

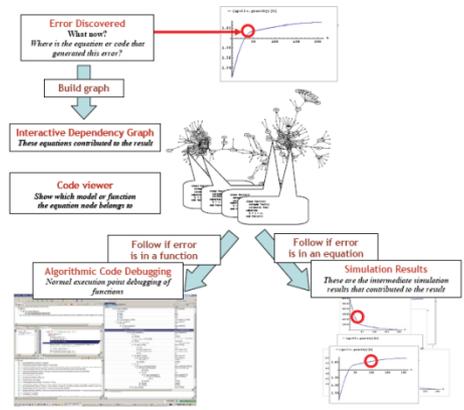
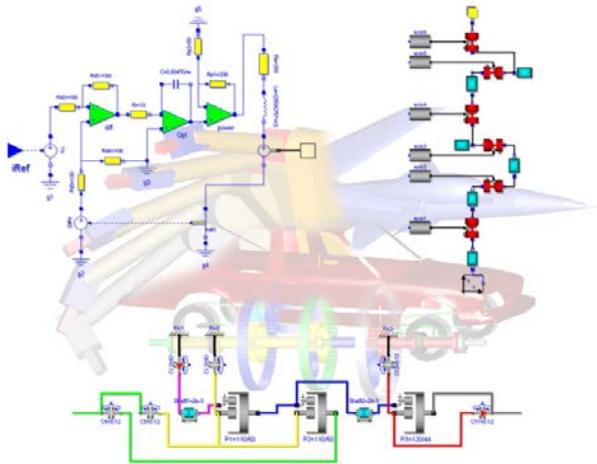
Technical Overview of OpenModelica and its Development Environment

Adrian Pop

2017-02-06

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

www.OpenModelica.org



$$\tau_2 = \frac{1}{k_2} \tau_1$$

$$e = \theta_{ref} - \theta_{out}$$

$$u = K \left(e + \frac{1}{T_I} \int_0^t e dt \right)$$

$$v = u \quad u_{ref} = R i \quad u_{emf} = k_1 \omega_{emf}$$

$$J_1 \frac{d^3 \theta_1}{dt^3} = \tau_{emf} + \tau_1$$

$$J_2 \frac{d^2 \theta_2}{dt^2} = \tau_2 + \tau_3$$

$$J_3 \frac{d^2 \theta_3}{dt^2} = -\tau_4 - \tau_{load}$$

$$v = u$$

$$\theta_2 = k_2 \theta_1$$

$$u_L = L \frac{di}{dt}$$

$$u = K \left(e + \frac{1}{T_I} \int_0^t e dt \right)$$

$$e = \theta_{ref} - \theta_{out}$$

$$v - u_R - u_L - u_{emf} = 0$$

$$u_{emf} = k_1 \omega_{emf} \quad i = \frac{1}{k_1} \tau_{emf}$$

$$\tau_2 = \frac{1}{k_2} \tau_1$$

$$\frac{J_1 - J_2 k_2^2}{k_2} \frac{d^2 \theta_1}{dt^2} = \tau_{emf} - k_2 \tau_3$$

- **OpenModelica**
 - What is OpenModelica?
 - The past
- **OpenModelica Technical Overview**
 - OMC, OMShell, OMNotebook,
 - OMEdit, ModelicaML
- **OpenModelica Development Environment**
 - MetaModelica (RML/OMC)
 - The Eclipse Environment (MDT)
- **OpenModelica Latest Developments (2016-2017)**

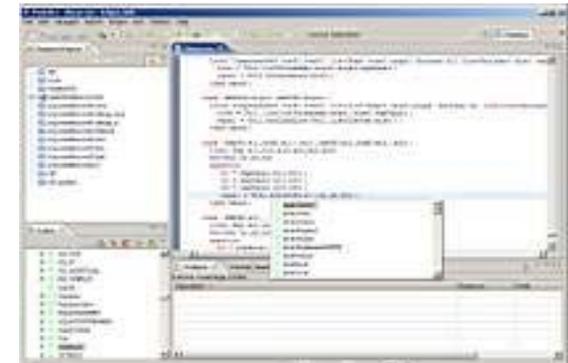
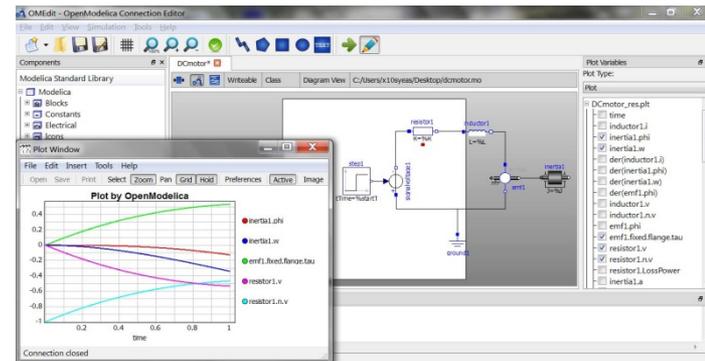
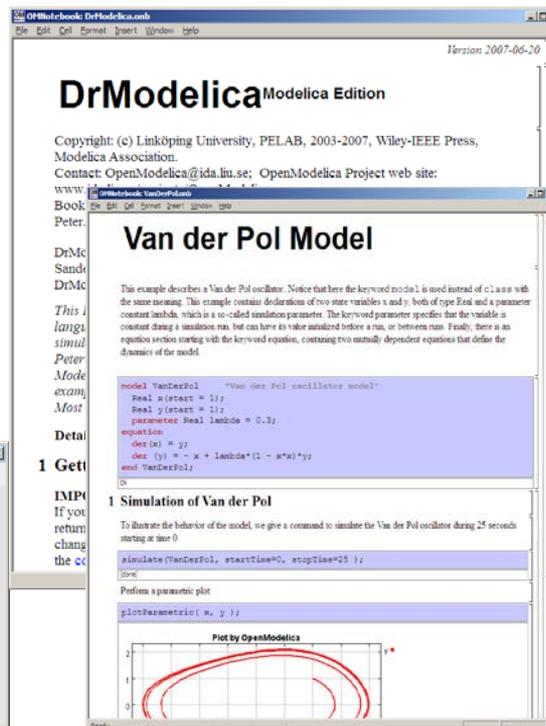
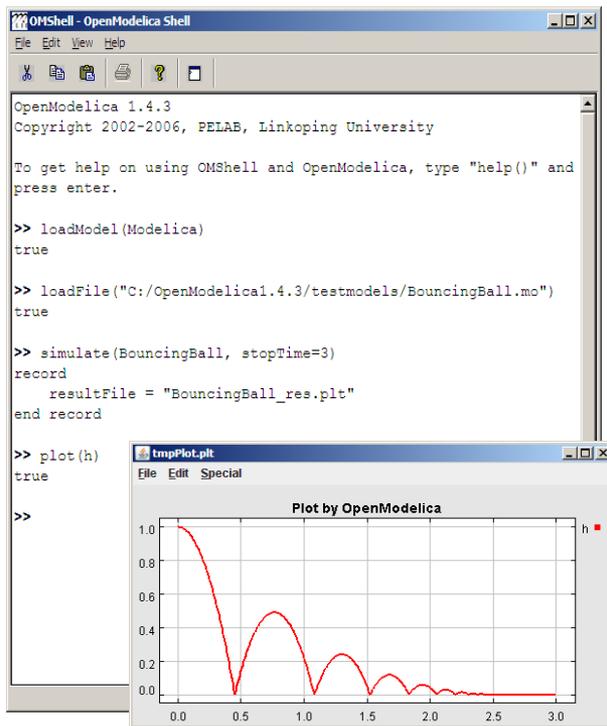
OpenModelica is ... its developers,
testers, bug reporters, contributors
and OSMC members

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, tmtuomas, 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, mflehmig, 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 - OpenModelica Python Environment
 - MDT - an advanced textual environment in Eclipse



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
 - *Slowly 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.

Updated an hour ago

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.

Updated an hour ago

OMLibraries

Updated 14 hours ago

People 21

Invite someone

OpenModelica

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

OpenModelica

WIKI Timeline Roadmap Browse Source View Tickets New Ticket Search Start Page Index History

