programming notebook
 - OMEdit - Connection Editor, *Transformational and Algorithmic Debugger*, 3D Viewer
 - OMPlot - OpenModelica Plotting
 - OMOptim - OpenModelica Optimization Editor
 - OMPython/OMJulia/OMMatlab - OpenModelica Python/Julia/Matlab Environment
 - MDT - an advanced textual environment in Eclipse
 - OMSimulator - co-simulation of composite models using FMUs or via TLM



What Is OpenModelica? (II)

- Advanced Eclipse-based Development Environment
- Modelica Development Tooling (MDT) - started in 2005
 - Code Assistance, Debugging, Outline & a lot more
 - *Used heavily for OpenModelica development*
 - Used in many OpenModelica Development Courses
 - *Should be replaced by OMEdit*
- ModelicaML UML/SysML integration

① System Modeling with ModelicaML

② Modelica Code Generation

③ System Simulation with Modelica Tools

What is OpenModelica? (III)

- Open-source community services
 - Website and Support Forum
 - Source versioning (github.com)
 - Trac with bug database
 - Development courses
 - Mailing lists

Welcome to OpenModelica

https://openmodelica.org

OpenModelica

HOME DOWNLOAD TOOLS & APPS USERS DEVELOPERS FORUM EVENTS RESEARCH

Top information

- OMEdit: Enhanced OpenModelica Connection Editor.
- OMPYthon: The new OpenModelica Python Interface.

Modelica/OpenModelica Videos

- Overview of Modelica, an...
- Modelica Cyber Physical...

Registration

Here is an overview presentation about Modelica and OpenModelica.

Introduction

OPENMODELICA is an open-source Modelica-based modeling and simulation environment intended for industrial and academic usage. Its long-term development is supported by a non-profit organization – the Open Source Modelica Consortium (OSMC).

The goal with the OpenModelica effort is to create a comprehensive Open Source Modelica modeling, compilation and simulation environment based on free software distributed in binary and source code form for research, teaching, and industrial usage. We invite researchers and students, or any interested developer to participate in the project and cooperate around OpenModelica, tools, and applications.

Donate

Please consider supporting our efforts.

Amount: SEK

Donate

Latest news

- CFP OpenModelica Workshop February 2014
- October 09: OpenModelica 1.9.0 released
- September 27: OpenModelica 1.9.0 RC1 released
- February 1: OpenModelica 1.9.0 Beta4 released
- October 19: OpenModelica 1.9.0 Beta2 released
- Oct 16 : CFP OpenModelica/MODPROD Workshops February 2013
- August 31: OpenModelica 1.9.0 Beta released
- April 4: OpenModelica 1.8.1 released

OpenModelica

https://github.com/OpenModelica

Search GitHub

Repositories People 21 Teams 5 Settings

Filters Find a repository... + New repository

OpenModelica

OpenModelica is an open-source Modelica-based modeling and simulation environment intended for industrial and academic usage.

OMCompiler

The OpenModelica Compiler is the core of the OpenModelica project, which is an open-source Modelica-based modeling and simulation environment intended for industrial and academic usage.

OMLibraries

OpenModelica

https://trac.openmodelica.org/OpenModelica/wiki

OpenModelica

OpenModelica Project

OpenModelica is an open-source Modelica-based modeling and simulation environment intended for industrial and academic usage. Its long-term development is supported by a non-profit organization – the Open Source Modelica Consortium (OSMC).

This Trac installation is intended to help with the OpenModelica Project management, development, bug fixing, etc.

Documentation

Automatically generated documentation of OpenModelica and Modelica.

Writing efficient MetaModelica code

Details on how to write efficient MetaModelica code for the bootstrapped compiler is here.

Modelica Compliant Libraries

We have made a list with compatibility issues between tools. To write compliant library follow this document.

Contribute

You can report a bug by adding a new ticket. Please have a look at all the open tickets first.

Testing

We run builds and tests using Hudson. Check the latest build and test status. Check the status of the (in development) compliance suite of the Modelica specification. Check the latest MSL 3.2.1 coverage. Check the latest ModelicaTest 3.2.1 coverage. Check the historical MSL coverage or trend of all tested libraries. Check the directory of all tested libraries.

MSL 3.2.1 Coverage

ModelicaTest 3.2.1 Coverage

Legend

- Target: 274
- Compile: 269
- Simulate: 248

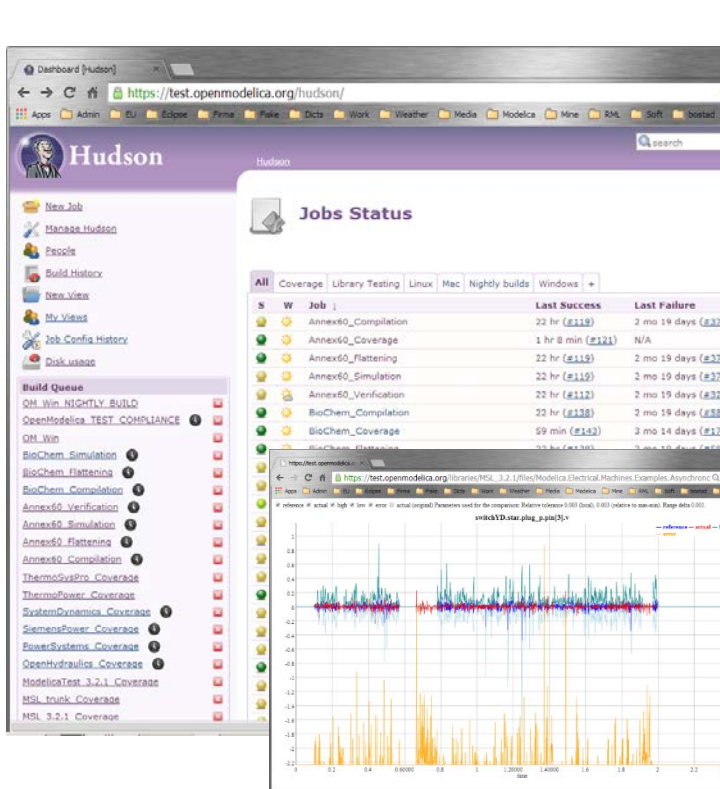
Legend

- Target: 431
- Compile: 414
- Simulate: 305

What is OpenModelica? (IV)

Open-source community services

- Extensive testing (unit & library coverage: 68 libraries, 12275 models) with interactive result comparison. 9 test servers currently
 - <https://libraries.openmodelica.org/branches/overview-combined.html>
 - Linux (GCC & CLANG), Windows (MinGW GCC), Mac OS (GCC)
 - Platforms: x86, x86_64, ARM
 - 3 runtimes: FMI, C runtime, C++ runtime
- ~3910 tests ran on each pull request via Hudson
- Automatic nightly builds for Window & Linux & Mac OS

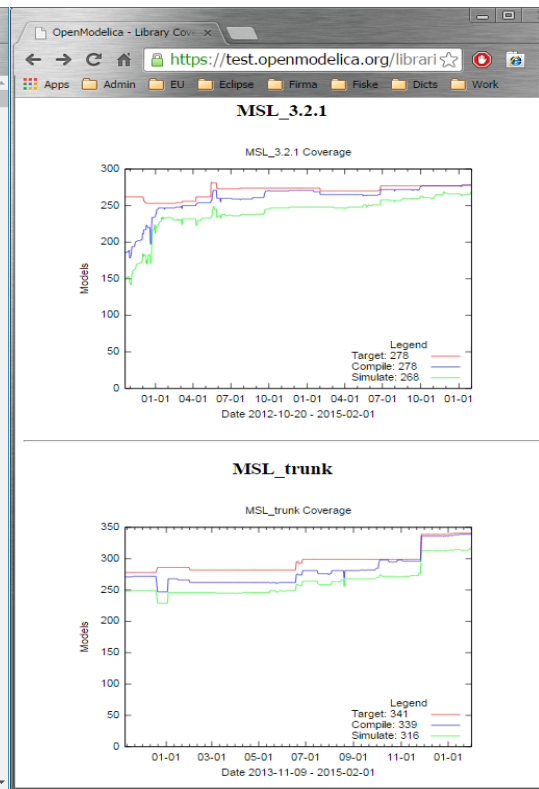


The 'OpenModelica Library Testing C' interface shows statistics for 68 libraries and 12275 models. It lists tested branches with columns for Branch, Version, Build time, Execution time, # Simulate, and Total. Below this, an 'Advanced Noise' table shows the results of various tests across different branches and versions.

Branch	Version	Build time	Execution time	# Simulate	Total
v1.8.1-ml	1.8.1 (f164542)	2019-02-01 23:38:56 2 days, 2:50:53	1277	12225	
v1.9.0-ml	1.9.0 (f17627)	2018-10-20 04:59:19 2 days, 7:22:20	4432	12133	
v1.9.1	1.9.1 (r22929) (Bootstrapping version)	2018-10-20 05:10:31 1 day, 5:07:24	2587	12133	
v1.9.2	1.9.2 (z25115 C)	2018-10-20 05:12:56 2 days, 5:34:16	5803	12133	
v1.9.3	OpenModelica 1.9.3	2018-06-09 09:30:59 2 days, 5:22:23	6429	11776	
v1.9	v1.9.7-v1.9.7.3 g6347e1f61	2018-06-09 10:33:44 1 day, 20:14:27	6888	11776	
v1.11	v1.11.0-v1.11.0.8+gbd4991c5b	2019-02-02 22:08:31 1 day, 18:38:04	7213	12225	
v1.12	OMCompiler v1.12.0-v1.12.0.7+ga21325026	2019-02-02 22:39:03 2 days, 12:09:04	8237	12225	
v1.13	OMCompiler v1.13.2	2019-02-02 23:13:40 2 days, 3:22:48	9120	10863	
master	OMCompiler v1.14.0-dev.92+g05c8b026d	2019-02-02 23:49:54 2 days, 8:12:45	10190	12225	