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

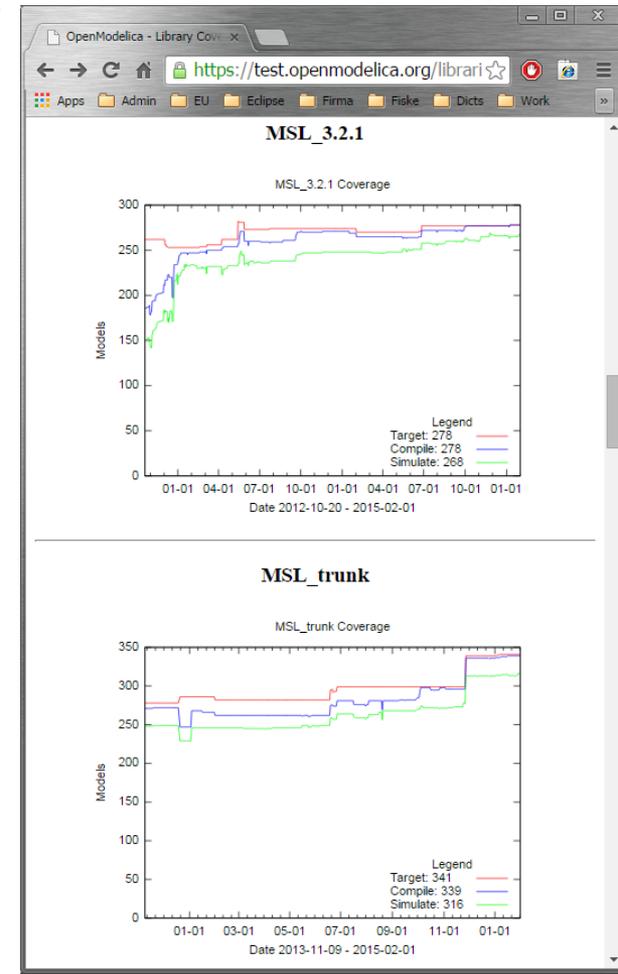
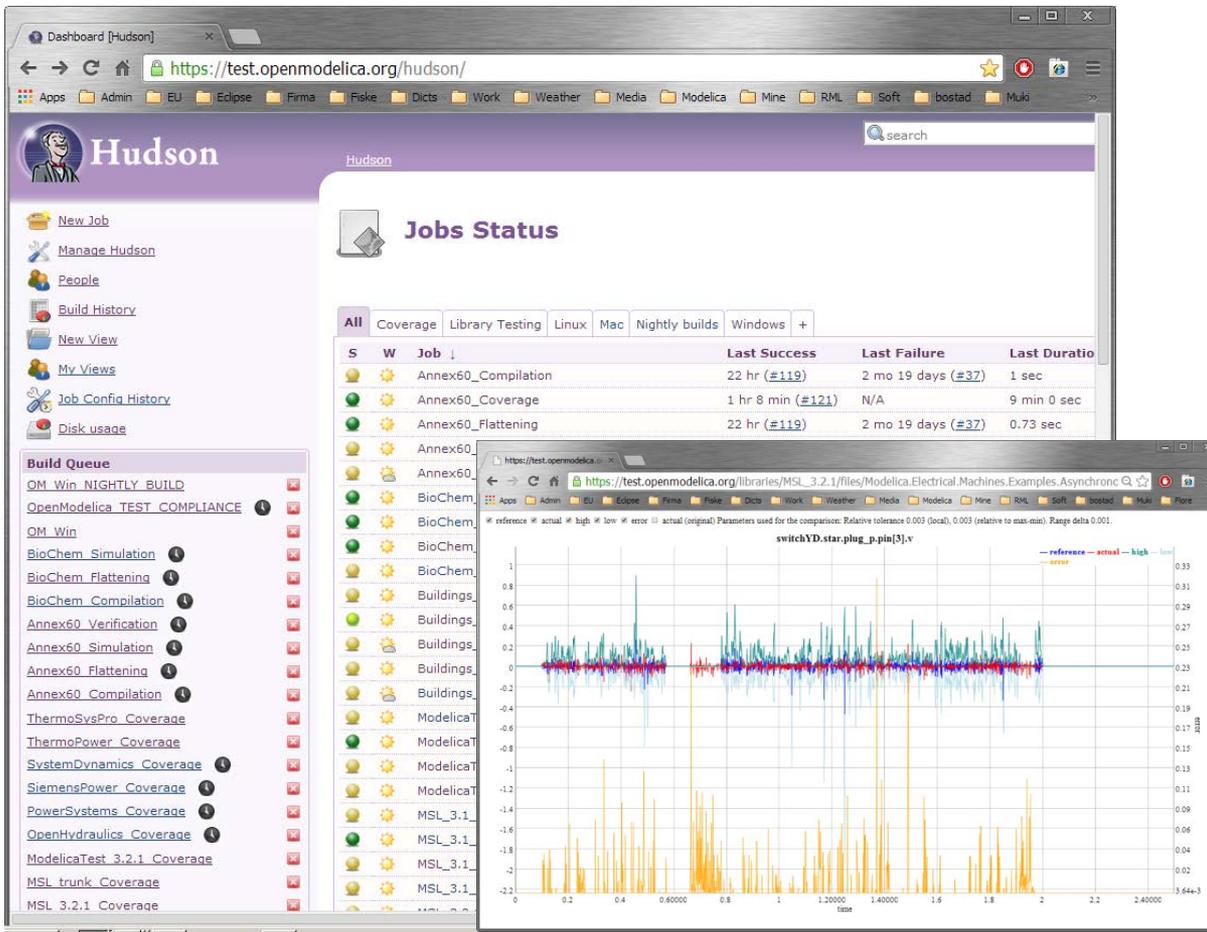
Legend

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

What is OpenModelica? (IV)

Open-source community services

- Extensive testing (unit & library coverage: MSL 3.2.x, ModelicaTest 3.2.x, PetriNet, Buildings, PowerSystems, OpenHydraulics, ThermoPower, and ThermoSysPro) with interactive result comparison
- ~3001 tests ran on each commit via Hudson (5 test servers currently)
 - Linux (GCC & CLANG), Windows (MinGW GCC), Mac OS (GCC)
- Automatic nightly builds for Window & Linux & Mac OS



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 2006-2016**
 - 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, InterCAX (MagicDraw SysML), 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

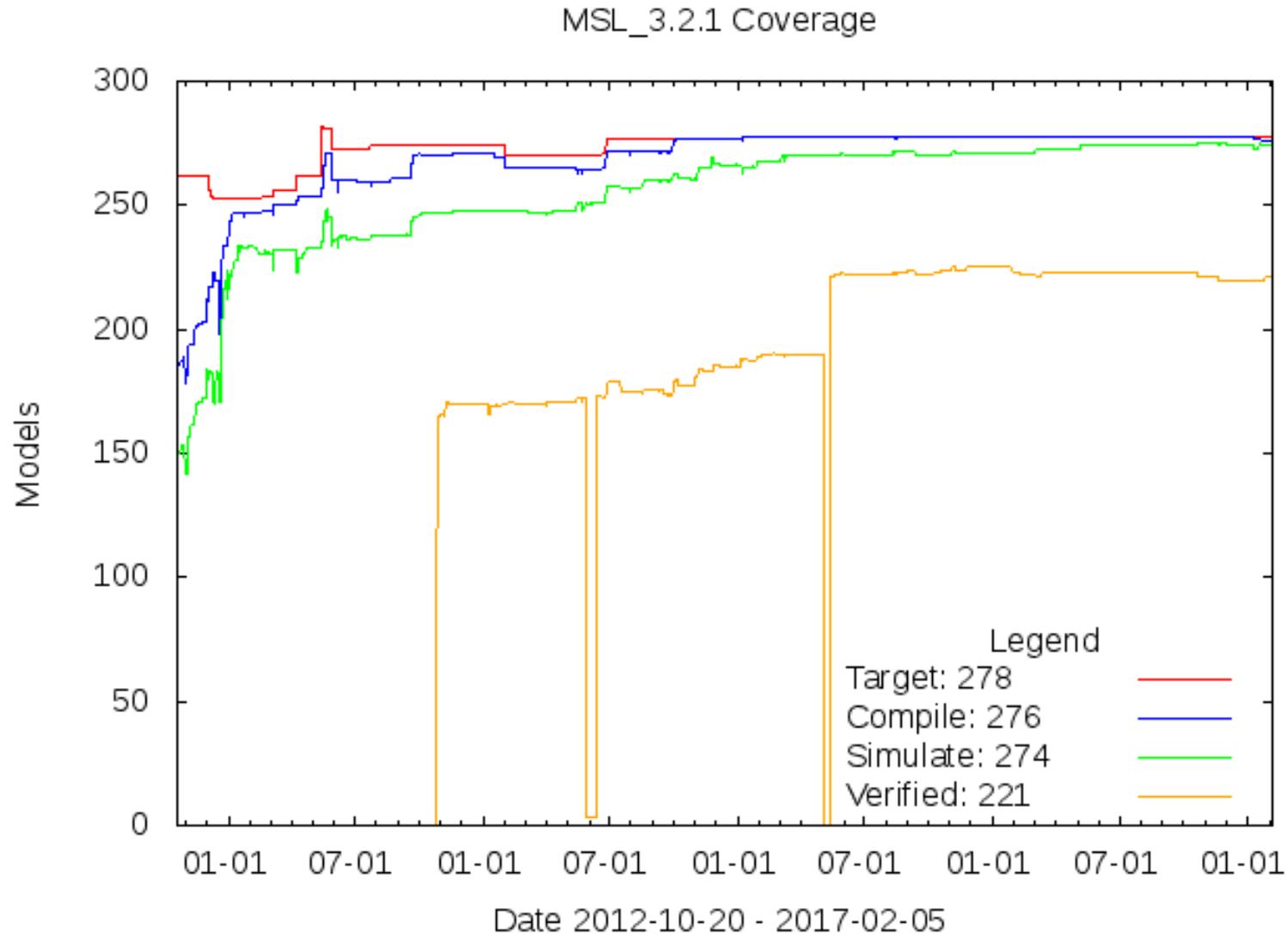
- Support for Modelica Standard Library 3.2.1 including Media & Fluid
- **Front-end**
 - Performance Enhancements
 - Media & Fluid work
 - Operator overloading
 - New instantiation module started
- **Back-end**
 - Modular back-end with more optimization modules (Jens, Willi, Martin)
 - New simulation 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 - 2016 - 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)