Version	v1.8.1-ml	v1.9.0-ml	v1.9.1	v1.9.2	v1.9.3	v1.9	v1.11	v1.12	v1.13	master
1.0.0	1.0.0	1.0.0	1.0.0	1.0.0	1.0.0	1.0.0	1.0.0	1.0.0	1.0.0	1.0.0
revision	revision	revision	revision	revision	revision	revision	revision	revision	revision	revision
1.0.0-35-	1.0.0-35-	1.0.0-35-	1.0.0-35-	1.0.0-35-	1.0.0-5-	1.0.0-5-	1.0.0-35-	1.0.0-35-	1.0.0-35-	1.0.0-35-
g7036d6f	g7036d6f	g7036d6f	g7036d6f	g7036d6f	gaa44378	gaa44378	g7036d6f	g7036d6f	g7036d6f	g7036d6f

Branch	Total	Parsing	Frontend	Backend	SimCode	Templates	Compilation	Simulation	Verification
v1.8.1-ml	15	15	0	0	0	0	0	0	0
v1.9.0-ml	15	15	0	0	0	0	0	0	0
v1.9.1	15	15	0	0	0	0	0	0	0
v1.9.2	15	15	0	0	0	0	0	0	0
v1.9.3	15	15	8	1	1	1	0	0	0
v1.9	15	15	8	3	3	3	1	1	0
v1.11	15	15	8	3	3	3	1	1	0
v1.12	15	15	8	3	3	3	1	1	0
v1.13	15	15	3	3	3	3	1	1	0
master	15	15	3	3	3	3	1	1	0



What is OpenModelica? (V)

- **An incubator platform for research**
 - 9 PhDs since 2004 (Debugging, Parallelization, PDEs Extensions)
 - 36 Master's theses since 2004
 - Both the students and the project benefit
- **Master theses at PELAB 2006-2018**
 - Refactoring/Parsing and Language extensions
 - UML/SysML view of Modelica code
 - 2D and 3D visualization tools
 - Static and runtime debugging tools
 - Advanced code generation and parallelization of simulation code
 - Bootstrapping and Java Interface
 - Function pointers
 - NVIDIA for Cuda and OpenCL parallel simulation
 - OMEdit - Modelica Connection Editor
 - OMWeb - server based Modelica simulation for teaching
 - OMCcc parser
 - PDE-solver using ParModelica
- **External Master theses**
 - Model based diagnostics at ISY (Dep. Of Electrical Engineering)
 - Monte-Carlo simulation of Satellite Separation Systems at SAAB
 - Interactive Simulations (EADS)
 - Additional Solvers + Event handling (FH-Bielefeld)
 - EADS - ModelicaML
- **A Base for commercial and open source products**
 - MathCore AB, Bosch Rexroth, VTT, Equa, Evonik, ABB

OpenModelica Roadmap - Past

1997 - started as a master thesis

2003 - first usable internal version

2004 - first external version: OpenModelica 1.1

2005 - more development: OpenModelica 1.3.1

2006 - major milestone

- Translated the whole compiler to MetaModelica
- Integrated Development Environment for the compiler
- OpenModelica website started
- Moved the code repository to Subversion management
- Extended the OpenModelica environment with new tools
- 4 versions released during the year
- External people start using OpenModelica
 - ~ 200 downloads/month
 - first development course at INRIA

OpenModelica Roadmap - Past

2007 - continued development and community involvement

- Improvement in website, support and documentation
- Answered ~1000 questions on the forum
- Portability is highly improved, ported to 4 platforms
 - Linux, Mac, Solaris, Windows (version 1.4.3)
- Improvement of the compiler development tools in Eclipse
- OpenModelica Community starts to react
 - contribute code & report bugs & request enhancements & participate in answering questions in the OpenModelica forum
 - participate at courses and workshops
- New server acquired for better community services
- Increased usage: ~600 downloads/month
- Open Modelica Consortium created in December 4
 - 4 months of work
 - 9 organizations as members already (3 Universities, 6 Companies)
 - discussions are ongoing with other 6 companies

2008 - Further work on the compiler

- Release 1.4.4 and 1.4.5
 - Linux, Mac, Solaris, Windows
- New Solver Interface
- Refactoring
- Dynamic loading of functions
- Merging of MathCore front-end code
- 744 commits in Subversion
- Other things I don't remember

2009

- Work mainly happened in OSMC (partially on a non-public branch)
- **Front-end**
 - Refactoring (OSMC)
 - Enumerations (OSMC)
 - Java Interface and Bootstrapping (Martin Sjölund)
 - MultiBody flattening (OSMC)
 - Constraint connection graph breaking (VTT + OSMC)
 - Support for Modelica 3.x and 3.x annotations (OSMC)
- **Back-end**
 - Tearing in the back-end (Jens Frenkel)
 - Template Code Generation and CSharp backend (Pavol Privitzer, Charles University Prague)
 - Interactive Simulations (EADS)
 - C++ Code generation (Bosch Rexroth)
 - Java Interface and Bootstrapping (Martin Sjölund)
 - Additional Solvers + Events (Willi Braun, FH-Bielefeld)
- **General**
 - New ModelicaML + SysML prototype (EADS)
 - 1144 commits in subversion (Since 2009 to February 8, 2010)
 - Bug fixes (OSMC)
 - Release 1.5.0 and 1.5.0-RC_X (Linux, Mac, Solaris, Windows)
- **More things I don't remember**

OpenModelica Roadmap - Past

2010 - 2011

- Support for Modelica Standard Library 3.1 (Media & Fluid in works)
- **Front-end**
 - MultiBody flattening (OSMC)
 - Support for Modelica 3.x and 3.x annotations (OSMC)
 - Performance Enhancements
 - Stream connectors
 - Media & Fluid work is on the way
- **Back-end**
 - Back-end redesign (Jens, Willi, Martin, Per, Adrian, Kristian, Filippo)
 - Tearing in the back-end (Jens Frenkel)
 - Template Code Generation and CSharp backend (Pavol Privitzer, Charles University Prague)
 - Interactive Simulations (EADS)
 - C++ Code generation (Bosch Rexroth)
 - Additional Solvers + Events + Linearization (Willi Braun, FH-Bielefeld)
- **General**
 - OMEdit - new connection editor
 - Bootstrapping OMC (90% finished)
 - 2550 commits in subversion from 2010 to Feb. 7, 2011 (double than 2009-2010)
 - Bug fixes ~300+ (OSMC)
 - Release 1.6.0 (Linux, Mac, Windows)
 - Downloads Windows (~16434) , Linux (~8301), Mac (~2816)
- **More things I don't remember**

OpenModelica Roadmap - Past

2012 - 2013

- Support for Modelica Standard Library 3.2.1 including Media & Fluid
- **Front-end**
 - Performance Enhancements
 - Media & Fluid work
 - Operator overloading
 - New instantiation module started
- **Back-end**
 - Modular back-end with more optimization modules (Jens, Willi, Martin)
 - New simulation runtime redesign (Willi, Lennart, Jens, Martin, Adrian)
 - C++ Code generation (Bosch Rexroth)
 - FMI export & import
 - Initialization, Jacobians (Lennart Lochel, Willi Braun, FH-Bielefeld)
 - Support for parallelization (Martin)
 - Parallel extensions in functions
- **General**
 - Uncertainties support (OpenTURNS connection & Data reconciliation)
 - MDT GDB debugging based on GDB and the bootstrapped compiler
 - OMEdit - improvements
 - Bootstrapping OMC (100% finished) using Boehm GC
 - 3909 commits in subversion from 2012 to Feb. 4, 2013
 - 2000 forum posts (questions and answers)
 - Bug fixes ~247+ (OSMC)
 - Release 1.9.0 (Linux, Mac, Windows)
 - Downloads Windows (~45307) , Linux (~15543), Mac (~5367)
- **More things I don't remember**

OpenModelica Roadmap - Past

- 2014 - 2017 - Most focus on libraries support & performance
 - MSL 3.2.1 (100% build/98% simulate), ModelicaTest 3.2.1, PetriNet, Buildings, PowerSystems, OpenHydraulics, ThermoPower, and ThermoSysPro
 - Switch to bootstrapped compiler
- Front-end, Back-end, Simulation Runtime, Graphical Clients
 - Development switched to bootstrapped compiler since November 2014
 - Partially new graph-based front-end with better support for libraries
 - Improved back-end: initialization, system solving, parallelization, cse optimization, dynamic optimization
 - Faster and much more user friendly OpenModelica Connection editor
- General
 - ~9000 commits in subversion from Feb. 2014 to Feb., 2016
 - Bug fixes
 - Release 1.9.2 (Linux, Mac, Windows)

- Testing procedure developed by Martin Sjölund
 - <https://libraries.openmodelica.org/branches/overview-combined.html>
 - Run tests on previous OpenModelica version until 1.8.1
 - Detect both model regression and performance regression, all information saved in a database
 - 68 libraries, 12275 models with interactive result comparison.
 - 3 dedicated test servers
 - Linux (GCC & CLANG), Windows (MinGW GCC), Mac OS (GCC)
 - Platforms: x86, x86_64, ARM
 - 5 runtimes: FMI, C runtime, C++ runtime, newInst, daeMode

Statistics

Number of libraries 68

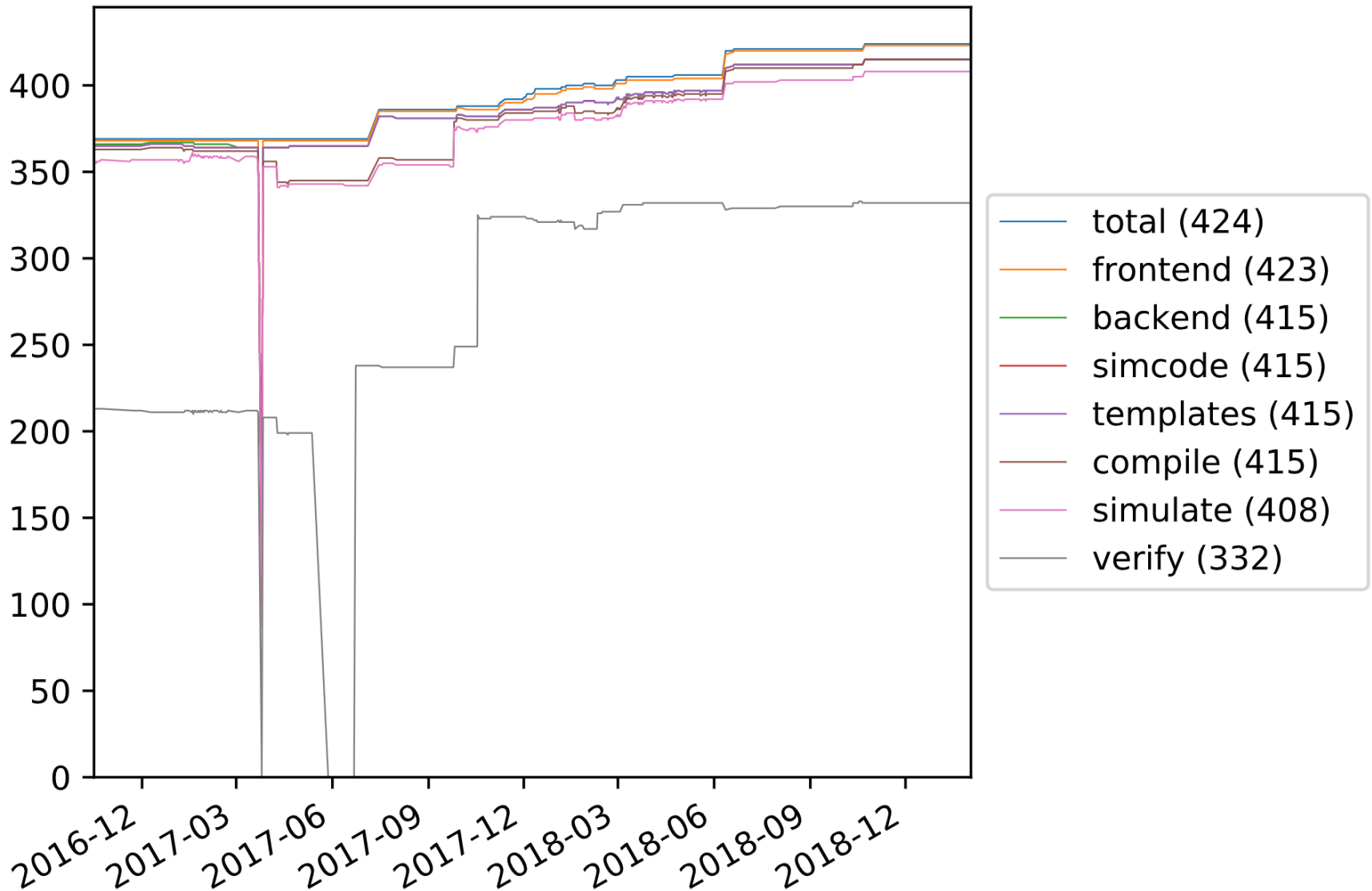
Number of models 12275

Tested branches

Branch	Version	Build time	Execution time	# Simulate	# Total
v1.8.1-rml	1.8.1 (r11645+2)	2019-02-01 23:38:56	2 days, 2:50:53	1277	12225
v1.9.0-rml	1.9.0 (r17627)	2018-10-20 04:59:19	2 days, 7:22:20	4432	12133
v1.9.1	1.9.1 (r22929) (Bootstrapping version)	2018-10-20 05:10:33	1 day, 5:07:24	2587	12133
v1.9.2	1.9.2 (r25115 C++)	2018-10-20 05:12:56	2 days, 5:34:16	5803	12133
v1.9.3	OpenModelica 1.9.3	2018-06-09 09:30:59	2 days, 5:22:23	6429	11776
v1.9	v1.9.7-v1.9.7.3+g6347e1f61	2018-06-09 10:33:44	1 day, 20:14:27	6888	11776
v1.11	v1.11.0-v1.11.0.8+gbda991e5b	2019-02-02 22:08:31	1 day, 18:38:04	7213	12225
v1.12	OMCompiler v1.12.0-v1.12.0.7+ga21325026	2019-02-02 22:39:03	2 days, 12:09:04	8237	12225
v1.13	OMCompiler v1.13.2	2019-02-02 23:13:40	2 days, 3:22:48	9120	10863
master	OMCompiler v1.14.0-dev.92+g05c8b026d	2019-02-02 23:49:54	2 days, 8:12:45	10190	12225

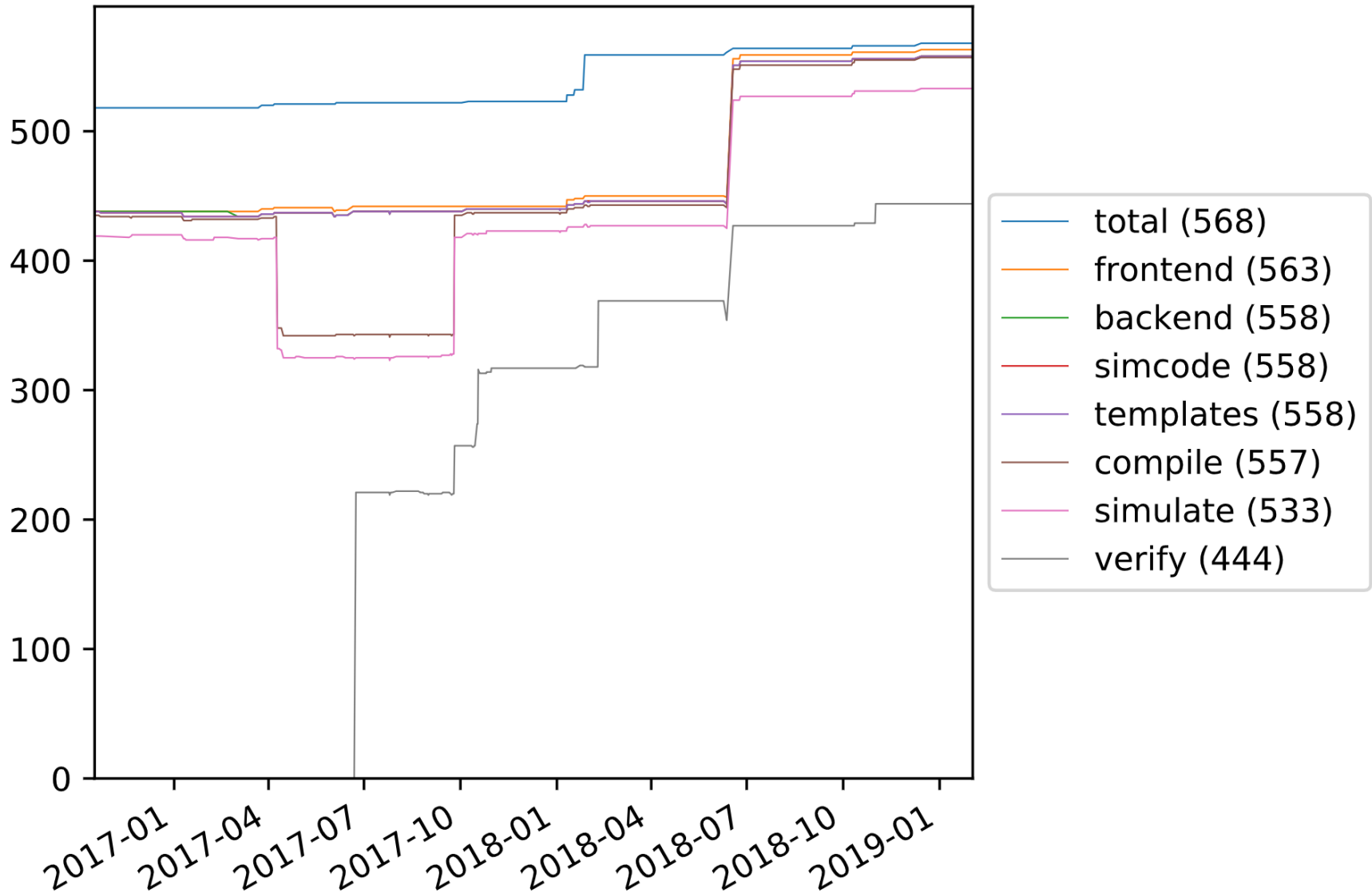
OpenModelica Testing (II)

- 2019-02-04 v1.14-dev - total 424 - build 415 (98%) - sim 408 (96%)
Modelica_trunk (master branch)



OpenModelica Testing (III)

- 2019-02-04 v1.14-dev - total 568 - build 557 (98%) - sim 533 (93%)
ModelicaTest_trunk (master branch)



- Moved the source code to github May 2015
- Mature code base: <https://github.com/OpenModelica>
- ~9000K lines of code and tests

- From Feb 2017 - Feb 2018
 - 20 contributors
 - 794 commits (OMCompiler)