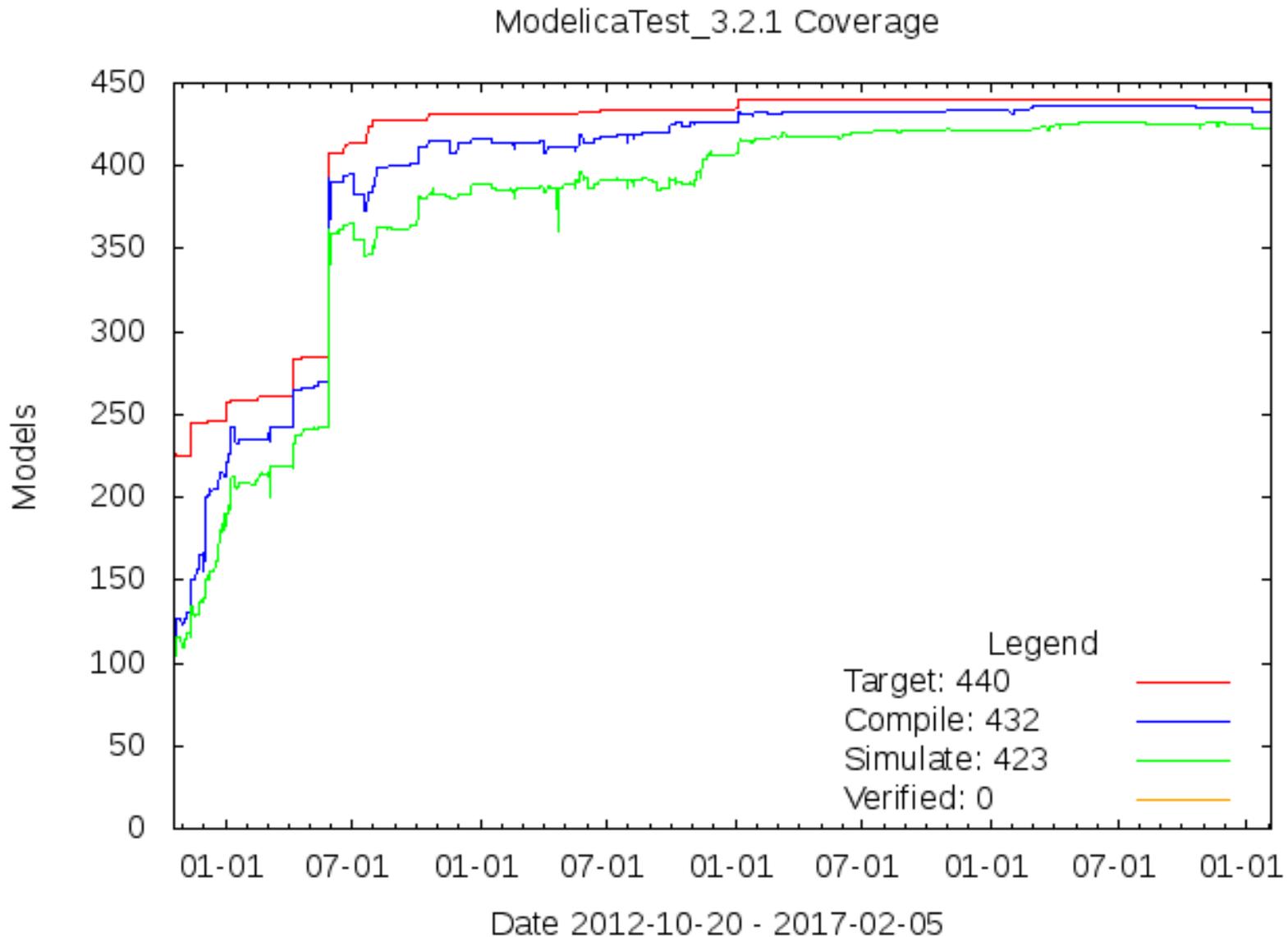
OpenModelica Testing (I)

- 2017-02-05 v1.12-dev - total 278 - build 276 (99%) - sim 274 (98%)



OpenModelica Testing (II)

- 2017-02-05 v1.12-dev - total 440 - build 432(97%) - sim 423 (96%)



- Moved the source code to github May 2015
 - Mature code base: <https://github.com/OpenModelica>
 - ~9000K lines of code and tests
-
- From Feb 2016 - Feb 2017
 - 60 contributors - up by 5 contributors
 - 1420 commits (OMCompiler)

OpenModelica Statistics (II)

Contributors

Traffic

Commits

Code frequency

Punch card

Network

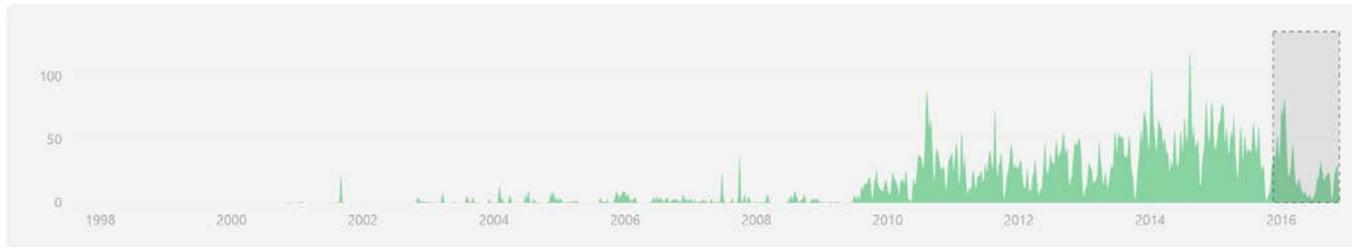
Members

Dependents

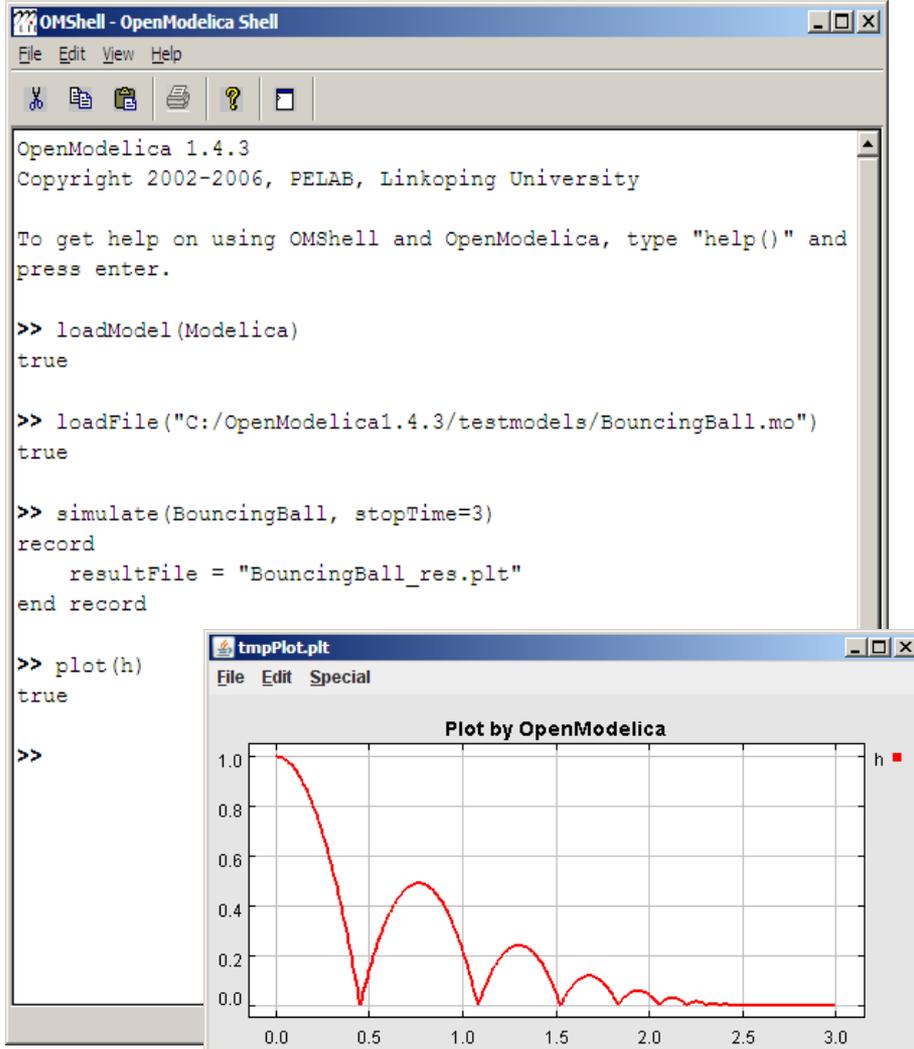
Feb 1, 2016 – Feb 5, 2017

Contributions: Commits ▾

Contributions to master, excluding merge commits



- OpenModelica
 - What is OpenModelica?
 - The past
- OpenModelica Technical Overview
 - OMC, OMShell, OMNotebook,
 - OMEdit, ModelicaML
- OpenModelica Development Environment
 - MetaModelica (RML/OMC)
 - The Eclipse Environment
- OpenModelica Latest Developments (2016-2017)



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

>>
```

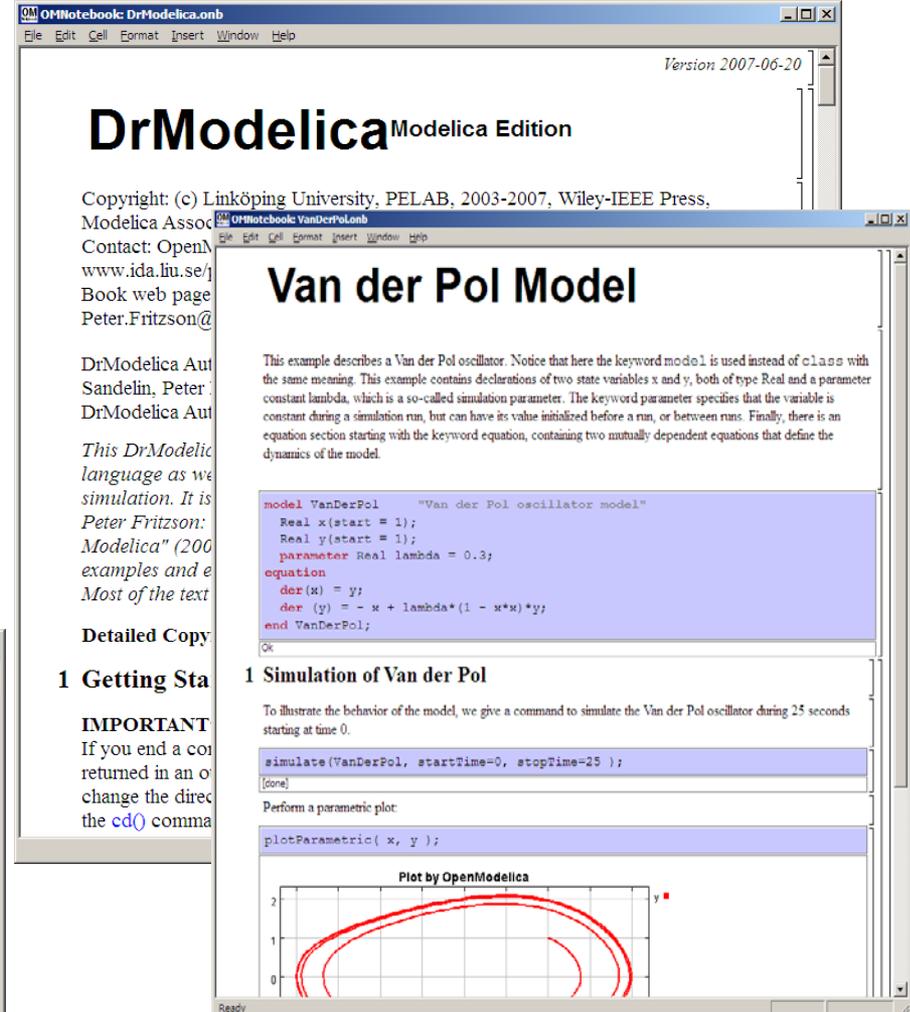
tmpPlot.plt

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	0.15
2.5	0.08
3.0	0.05



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 Getting Started

IMPORTANT
If you end a cell returned in an orange box, you can change the directory by using the `cd()` command.

1 Simulation of Van der Pol

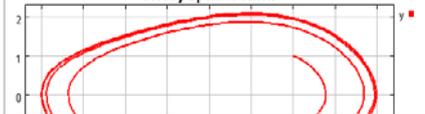
To illustrate the behavior of the model, we give a command to simulate the Van der Pol oscillator during 25 seconds starting at time 0.

```
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	1.0
1.0	0.0
0.0	0.0
1.0	0.0
0.0	0.0
1.0	0.0
0.0	0.0
1.0	0.0
0.0	0.0
1.0	0.0

OMEdit- OpenModelica Connection Editor

The screenshot displays the OMEdit - OpenModelica Connection Editor interface. The window title is "OMEdit - OpenModelica Connection Editor". The menu bar includes File, Edit, View, Simulation, FMI, Export, Tools, and Help. The toolbar contains various icons for file operations, simulation, and editing.

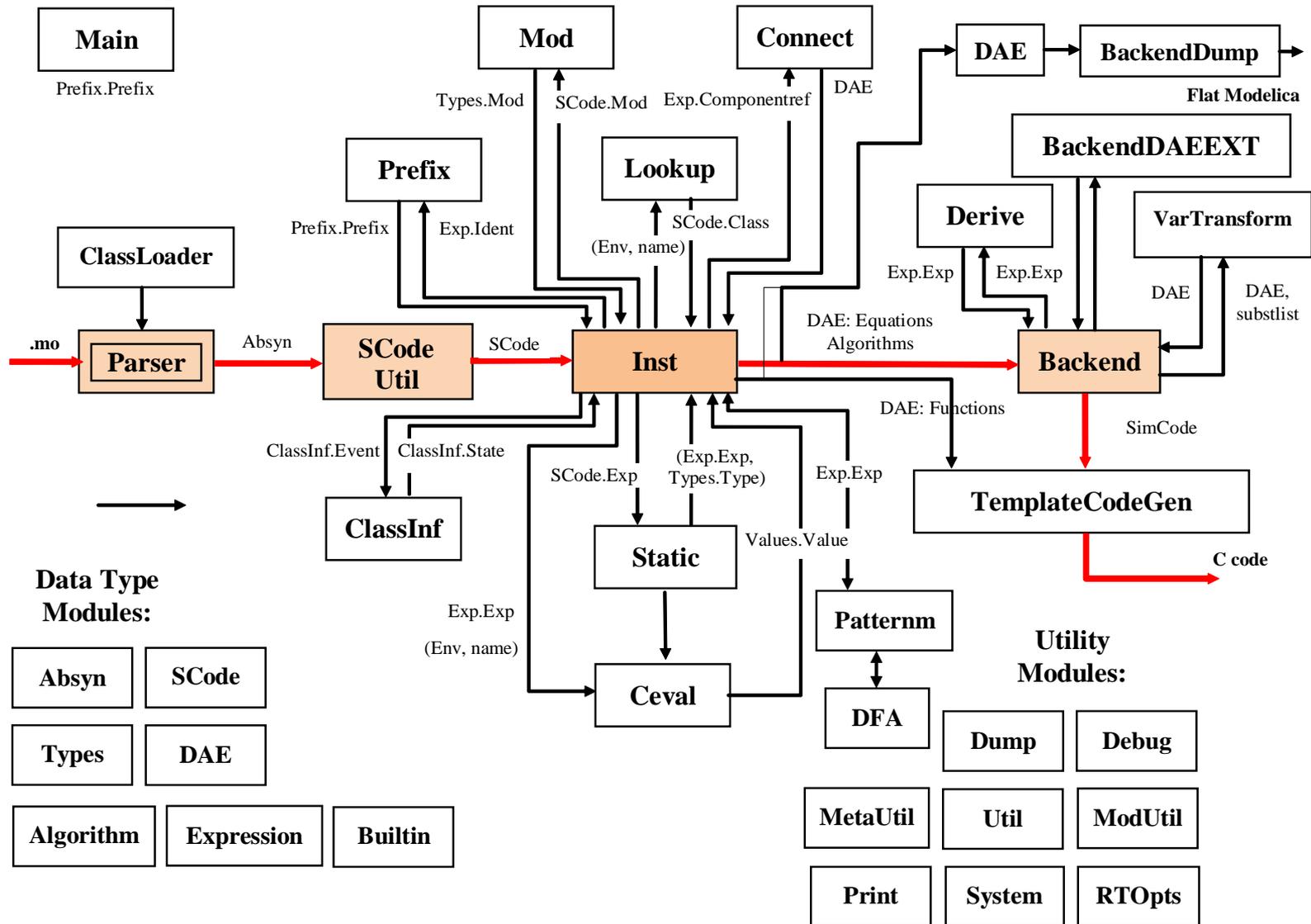
The Libraries Browser on the left shows a tree structure of libraries. The "Modelica" library is expanded, showing sub-libraries like "UsersGuide", "Blocks", "ComplexBlocks", "StateGraph", "Electrical", "Magnetic", "Mechanics", "MultiBody", "UsersGuide", "World", "Examples", and "Elementary". The "Elementary" sub-library is further expanded, showing "DoublePendulum" selected.

The main workspace displays a mechanical model of a double pendulum. The model consists of a "world" frame, a "damper" block with a damping coefficient $d=0.1$, and two "boxBody" blocks connected by revolute joints. The first revolute joint is labeled "revolute1" and the second is labeled "revolute2". The "boxBody1" and "boxBody2" blocks are positioned at $r=(0.5, 0, 0)$ relative to their respective joints.

The status bar at the bottom shows the current coordinates: X: -49.98, Y: 70.64. The status bar also includes buttons for "Welcome", "Modeling", and "Plotting".