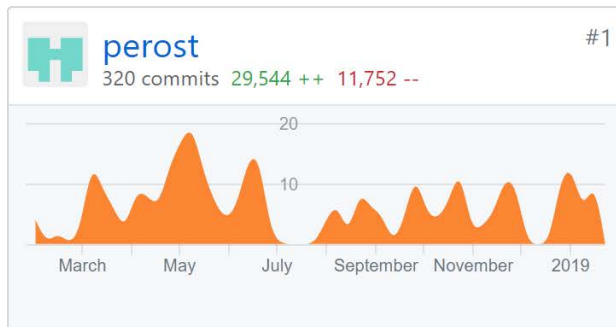
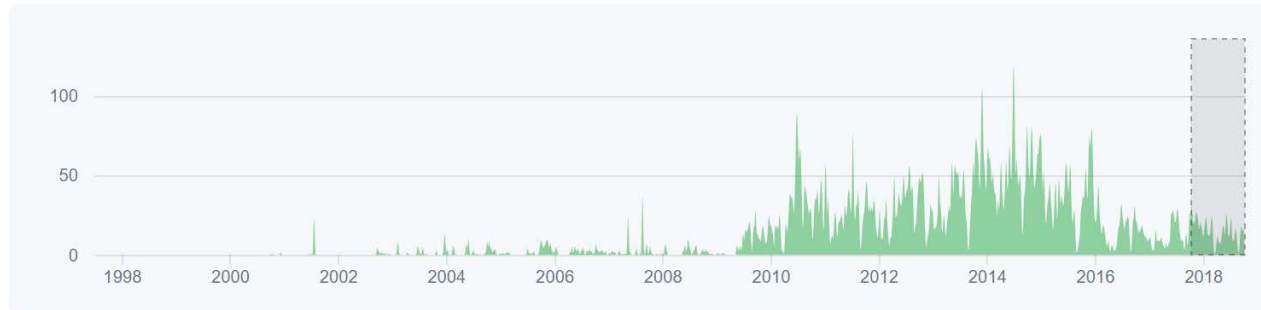
- From Feb 2018 - Feb 2019
 - 30+ contributors
 - 800 commits (OMCompiler)
 - 969 commits (OMSimulator)
 - 213 commits (OMEdit)

OpenModelica Statistics (II)

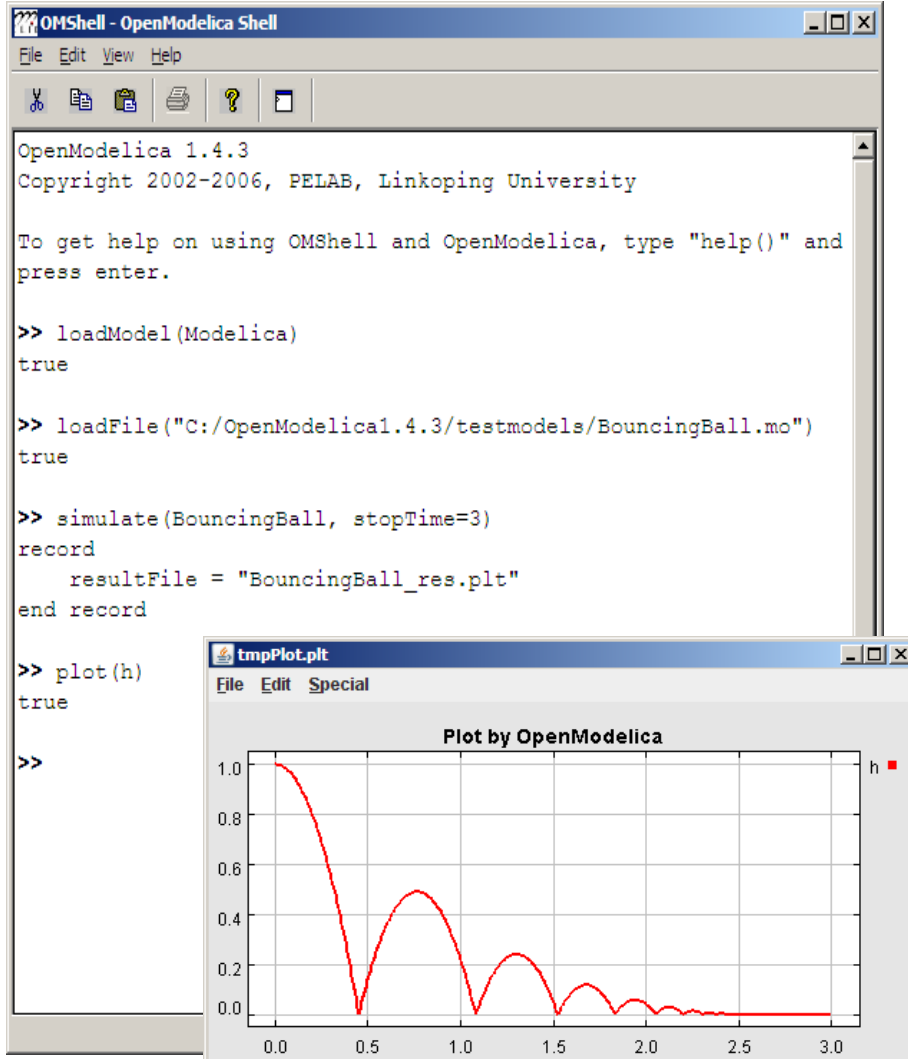
Feb 5, 2018 – Feb 3, 2019

Contributions: Commits ▾

Contributions to master, excluding merge commits



- OpenModelica
 - What is OpenModelica?
 - The past
- OpenModelica Technical Overview
 - OMC, OMShell, OMNotebook, OMEdit, ModelicaML, OMSimulator, OMPython, OMJulia, OMMatlab
- OpenModelica Development Environment
 - MetaModelica
 - The Eclipse Environment (MDT)
- OpenModelica Latest Developments (2018-2019)



OMShell - OpenModelica Shell

File Edit View Help

OpenModelica 1.4.3
Copyright 2002-2006, PELAB, Linköping University

To get help on using OMShell and OpenModelica, type "help()" and press enter.

```
>> loadModel(Modelica)
true

>> loadFile("C:/OpenModelica1.4.3/testmodels/BouncingBall.mo")
true

>> simulate(BouncingBall, stopTime=3)
record
  resultFile = "BouncingBall_res.plt"
end record

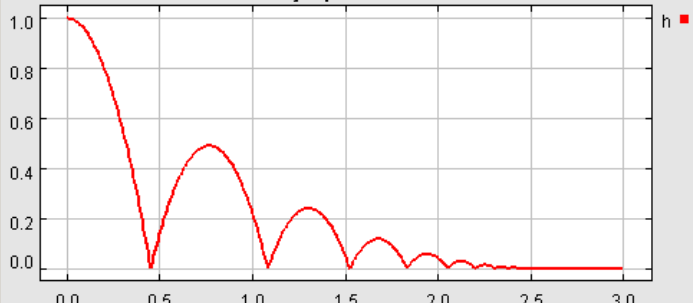
>> plot(h)
true

>>
```

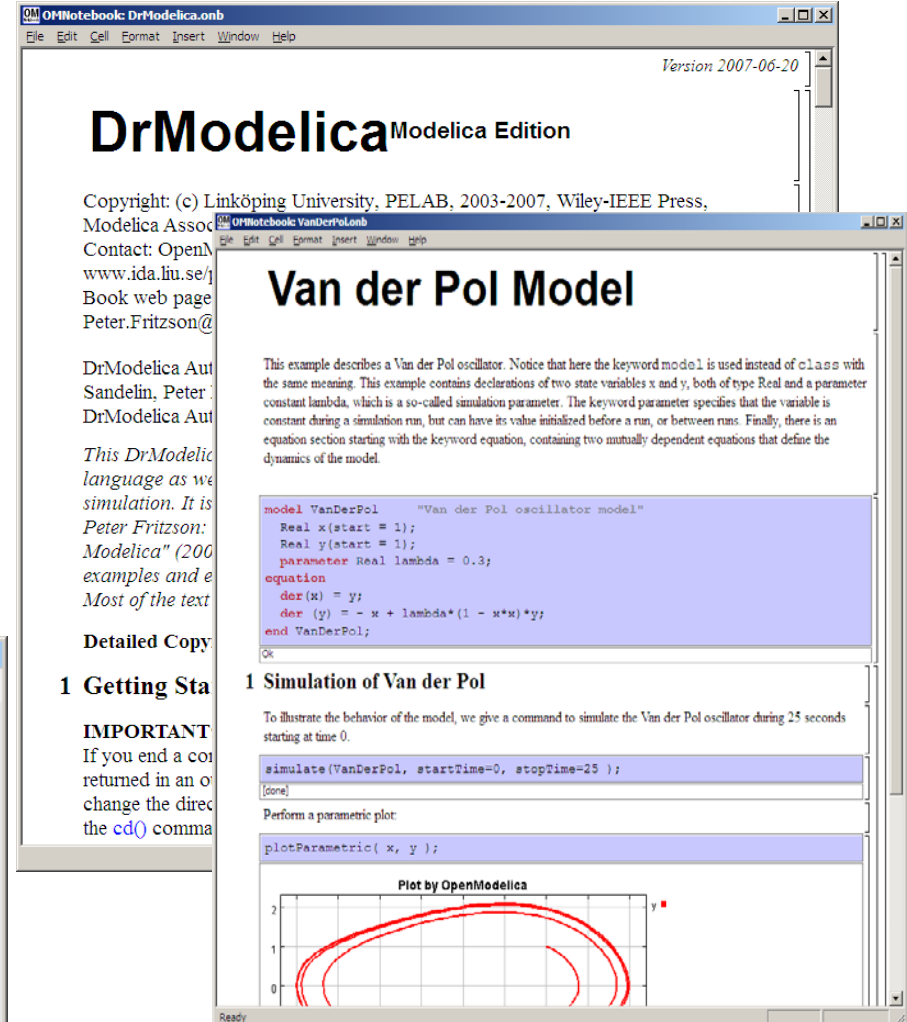
tmpPlot.plt

File Edit Special

Plot by OpenModelica



Time (s)	Height (h)
0.0	1.00
0.5	0.00
1.0	0.50
1.5	0.25
2.0	0.12
2.5	0.06
3.0	0.03



OMNotebook: DrModelica.onb

File Edit Cell Format Insert Window Help

Version 2007-06-20

DrModelica Modelica Edition

Copyright: (c) Linköping University, PELAB, 2003-2007, Wiley-IEEE Press,
Modelica Assoc. www.ida.liu.se/
Book web page
Peter.Fritzson@

Van der Pol Model

This example describes a Van der Pol oscillator. Notice that here the keyword `model` is used instead of `class` with the same meaning. This example contains declarations of two state variables `x` and `y`, both of type `Real` and a parameter constant `lambda`, which is a so-called simulation parameter. The keyword parameter specifies that the variable is constant during a simulation run, but can have its value initialized before a run, or between runs. Finally, there is an equation section starting with the keyword `equation`, containing two mutually dependent equations that define the dynamics of the model.

```
model VanDerPol "Van der Pol oscillator model"
  Real x(start = 1);
  Real y(start = 1);
  parameter Real lambda = 0.3;
equation
  der(x) = y;
  der(y) = -x + lambda*(1 - x*x)*y;
end VanDerPol;
```

Ok

1 Simulation of Van der Pol

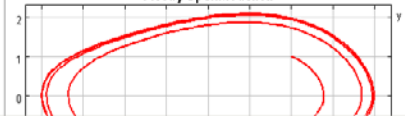
To illustrate the behavior of the model, we give a command to simulate the Van der Pol oscillator during 25 seconds starting at time 0.

```
simulate(VanDerPol, startTime=0, stopTime=25);
```

[done]

Perform a parametric plot:

```
plotParametric(x, y);
```



Plot by OpenModelica

x	y
1.0	1.0
0.0	1.5
-1.0	1.0
-0.5	0.0
0.0	-0.5
0.5	0.0
1.0	0.5
1.5	1.0

OMEdit- OpenModelica Connection Editor

The screenshot displays the OMEdit - OpenModelica Connection Editor window. The title bar shows the application name and standard window controls. The menu bar includes File, Edit, View, Simulation, FMI, Export, Debug, QMSimulator, Git, Tools, and Help. The toolbar contains various icons for file operations, simulation, and editing.

The Libraries Browser on the left side shows a tree view of available components. The 'Elementary' folder is expanded, and 'DoublePendulum' is selected. Other visible folders include OpenModelica, ModelicaReference, ModelicaServices, Complex, Modelica, UsersGuide, Blocks, ComplexBlocks, StateGraph, Electrical, Magnetic, Mechanics, MultiBody, and Utilities.

The main workspace displays a mechanical system diagram. It features a coordinate system with x and y axes and a 'world' frame with three downward-pointing arrows representing gravity. The system consists of two masses, 'boxBody1' and 'boxBody2', connected by a spring-damper mechanism. A damper component is connected to the first mass, with a damping coefficient $d=0.1$. The masses are represented by blocks with 'a' and 'b' ports. The first mass has a position vector $r=\{0.5, 0, 0\}$ and a normal vector $n=\{0, 0, 1\}$. The second mass has a position vector $r=\{0.5, 0, 0\}$ and a normal vector $n=\{0, 0, 1\}$.

The Messages Browser at the bottom shows tabs for All, Notifications, Warnings, and Errors, with no messages currently displayed.

- Implemented mainly in MetaModelica (401 packages) and a C/C++ runtime
- Is available as a dynamic library (faster than CORBA/ZMQ)
- Used from OMEdit, OMNotebook, OMShell, OMOptim, OMPython, MDT
- Automatically generated API that can be used from QT

Modelica->AST->SCode->DAE->C Code

// Parse the file and get an AST back

```
ast = Parse.parse(modelicaFile);
```

// Translate to simplified C code

```
scode = SCode.absyn2SCode(ast);
```

// flatten the simplified code

```
(cache, dae1) = Inst.instantiate(Env.emptyCache, scode);
```

// Call the function that optimizes the DAE

```
optimizeDae(scode, ast, dae, dae, lastClassName);
```

- OpenModelica
 - What is OpenModelica?
 - The past and present
- OpenModelica Technical Overview
 - OMC, OMShell, OMNotebook, OMEdit, ModelicaML, OMSimulator, OMPython, OMJulia, OMMatlab
- OpenModelica Development Environment
 - MetaModelica
 - The Eclipse Environment
- OpenModelica Latest Developments (2017-2018)

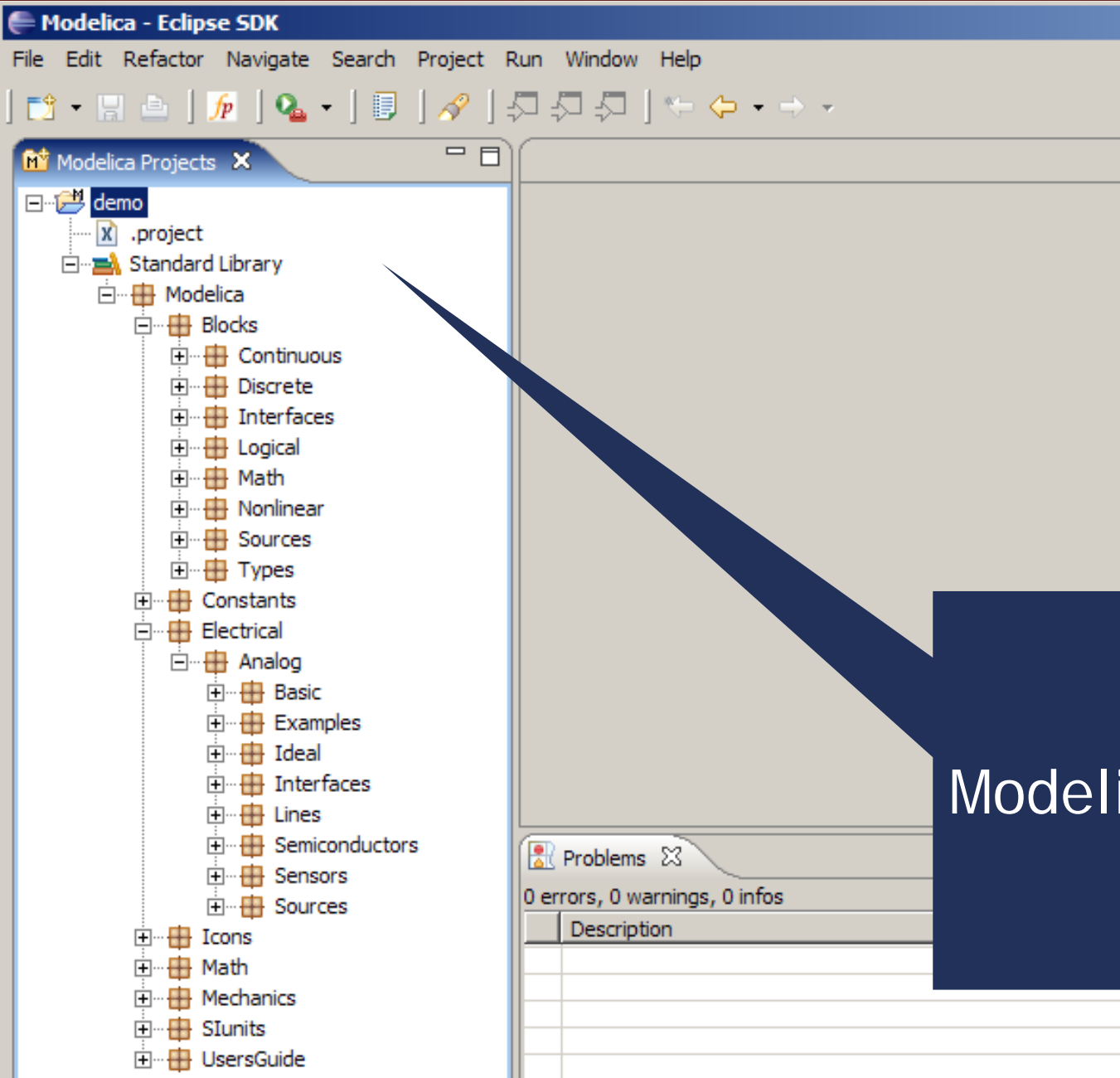
- **OMC**
 - Implemented mainly in MetaModelica and C/C++
- **Modelica**
 - classes, models, records, functions, packages
 - behavior is defined by equations or/and functions
 - equations
 - differential algebraic equations and conditional equations
- **MetaModelica extensions**
 - local equations
 - pattern equations
 - match expressions
 - high-level data structures: lists, tuples, option and uniontypes

MDT - Creating Modelica projects (I)

The screenshot illustrates the steps to create a Modelica project in Eclipse. The main window is titled "Modelica - Eclipse SDK" and shows the "File" menu with "New" > "Project..." selected. A "New Project" dialog box is open, titled "Select a wizard", with a tree view of wizards. The "Modelica" folder is expanded, and "Modelica Project" is selected. A "New Modelica Project" dialog box is also open, titled "Create a Modelica project", with the "Project name" field containing "demo".

Creation of Modelica projects using wizards

Creating Modelica projects (II)