The OMC Compiler

- Implemented mainly in MetaModelica and C/C++
- The compiler has 258 packages



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);
```

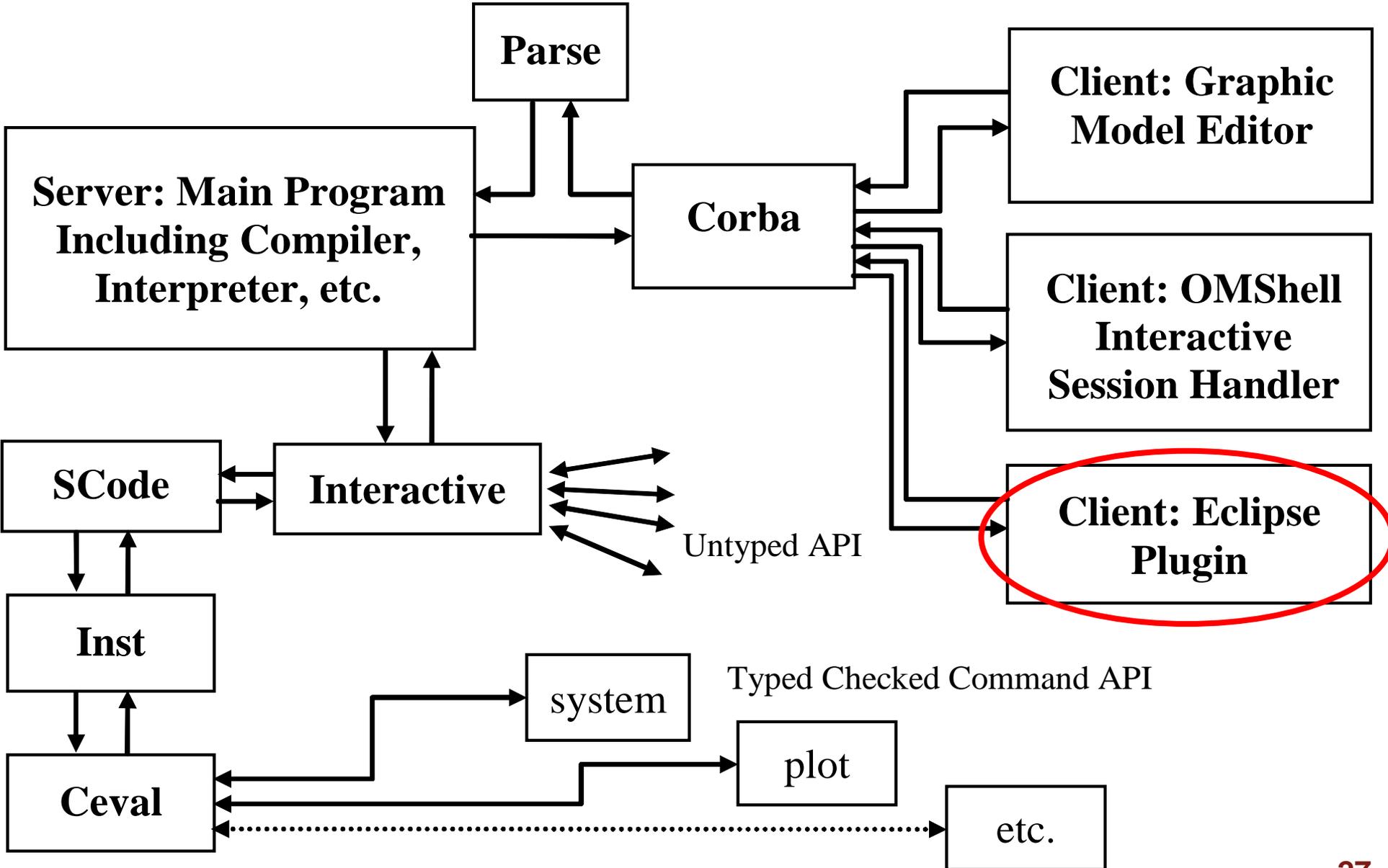
```
// Call the function that optimizes the DAE
```

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

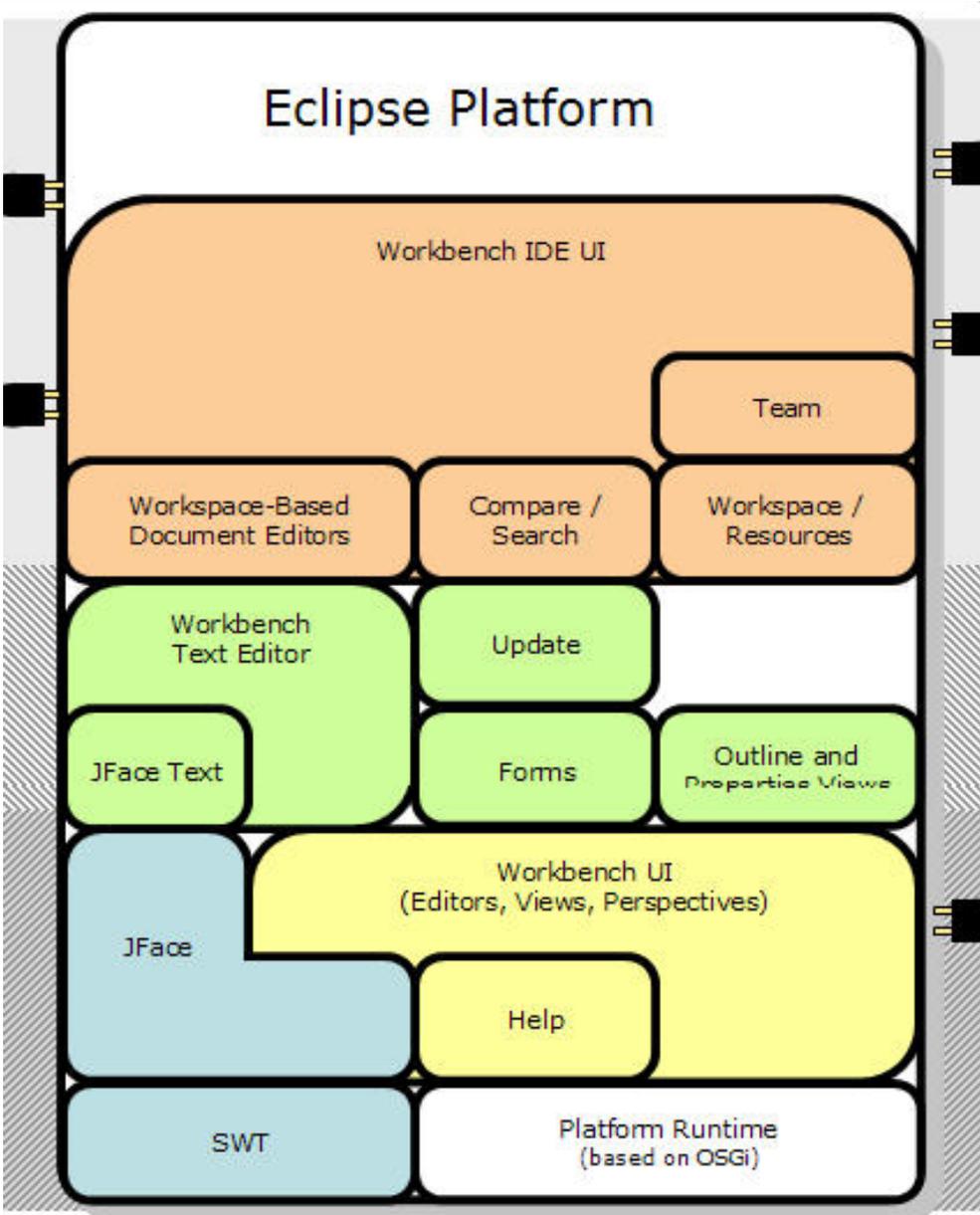
- OpenModelica
 - What is OpenModelica?
 - The past and present
- OpenModelica Technical Overview
 - OMC, OMShell, OMNotebook
 - OMEdit, ModelicaML, SimForge
- **OpenModelica Development Environment**
 - MetaModelica
 - The Eclipse Environment
- **OpenModelica Latest Developments (2016-2017)**

- **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

OpenModelica Context



The MDT Eclipse Environment (I)



Modelica Browser

Modelica Editor

Modelica Code Assistant

MetaModelica Debugging

Modelica Perspective

Creating Modelica projects (I)

The image shows the Eclipse IDE interface with the 'File' menu open and 'Project...' selected. A 'New Project' wizard dialog is open, showing a list of project types. The 'Modelica' folder is expanded, and 'Modelica Project' is selected. A 'New Modelica Project' dialog is also open, showing the project name 'demo' and navigation buttons.

File Edit Refactor Navigate Search Project Run Window Help

New Alt+Shift+N ▶ Project...
Open File...
Close Ctrl+F4
Close All Ctrl+Shift+F4
Save Ctrl+S
Save As...
Save All Ctrl+Shift+S
Revert
Move...
Rename... F2
Refresh F5
Convert Line Delimiters To
Print... Ctrl+P
Switch Workspace...
Import

New Project
Select a wizard
Create a new Modelica project.

Wizards:

- Plug-in Project
- C
- C++
- CVS
- Eclipse Modeling Framework
- EJB
- Functional Programming