Modelica project

Creating Modelica packages

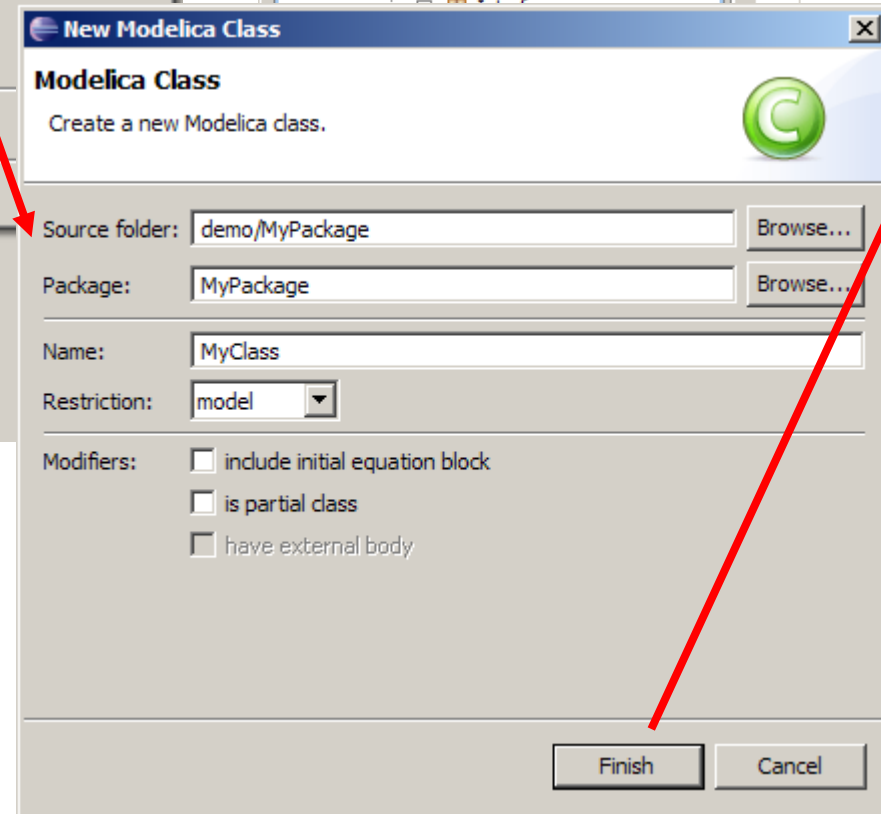
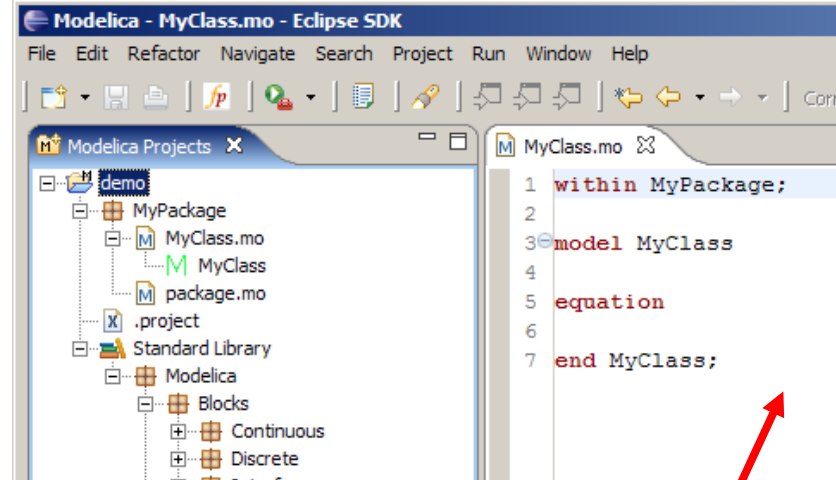
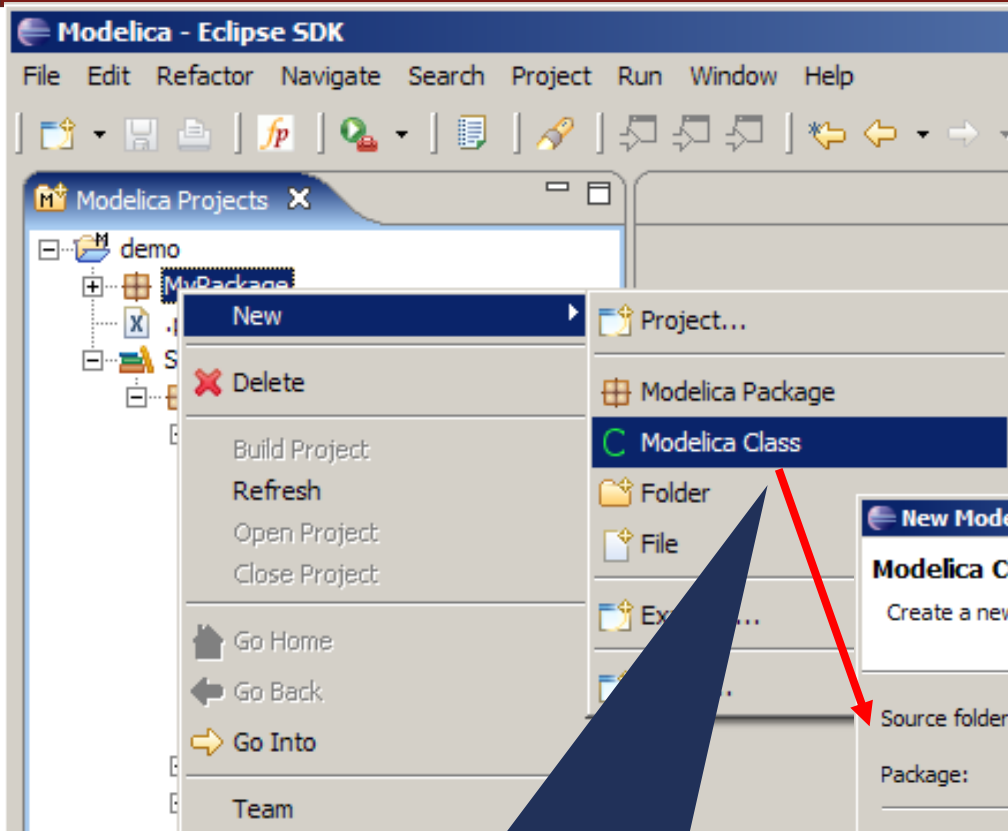
The image shows the Eclipse IDE interface for creating a Modelica package. The 'New' menu is open, and 'Modelica Package' is selected. The 'New Modelica Package' wizard is displayed, with the following fields filled:

- Source folder: demo
- Package: (empty)
- Name: MyPackage
- Description: A Modelica Package
- is encapsulated package

The 'Finish' button is highlighted with a red arrow, indicating the next step in the process. The 'Modelica Projects' view shows the resulting package structure, including 'MyPackage', 'package.mo', and '.project' files.

Creation of Modelica packages using wizards

Creating Modelica classes



Creation of Modelica classes, models, etc, using wizards

Code browsing

The screenshot displays the Eclipse IDE interface for a Modelica project. The left-hand side shows a project explorer with a tree view of the project structure. The 'VanDerPol' model is expanded, and the 'lambda' parameter is selected. The main editor window shows the source code for 'VanDerPol.mo', with the line defining 'lambda' highlighted. A callout box points to this line, explaining the code browsing feature.

```
1 // Van der Pol model
2
3 model VanDerPol "Van der Pol oscillator model"
4   Real x(start = 1);
5   Real y(start = 1);
6   parameter Real lambda = 0.3;
7   parameter Real e = Modelica.Constants.e;
8 equation
9   der(x) = y;
10  der(y) = - x + lambda*(1 - x*x)*y;
11 end VanDerPol;
12
```

Code Browsing for easy navigation within Modelica files. Automatic update on file save.

Error detection (I)

The screenshot shows the Eclipse IDE interface with the following components:

- Top Bar:** Title bar "Modelica - VanDerPol.mo - Eclipse SDK" and menu items: File, Edit, Refactor, Navigate, Search, Project, Run, Window, Help.
- Toolbar:** Includes icons for file operations and a "Correct Indentation" button.
- Left Panel (Project Explorer):** Shows a project structure with folders like "demo", "MyPackage", and "Standard Library".
- Editor Window:** Displays the code for "VanDerPol.mo". The code is as follows:

```
1 // Van der Pol model
2
3 model VanDerPol "Van der Pol oscillator model"
4   Real x(start = 1);
5   Real y(start = 1);
6   parameter Real lambda = 0.3;
7   parameter Real e = Modelica.Constants.e;
8 equation
9   der(x) = y;
10  der(y) = - x + lambda*(1 - x*x)*y;
11 end VanDerPol;
12
```
- Bottom Panel (Problems):** Shows "1 error, 0 warnings, 0 infos". A table lists the error:

Description	Resource	In Folder	Location
unexpected token: lambda, parsing resumed at token ';' on line 6, column 29	VanDerPol.mo	demo	line 6

Parse error
detection on
file save

Error detection (II)

The screenshot shows the Eclipse IDE interface for the Modelica project. The left sidebar displays a tree view of 'Modelica Projects' including folders like 'Compiler', 'absyn_builder', 'doc', 'modpar', 'omc_debug', 'omc_release', 'report', 'rml2mmo', 'rml2sig', 'runtime', 'scripts', 'test_codegen', 'tools', 'VC7', 'winruntime', and files like 'Absyn.mo', 'Algorithm.mo', 'Builtin.mo', 'Ceval.mo', 'ClassLoader.mo', 'Codegen.mo', 'Connect.mo', 'Corba.mo', 'DAE.mo', 'DAEEXT.mo', 'DAELow.mo', 'Debug.mo', and 'Derive.mo'. The main editor window shows the source code for 'Absyn.mo' with the following content:

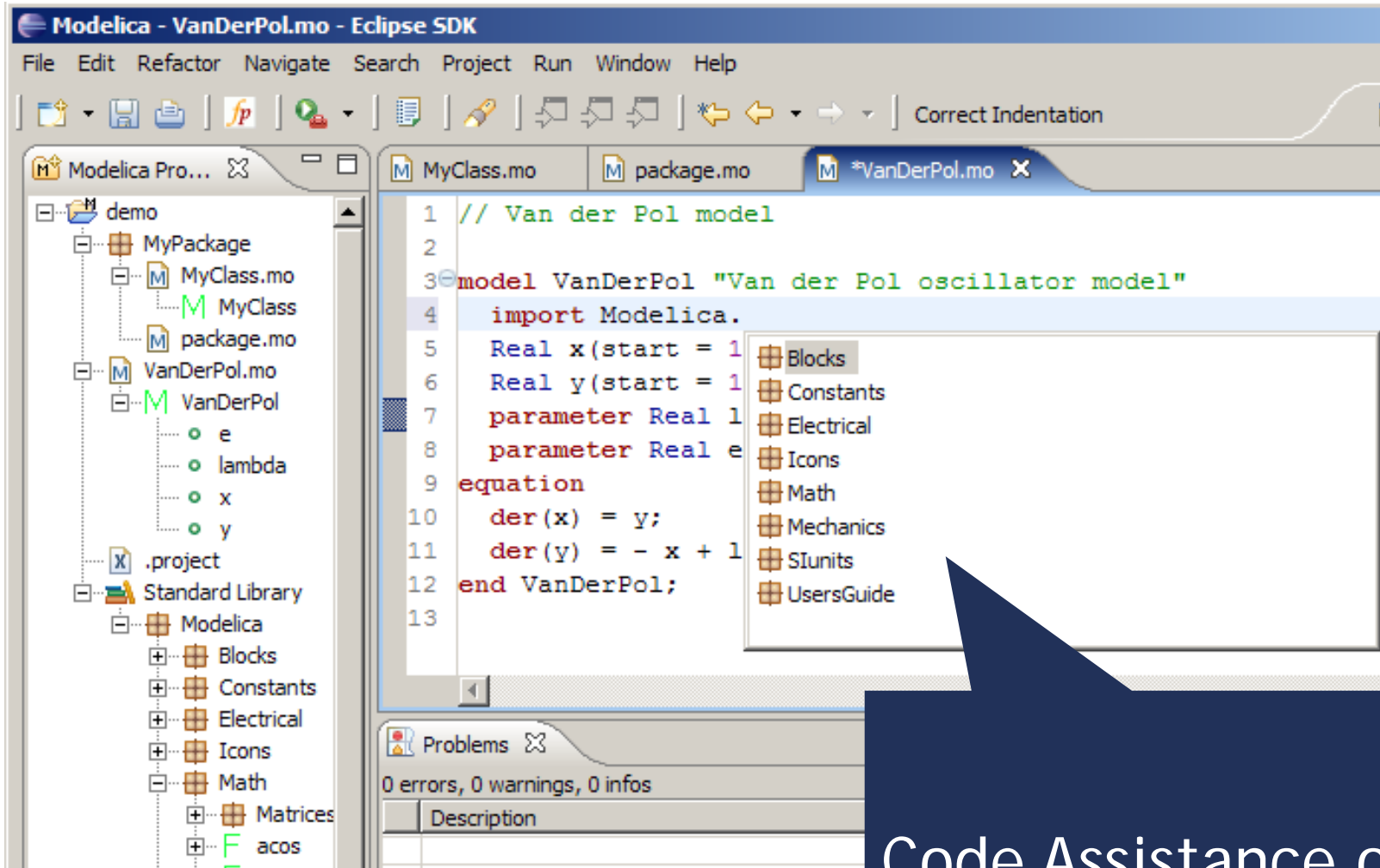
```
69 public
70 uniontype Program "- Programs, the top level construct
71 A program is simply a list of class definitions declared at top
72 level in the source file, combined with a within statement that
73 indicates the hieractical position of the program.
74 "
75 record PROGRAM
76 list<Class> classes "classes ; List of classes" ;
77 Withi within_ "within ; Within statement" ;
78 end PROGRAM;
```

The 'Problems' window at the bottom shows the error log with the following text:

```
<terminated> OMDev-MINGW-OpenModelicaBuilder [Program] c:\OMDev\tools\msys\bin\make.exe
cp -p ../Static.mo Static.mo
cp -p ../SimCodegen.mo SimCodegen.mo
cp -p ../Values.mo Values.mo
cp -p ../System.mo System.mo
/c/OMDev//tools/rml/bin/rmlc -v -Wc,-O3 -c Absyn.mo
"/c/OMDev//tools/rml//bin/rml" -Eplain Absyn.mo
Absyn.mo:77.5-77.9 Error: unbound type constructor Withi
Error: StaticElaborationError
make[2]: Leaving directory `/c/bin/mingw/home/...
make[1]: Leaving directory `/c/bin/cy.../home
make[2]: *** [Absyn.h] Error 1
make[1]: *** [omc_release] Error 2
make: *** [omc] Error 2
```

A blue callout box with a white arrow points to the error message in the console, containing the text: 'Semantic error detection on compilation'.

Code assistance (I)



The screenshot shows the Eclipse IDE interface with the following components:

- Project Explorer (Left):** Shows a project named 'demo' containing a package 'MyPackage' with files 'MyClass.mo', 'package.mo', and 'VanDerPol.mo'. The 'VanDerPol.mo' file is expanded to show a class 'VanDerPol' with parameters 'e', 'lambda', 'x', and 'y'.
- Editor (Center):** Displays the code for 'VanDerPol.mo'. The code is as follows:

```
1 // Van der Pol model
2
3 model VanDerPol "Van der Pol oscillator model"
4   import Modelica.
5   Real x(start = 1
6   Real y(start = 1
7   parameter Real l
8   parameter Real e
9   equation
10    der(x) = y;
11    der(y) = - x + 1
12 end VanDerPol;
13
```
- Code Assistance (Right):** A pop-up window is displayed over the 'import Modelica.' line, showing a list of available packages for import:
 - Blocks
 - Constants
 - Electrical
 - Icons
 - Math
 - Mechanics
 - SIunits
 - UsersGuide
- Problems (Bottom):** Shows '0 errors, 0 warnings, 0 infos'.

Code Assistance on imports

Code assistance (II)

The screenshot shows the Eclipse IDE with the following components:

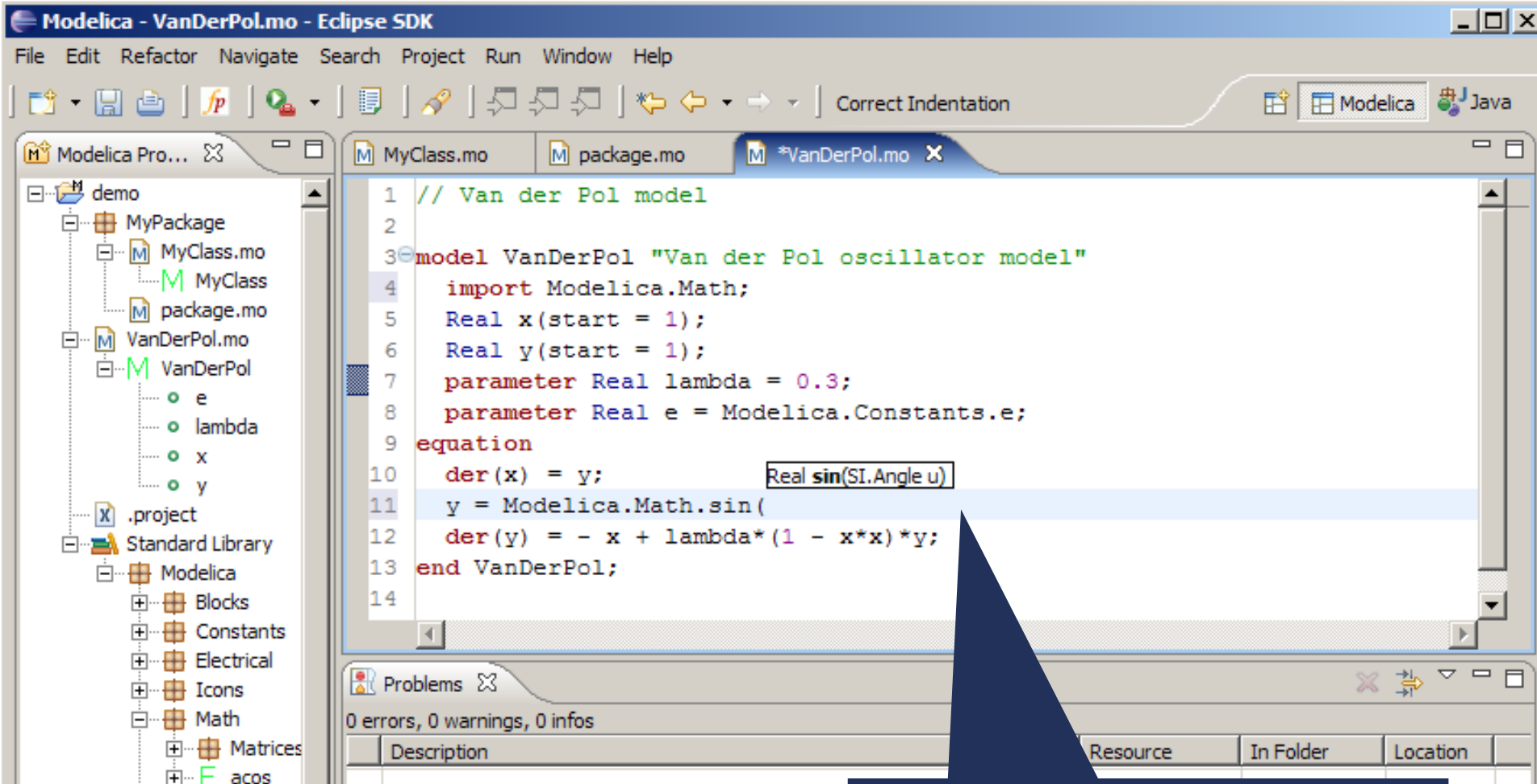
- Project Explorer:** Shows a project named 'demo' containing a package 'MyPackage' with files 'MyClass.mo', 'MyClass', and 'package.mo'. It also contains a file 'VanDerPol.mo' with a class 'VanDerPol' and parameters 'e', 'lambda', 'x', and 'y'. A 'Standard Library' is also visible with categories like 'Modelica', 'Blocks', 'Constants', 'Electrical', 'Icons', 'Math', 'Matrices', 'acos', and 'asin'.
- Editor:** Displays the code for 'VanDerPol.mo'. The code is as follows:

```
1 // Van der Pol model
2
3 model VanDerPol "Van der Pol oscillator model"
4   import Modelica.Math;
5   Real x(start = 1);
6   Real y(start = 1);
7   parameter Real lambda = 0.3;
8   parameter Real e = Modelica.Constants.
9 equation
10  der(x) = y;
11  der(y) = - x + lambda*(1 - x*x)*y;
12 end VanDerPol;
13
```

Line 8 is highlighted, and a code completion list is shown on the right, listing constants like 'c', 'D2R', 'e', 'eps', 'epsilon_0', 'G', 'g_n', 'h', and 'inf'. The 'e' constant is selected.
- Problems View:** Shows '0 errors, 0 warnings, 0 infos'.
- Table:** A table with columns 'Description', 'Resource', 'In Folder', and 'Location' is partially visible at the bottom.

Code Assistance on assignments

Code assistance (III)



Code Assistance on
function calls

Code indentation

```
1 // Van der Pol model
2
3 model VanDerPol "Van der Pol oscillator model"
4 import Modelica.Math;
5 Real x(start = 1);
6 Real y(start = 1);
7 parameter Real lambda = 0.3;
8 parameter Real e = Modelica.Constants.e;
9 equation
10   der(x) = y;
11   der(y) = - x + lambda*(1 - x*x)*y;
12 end VanDerPol;
```

Code Indentation

```
1 // Van der Pol model
2
3 model VanDerPol "Van der Pol oscillator model"
4 import Modelica.Math;
5   Real x(start = 1);
6   Real y(start = 1);
7   parameter Real lambda = 0.3;
8   parameter Real e = Modelica.Constants.e;
9 equation
10   der(x) = y;
11   der(y) = - x + lambda*(1 - x*x)*y;
12 end VanDerPol;
13
```

Code Outline and Hovering Info

The screenshot displays the Eclipse IDE with the following components:

- Project Explorer:** Shows a tree view of Modelica projects including rml2sig, runtime, scripts, test_codegen, tools, VC7, and various .mo files like Absyn.mo, Algorithm.mo, BuiltIn.mo, etc.
- Code Editor:** Displays the source code of Absyn.mo. A tooltip is shown over the function `getCrefFromExp`, providing its signature and description: "function getCrefFromExp 'function: getCrefFromExp Returns a flattened list of the component references in an expression'".
- Outline View:** Located at the bottom left, it shows a hierarchical outline of the code structure for Absyn, listing various algorithmic constructs like `ADD`, `ALG_ASSIGN`, `ALG_BREAK`, etc.
- Problems Console:** Located at the bottom center, it shows 113 errors, 0 warnings, and 0 infos. The error messages indicate issues with identifiers at the start and end of lines.

Identifier Info on Hovering

Code Outline for easy navigation within Modelica files

64M of 254M

Ctrl Contrib (Bottom)

Eclipse Debugging Environment

The screenshot displays the Eclipse IDE with the following components:

- Breakpoints Panel:** Shows a list of variables and their values. The variable `p` is of type `Absyn.Program` and contains a record with fields like `classes`, `classParts`, `contents`, `comment`, `info`, and `within_`.
- Console Panel:** Shows the output of the program, including the message "Parsed program".
- Outline Panel:** Shows the project structure, including the file `Bla.mo`.
- Code Editor:** Shows the source code of `Bla.mo`, with the `translateFile` function highlighted.

Name	Value	Declared Type
p	Absyn.Program	Absyn.Program
[record]	Absyn.PROGRAM[2]	((Absyn.Class list, Absyn.Within) :
classes	LIST	Absyn.Class list
[0]	Absyn.CLASS[7]	((string, bool, bool, bool, Absyn.R
name	"Bla"	string
partial_	false	bool
final_	false	bool
encapsulated_	false	bool
restriction	1:enum:Absyn.R_MODEL	Absyn.Restriction
body	Absyn.PARTS[2]	((Absyn.ClassPart list, string optio
classParts	LIST	Absyn.ClassPart list
[0]	Absyn.PUBLIC[1]	((Absyn.ElementItem list) => (Abs
contents	LIST	Absyn.ElementItem list
[0]	Absyn.ELEMENTITEM[1]	((Absyn.Element) => (Absyn.Elen
comment	NONE[0]	string option
info	Absyn.INFO[6]	((string, bool, int, int, int, int) =>
within_	Absyn.TOP[0]	Absyn.Within
f	string	string
->	"Bla.mo"	string

- Type information for all variables
- Browsing of complex data structures
- GDB based

OMEdit Debugging Environment

The screenshot displays the OMEdit - Transformational Debugger interface. The main window title is "OMEdit - Transformational Debugger" and the file path is "C:/Users/adeas31/AppData/Local/Temp/OpenModelica/OMEdit/Debugging.SolverFailure.NonlinearSolverSimulation_info.xml".

Variables Browser: Shows a list of variables with columns for Variables, Comment, Line, and Location. Variables include A, Kv, T0, T1, and Tref.

Equations Browser: Shows a list of equations with columns for Index, Type, and Equation. Equations include initial assignments and a nonlinear equation.

Defined In Equations: A table showing variables defined in equations:

Index	Type	Equation
1	initial	(assignment) ...*(T0 - Tref)
28	parameter	(assignment) ...*(T0 - Tref)

Used In Equations: A table showing variables used in equations:

Index	Type	Equation
1	initial	(assignment) ...*(T0 - Tref)
28	parameter	(assignment) ...*(T0 - Tref)

Equation Operations: Shows operations for the variable h0:

```
solved: h0 = cp * (T0 - Tref)
```

Source Browser: Displays the source code for "C:/Users/adeas31/Desktop/Debugging.mo". The code includes parameter declarations and equations for enthalpy computation, pump discharge pressure, and reservoir level.

```
enthalpy computation";
parameter
SI.SpecificHeatCapacity
cp=4186 "Cp of the fluid";
SI.MassFlowRate w_pump
"Mass flow rate from the
pump";
SI.Pressure p1 "Pump
discharge pressure";
SI.Pressure p2 "Storage
tank inlet pressure";
SI.Pressure dp_pump
"Pump dp";
SI.Pressure dp_valve
"Valve dp";
Real sqrt_dp
"Regularized sqrt(dp)";
SI.SpecificEnthalpy h0
"Pump inlet specific
enthalpy";
SI.SpecificEnthalpy h1
"Pump discharge specific
enthalpy";
SI.Power W;
SI.Length y(start=40,
fixed=true) "Reservoir
level";
Real eta(final
unit="1") = (p1 -
patm)*w_pump/rho/W "Pump
efficiency";
SI.Temperature T1 "Pump
discharge temperature";
SI.Time tau=1 "Time
constant of temperature
sensor";
equation
dp_pump = p1 - patm
dp";
```

Tutorial 1 - tomorrow at ModProd 2019!

- OpenModelica
 - What is OpenModelica?
 - The past
- OpenModelica Technical Overview
 - OMC, OMShell, OMNotebook, OMEdit, ModelicaML, OMSimulator, OMPython, OMJulia, OMMatlab
- OpenModelica Development Environment
 - MetaModelica
 - The Eclipse Environment (MDT)
- OpenModelica Latest Developments (2018-2019)

Latest Developments (2018-2019) (I)

- 2018 - 2019 - focus on performance, scalability, bug fixes
- OMC & Clients
 - Performance & scalability improvements
 - Bug fixes to OMC, OMEdit, FMI
- OMSimulator
 - Combined FMI & TLM support, SSP support
 - OMEdit GUI support
- OMJulia
 - API to access OpenModelica from Julia
- General
 - From Feb 2018 - Feb 2019
 - 30+ contributors
 - 800 commits (OMCompiler)
 - 969 commits (OMSimulator)
 - 213 commits (OMEdit)
 - Releases 1.13.0, 1.13.1, 1.13.2

Latest Developments (2018-2019) (II)

■ New Front-End - status

- The new front-end ~90% complete, (see #4138 on Trac)
- 100+ times faster, 5+ times less memory consumption (no array expansions, no expansion of for loops in equations)
- The new front-end also brings better support for libraries
- Developed in line with MCP-0019: Flattening
- Currently 423/424 models from MSL 3.2.3 pass the new front-end
- Last year 107/387 models from MSL 3.2.3 passed the new front-end

■ New Front-End - remaining work

- Expandable connectors (add virtual nodes)
- Making the backend cooperate with the new way the DAE is produced
- Support for state machines
- (Support for MetaModelica)

Latest Developments (2018-2019) (III)

- OMEdit - better Modelica support
 - Much more stable OMEdit, a lot of bug fixes and new usability features
 - Auto completion support
 - Support for OMSimulator
- Redeclare and Replaceable Support
 - Support for redeclare/replaceable is implemented
 - Waiting for the new front-end to become mature enough so we don't frustrate users

- OMC / OMEdit - new API for instance hierarchy editing
 - Concept testing - work in progress
 - Use the new front-end to instantiate the Model
 - Give the instance tree to OMEdit, automatically generated C++ classes for walking the tree
 - Allow OMEdit to edit the instance tree directly
 - Propagate the instance tree edits to the top level class
 - Build a simulation from the changed instance tree
- Julia instead of MetaModelica?
 - Concept testing - work in progress
 - Change the entire compiler from MetaModelica to Julia
 - Benefits
 - Access to Julia libraries, graph and numerical algorithms, etc
 - Support for variable structure systems (call OpenModelica compiler at runtime if everything is implemented in Julia)

Thank You!

Questions?

asodja, sjoelund.se, sebco011, lochel, wbraun, niklwors, hubert.thieriot, petar, perost, Frenkel TUD, Unknown, syeas460, adeas31, ppriv, ricli576, haklu, dietmarw, levsu, mahge930, x05andfe, mohsen, nutaro, x02lucpo, florox, x06hener, x07simbj, stebr461, x08joekl, x08kimja, Dongliang Li, jhare950, x97davka, krsta, edgarlopez, hanke, henjo, wuzhu.chen, fbergero, harka011, tmtuomas, bjozac, AlexeyLebedev, x06klasj, ankar, kajny, vasaie_p, niemisto, donida, hkiel, darbr, otto@mathcore.com, Kaie Kubjas, x06krino, afshe, x06mikbl, leonardo.laguna, petfr, dhedberg, g-karbe, x06henma, abhinck, azazi, x02danhe, rruusu, x98petro, mater, g-bjoza, x02kajny, g-pavgr, x05andre, vaden, jansilar, ericmeyers, x05simel, andsa, leist, choeger, Ariel.Liebman, frisk, vaurich, mwalther, mtiller, ptauber, casella, vitalij, hkiel, jank, rfranke, mflehmg, crupp2, kbalzereit, marchartung, adrpo

OpenModelica Project

<http://www.OpenModelica.org>