- J2EE
- Java
- Modelica
 - Modelica Project
- Plug-in Development
- Simple
- Web
- Examples

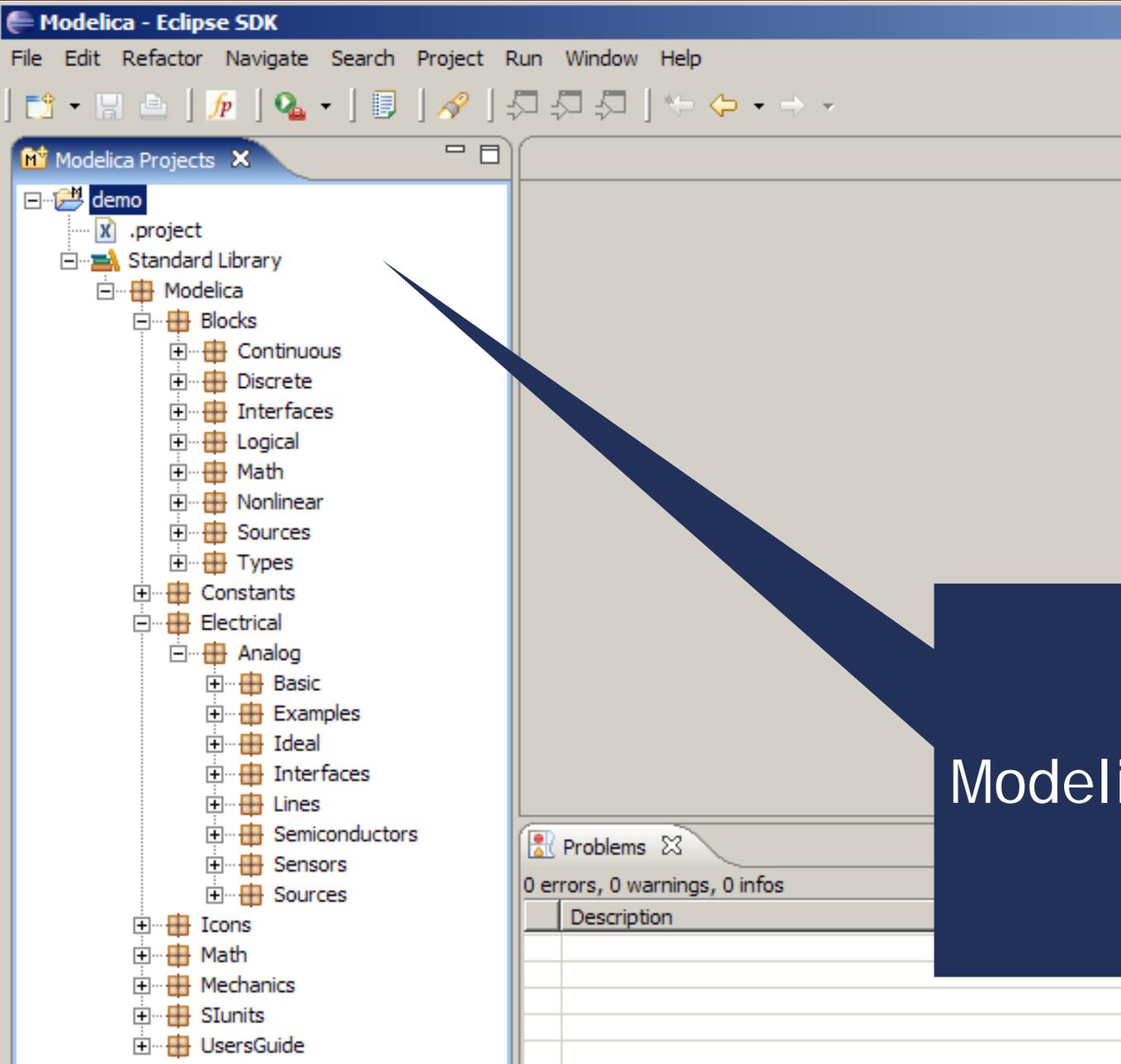
New Modelica Project
Create a Modelica project
Create a Modelica project in the workspace.

Project name: demo

< Back Next >
< Back Next > Finish Cancel

Creation of Modelica projects using wizards

Creating Modelica projects (II)



Modelica project

Creating Modelica packages

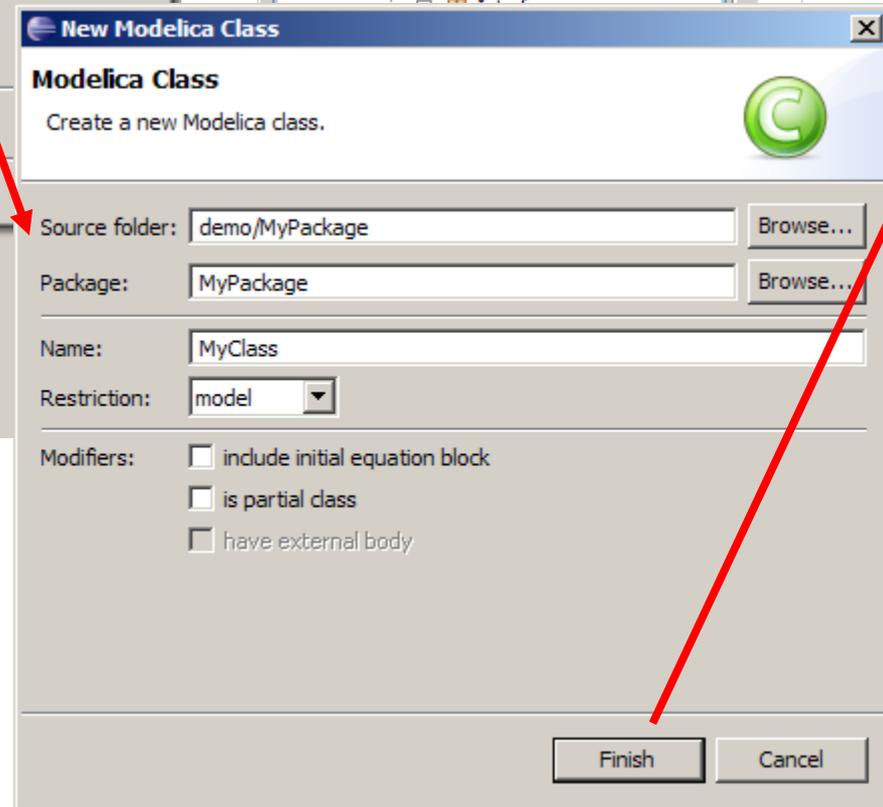
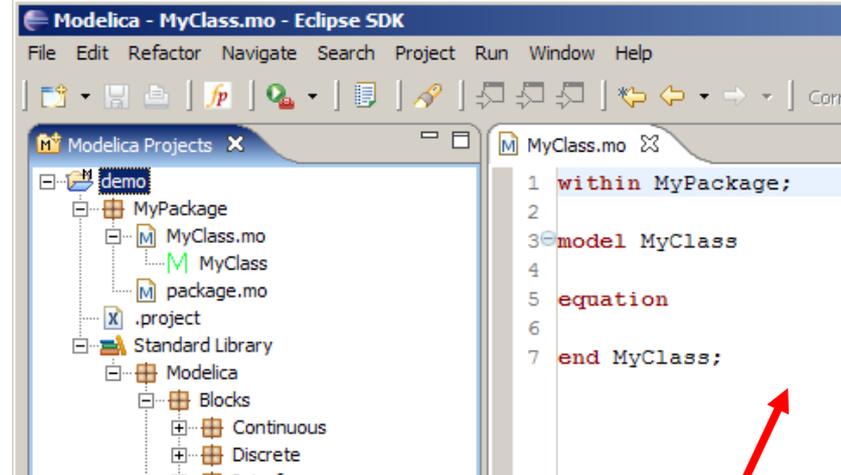
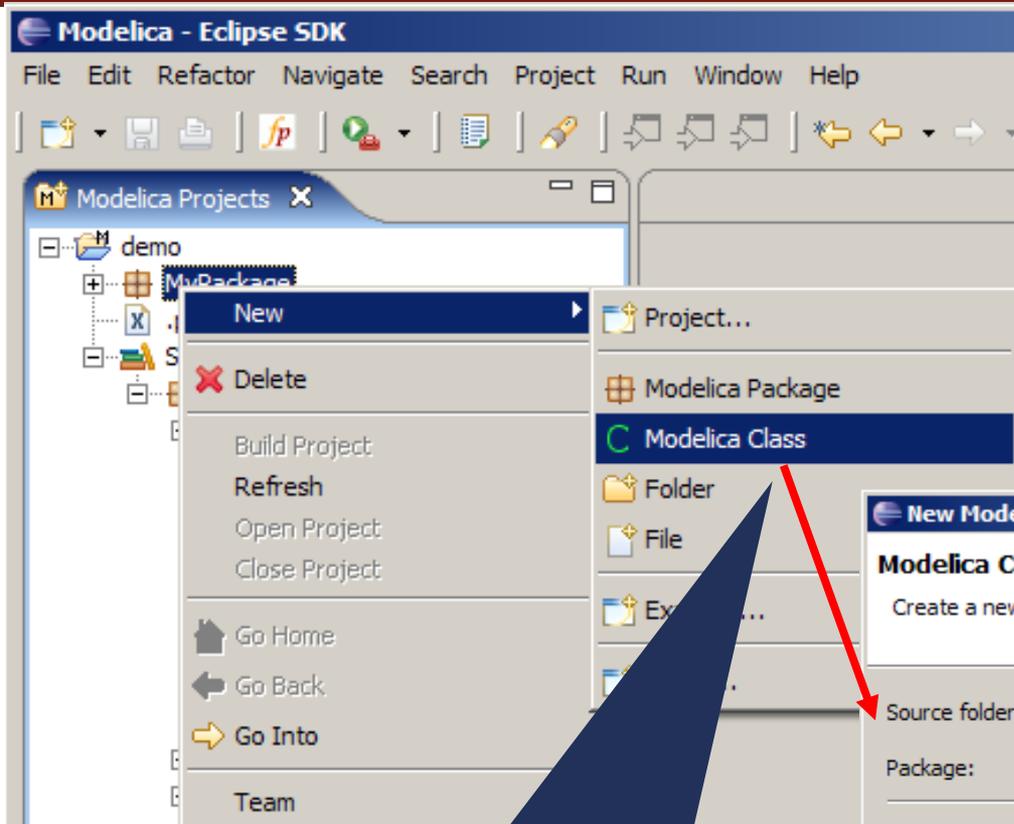
The image shows the Eclipse IDE interface for creating a new Modelica package. The 'New Modelica Package' wizard is open, displaying the following fields:

- 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.

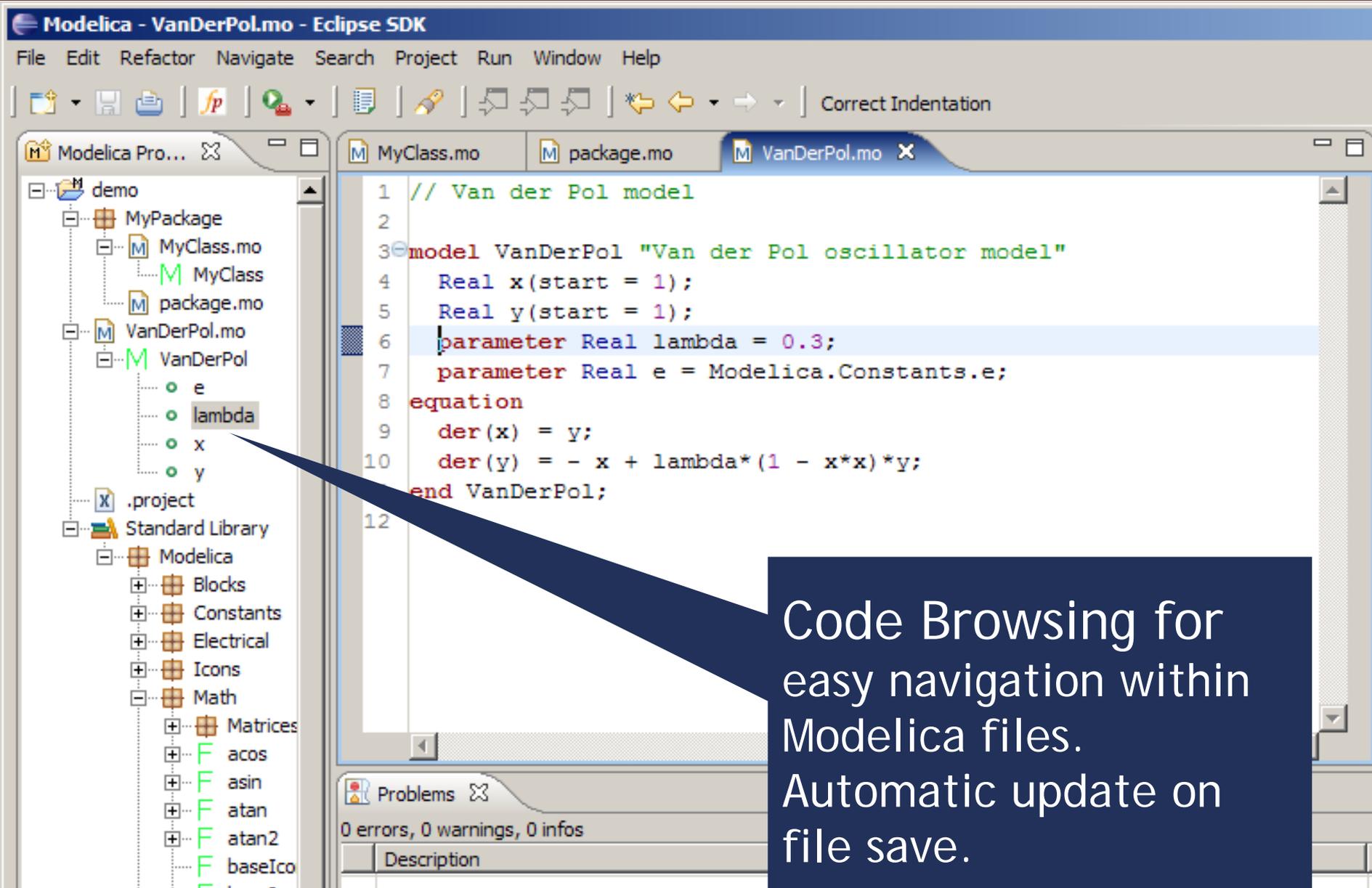
Creation of Modelica packages using wizards

Creating Modelica classes



Creation of Modelica classes, models, etc, using wizards

Code browsing



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

- Menu Bar:** File, Edit, Refactor, Navigate, Search, Project, Run, Window, Help.
- Toolbar:** Includes icons for file operations and a 'Correct Indentation' button.
- Project Explorer (Left):** Shows a project named 'demo' with a package 'MyPackage' containing files 'MyClass.mo', 'package.mo', and 'VanDerPol.mo'. The 'VanDerPol' model is expanded, showing parameters 'e' and 'lambda', and variables 'x' and 'y'. The 'lambda' parameter is selected, and a callout points to its definition in the code editor.
- Code Editor (Center):** Displays the source 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
```
- Problems View (Bottom):** Shows '0 errors, 0 warnings, 0 infos'.

A dark blue callout box contains the text: "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:

- Project Explorer:** Shows a project named 'demo' containing a package 'MyPackage' with files 'MyClass.mo', 'package.mo', and 'VanDerPol.mo'. The 'VanDerPol' model is expanded to show variables 'e', 'x', and 'y'.
- Code Editor:** Displays the content of '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
```

Line 6 is highlighted in blue, and a red 'X' icon is visible in the left margin next to it.
- Problems Window:** Located at the bottom, it displays the error message: "1 error, 0 warnings, 0 infos". The error table is as follows:

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. On the left, the 'Modelica Projects' view displays a tree structure of files, with 'Absyn.mo' selected. The main editor window shows the source code of 'Absyn.mo'. The code defines a 'Program' uniontype and a 'PROGRAM' record. Line 77 contains the line: `Withi within_ "within ; Within statement" ;`. A red 'X' icon is visible to the left of this line, indicating an error. Below the editor, the 'Problems' view shows the error message: `Absyn.mo:77.5-77.9 Error: unbound type constructor Withi`. The 'Console' view shows the compilation process, including the command `make` and the error messages: `make[2]: *** [Absyn.h] Error 1`, `make[1]: *** [omc_release] Error 2`, and `make: *** [omc] Error 2`. A blue callout box with a white arrow points to the error message in the Problems view, containing the text: **Semantic error detection on compilation**.

Code assistance (I)

The screenshot shows the Eclipse IDE with the following components:

- Project Explorer (Left):** A tree view showing 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):** The 'VanDerPol.mo' file is open, showing the following code:

```
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 dropdown menu is visible over the 'import Modelica.' statement, listing various Modelica packages: Blocks, Constants, Electrical, Icons, Math, Mechanics, SIunits, and UsersGuide.
- Problems (Bottom):** A panel showing '0 errors, 0 warnings, 0 infos'.

Code Assistance on imports

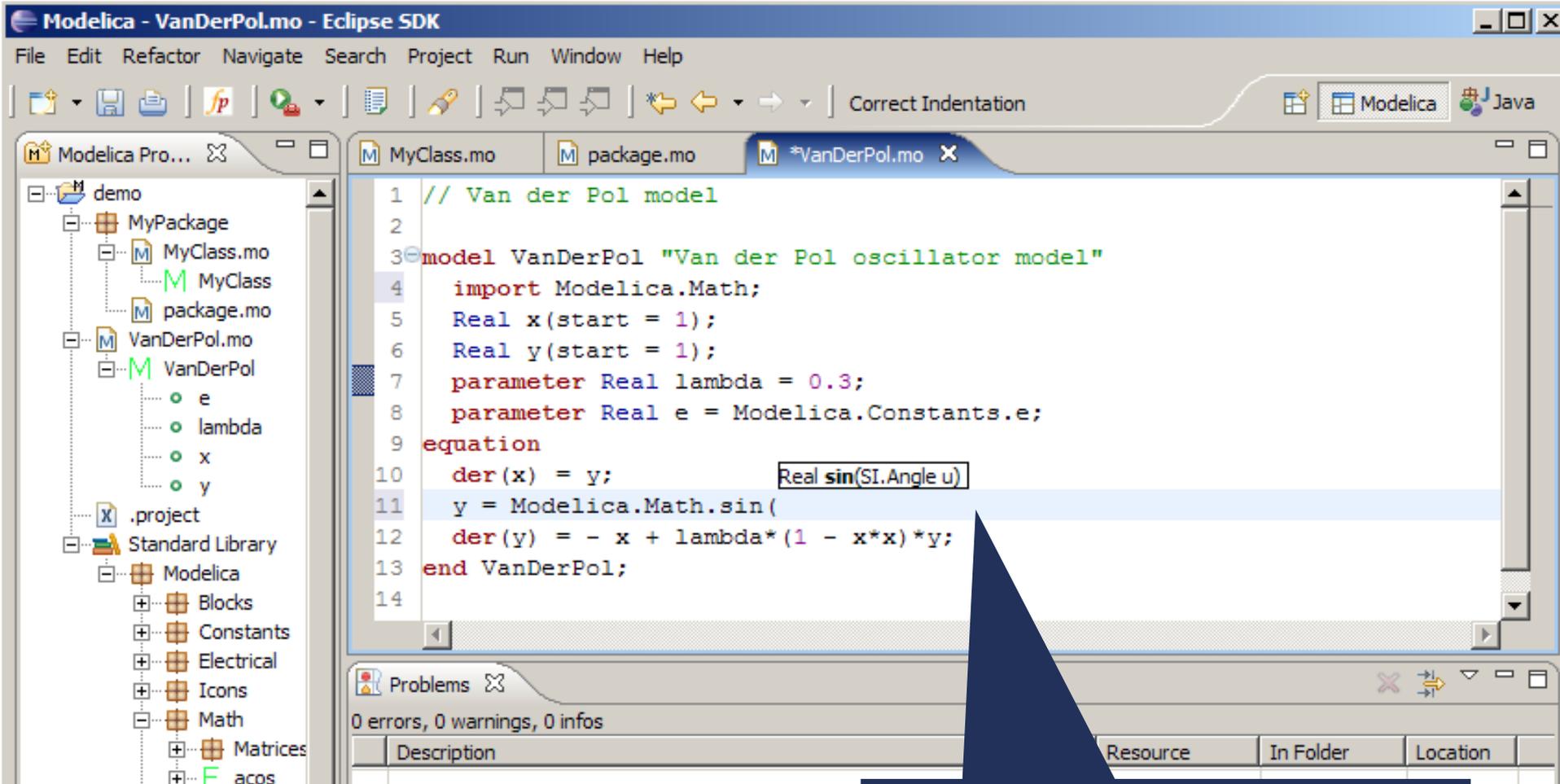
Code assistance (II)

The screenshot shows the Eclipse IDE with the following components:

- Project Explorer:** Shows a project named 'demo' with a package 'MyPackage' containing 'MyClass.mo', 'MyClass', and 'package.mo'. A 'VanDerPol.mo' file is also present, containing a 'VanDerPol' model with parameters 'e', 'lambda', 'x', and 'y'.
- Editor:** Displays the code for 'VanDerPol.mo'. The current line is:
`parameter Real e = Modelica.Constants.`
The IDE provides a list of suggestions for the variable 'e', including 'c', 'D2R', 'e', 'eps', 'epsilon_0', 'G', 'g_n', 'h', and 'inf'. The 'e' suggestion is highlighted.
- Problems View:** Shows '0 errors, 0 warnings, 0 infos'.
- Table:** A table with columns 'Description', 'Resource', 'In Folder', and 'Location' is visible at the bottom.

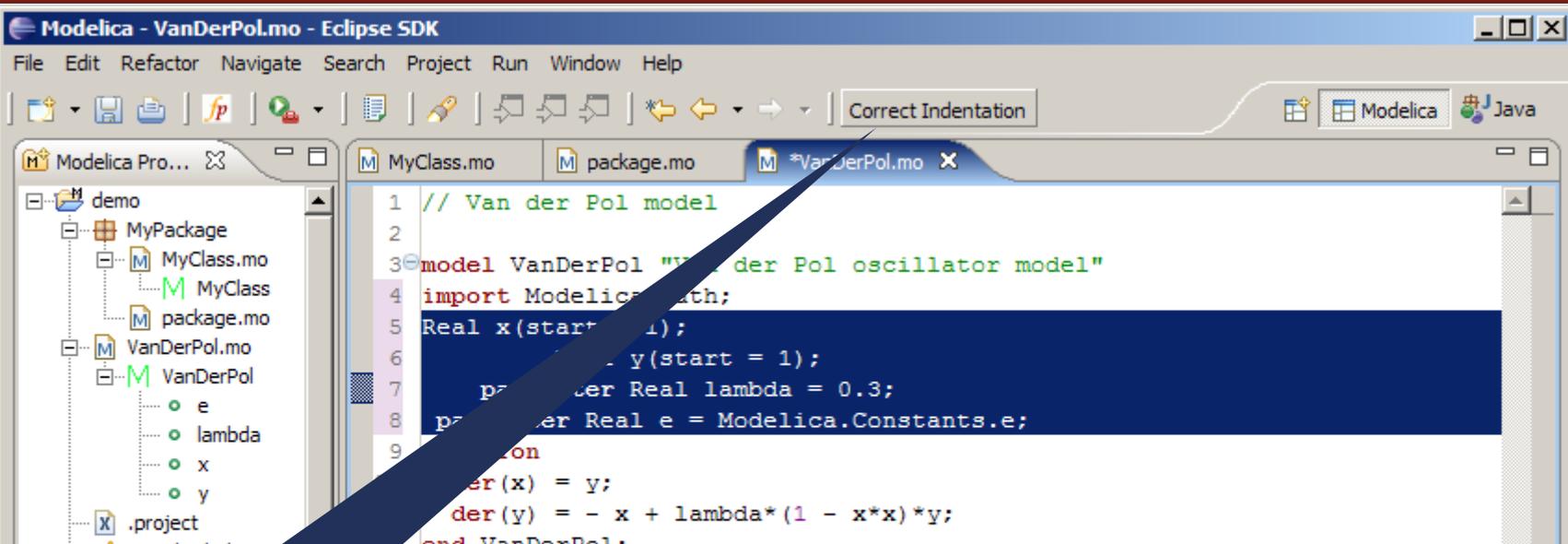
Code Assistance on assignments

Code assistance (III)



Code Assistance on
function calls

Code indentation

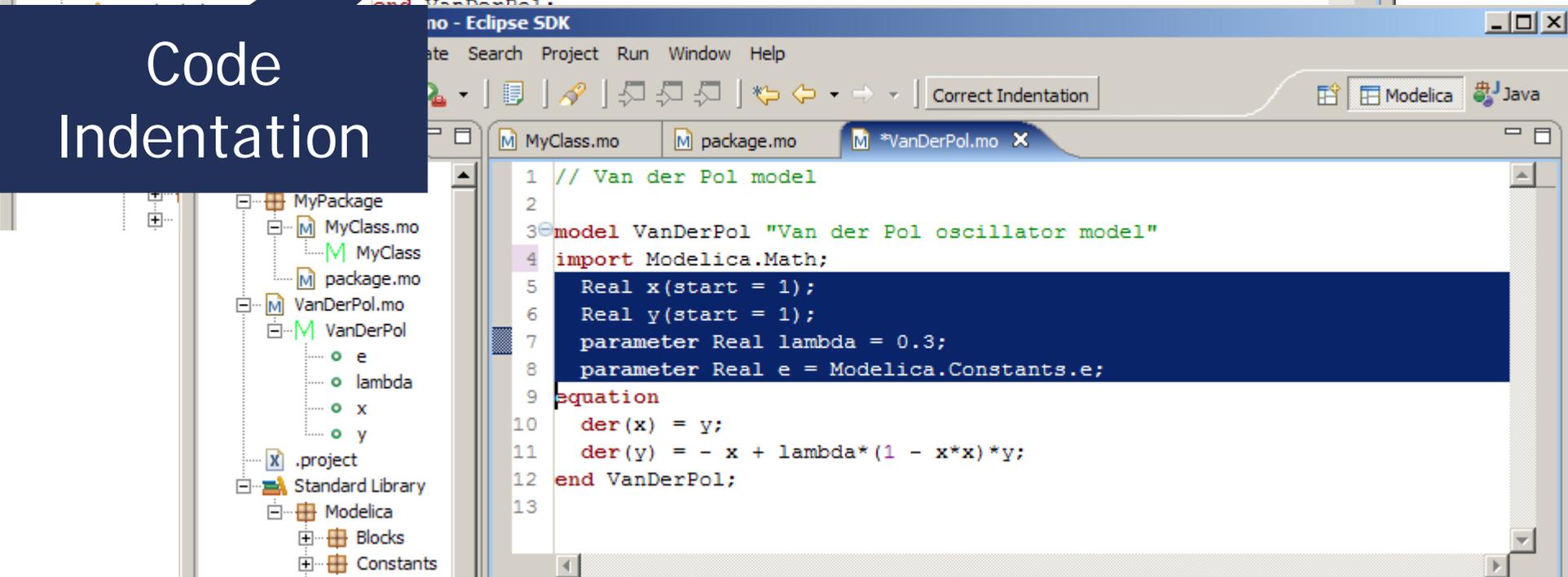


The screenshot shows the Eclipse IDE with a Modelica file named `VanDerPol.mo`. The code is as follows:

```
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.e;
9 equation
10   der(x) = y;
11   der(y) = - x + lambda*(1 - x*x)*y;
12 end VanDerPol;
```

The code is not properly indented. A blue arrow points from the text 'Code Indentation' to the code, highlighting the lack of consistent indentation for the `equation` block and its contents.

Code
Indentation



The screenshot shows the same Eclipse IDE with the `VanDerPol.mo` file, but now with consistent 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;
```

The code is now properly indented, with the `equation` block and its contents indented relative to the `model` block.

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 'Absyn.mo' which is currently selected.
- Code Editor:** Displays the source code for 'Absyn.mo'. A function definition for `getCrefFromExp` is highlighted, and a tooltip provides its description: "function: getCrefFromExp Returns a flattened list of the component references in an expression".
- Outline:** Provides a hierarchical view of the code structure, listing various algorithm items like `ADD`, `ALG_ASSIGN`, etc.
- Problems:** Shows a list of errors, with the first one being "The identifier at start and end are different".

Code Outline for easy navigation within Modelica files

Identifier Info on Hovering

64M of 254M | Ctrl Contrib (Bottom)

Eclipse Debugging Environment

The screenshot displays the Eclipse IDE with the following components:

- Breakpoints**: A panel showing no active breakpoints.
- Variables**: A table showing the state of variables during a debug session.
- Console**: A panel for output, currently empty.
- Outline**: A panel showing the project structure and a list of functions.
- Code Editor**: Shows the source code of 'Bla.mo' and 'Main.mo'.

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

```
model Bla
  Integer z[10];
end Bla;
```

```
local String s;
equation
  isModelicaFile(f);
  p = Parser.parse(f);
  Debug.fprint("dump", "\n----- Parsed progr
  Debug.fcall("dumpgraphviz", DumpGraphviz.dump, p);
  Debug.fcall("dump", Dump.dump, p);
```

Outline:

- readSettingsFile(String filePath, Interactive.InteractiveSy
- runBackendQ => Boolean
- runModparQ => Boolean
- serverLoop(Integer inInteger, Interactive.InteractiveSym
- serverLoopCorba(Interactive.InteractiveSymbolTable inIn
- simcodegen(Absyn.Path inPath1, SCode.Program inProgr
- transformFlatProgram(Absyn.Program p, String filename)
- translateFile(list<String> inStringLst)
- versionRequest
- import Absyn;
- import Ceval;
- import Corba;

- 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".

The interface is divided into several panes:

- Variables Browser:** Contains a search field, "Case Sensitive" checkbox, "Regular Expression" dropdown, "Expand All" and "Collapse All" buttons, and a table of variables.
- Defined In Equations:** A table with columns "Index", "Type", and "Equation".
- Used In Equations:** A table with columns "Index", "Type", and "Equation".
- Equations Browser:** A table with columns "Index", "Type", and "Equation".
- Source Browser:** Displays the source code for "C:/Users/adeas31/Desktop/Debugging.mo".
- Equation Operations:** A section showing operations for the selected equation.

Variables Browser Table:

Variables	Comment	Line	Location
-A	Storage ... section	120	C:\User
-Kv	Valve coefficient	112	C:\User
-T0	Tempera...g fluid	118	C:\User
-T1	Pump di...erature	138	C:\User
-Tref	Referen...utation	124	C:\User

Defined In Equations Table:

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

Used In Equations Table:

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

Equations Browser Table:

Index	Type	Equation
-1	initial	(assignment) ...*(T0 - Tref)
-2	initial	(assignment)...o * y + patm
-3	initial	(assignment..._pump ^ 2.0
-4	initial	(assignmen...ump + patm
-5	initial	(assignment)... Line: 144")
-6	initial	(assignment)...ve = p1 - p2
-7	initial	(residual,sqr...5 - dp_valve)
8	initial	(nonlinear)
3	initial	(assignment..._pump ^ 2.0
4	initial	(assignmen...ump + patm
5	initial	(assignment)... Line: 144")
6	initial	(assignment)...ve = p1 - p2
7	initial	(residual,sqr...5 - dp_valve)
9	initial	(assignment)..._4(String)#
10	initial	(assignment...a3

Equation Operations Table:

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

Source Browser Code:

```
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 2017!

- OpenModelica
 - What is OpenModelica?
 - The past
- OpenModelica Technical Overview
 - OMC, OMShell, OMNotebook,
 - OMEdit, ModelicaML
- OpenModelica Development Environment
 - MetaModelica
 - The Eclipse Environment
- OpenModelica Latest Developments (2016-2017)

Latest Developments (2016-2017) (I)

- 2016 - 2017 -focus on libraries support, performance and usability
 - MSL 3.2.2 (98% build/97% simulate), ModelicaTest 3.2.2, PetriNet, Buildings, PowerSystems, OpenHydraulics, ThermoPower, and ThermoSysPro, Modelica_Synchronous
- **Front-end, Back-end, Simulation Runtime, Graphical Clients**
 - Improved support for synchronous language features and state machines
 - Better support for some libraries in the front-end and back-end
 - Improved back-end: initialization, system solving, parallelization, cse optimization, dynamic optimization
 - Performance & scalability improvements, ScalableTestSuite 10 times faster
 - Faster and much more user friendly OpenModelica Connection Editor
 - 3D viewer
 - Transformational & Algorithmic Debugger
 - Compiler development support
 - Code folding, minimal difference on graphical editing to support source versioning
 - Support for replaceable will be available for testing later this week
 - Bug fixes to FMI support for Model Exchange and Co-Simulation
- **General**
 - Feb 2015 - Feb 2016
 - 60 contributors - up by 5 contributors
 - 1420 commits (OMCompiler)
 - Bug fixes
 - 64bit and 32bit versions available
 - Release 1.9.7 (Linux, Mac, Windows), Release 1.11.0 today (beta4 available)

Latest Developments (2016-2017) (II)

■ Improved scalability

- faster compilation speed & reduced memory requirements
- 2015 - OMC handles ~60000 equations in 700 seconds
- 2017 - OMC handles ~60000 equations in *45 seconds (~15x faster)*

■ New Front-End

- Work is progressing on the new front-end ~50% complete, more developers are working in parallel (see #4138 on Trac)
- 20 times faster or more, much more scalable both in time and memory (no array expansions, no expansion of for loops in equations)
- The new front-end will also bring better support for libraries also (the last 20% we are missing, mostly because of issues with package constants and complex redeclares)
- Developed in line with MCP-0019: Flattening

■ OMEdit

- Much more robust, doesn't crash that often anymore
- We get crash reports with stack traces when users experience a crash

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, 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

OpenModelica Project

<http://www.OpenModelica.org>

- OMC
 - The OMC API is quite inefficient and hugely non-homogeneous
- OMEdit
 - Still slow for displaying some models
 - Doesn't understand Modelica code directly, needs to talk with OMC
 - Is missing basic features for models using redeclare & replaceable