Model-Based Systems Engineering Center

Georgia Institute of Technology

Debugging Modelica Models

Chris Paredis

Objectives

- Learn how to prevent modeling problems in Modelica
  - Common mistakes
  - Tips and best practices
- Learn how to debug Modelica models
  - Dymola's tools
  - Common bugs
    - » Symptoms
    - » Tests
    - » Explanations and remedies
- Practice debugging

Common Mistakes: Syntax Errors

- How to detect?
  - Dymola error message when checking or translating (may be cryptic!)
- Look out for:
  - Proper use of keywords and operators
  - Case sensitivity
  - Semicolon at end of every line
  - Every variable must be declared

Syntax Error Example: What's Wrong Here?

```
1  model EllipseSweep "Draw an ellipse"
2   //equations from http://en.wikipedia.org/wiki/Ellipse
3   import SI = Modelica.SIunits;
4   constant Integer X = 1 "Index of x coordinate";
5   constant Integer Y = 2 "Index of y coordinate";
6   parameter SI.Length L[2] = {1, 2} "axis lengths {semimajor, semiminor}"
7   parameter SI.Position[2] c = {0,0} "center position {x, y}";
8   parameter SI.Angle Phi=Modelica.Constants.pi/4 "Rotation angle";
9   SI.Position p[2] "Position of point";
10  SI.Angle Theta(start = 0, fixed = true) "Sweep angle";
11 equation
12  der(Theta) = 2*pi;
13  p(x) = c(x) + L(x)*cos(Theta)*cos(Phi) - L(y)*sin(Theta)*sin(Phi);
14  p(y) = c(y) + L(y)*sin(Theta)*cos(Phi) + L(x)*cos(Theta)*sin(Phi);
15 end EllipseSweep;
```

- Dymola error message 1:

Error: Use of undeclared variable pi

In class DebuggingLectureHandout.SyntaxErrors.EllipseSweepAllErrors.

In file: C:/Users/cparedis/Documents/Dymola/DebuggingLectureHandout backup/package.mo

Errors found in: der(Theta) = 2*pi

Modelica Text: line 12

Example: What's Wrong Here?

```
1  model EllipseSweep "Draw an ellipse"
2   //equations from http://en.wikipedia.org/wiki/Ellipse
3   import SI = Modelica.SIunits;
4   constant Integer X = 1 "Index of x coordinate";
5   constant Integer Y = 2 "Index of y coordinate";
6   parameter SI.Length L[2] = {1, 2} "axis lengths {semimajor, semiminor}"
7   parameter SI.Position                                    y}";
8   parameter SI.Angle Ph                                  n angle";
9   SI.Position p[2] "Position of point";
10  SI.Angle Theta(start = 0, fixed = true) "Sweep angle";
11 equation
12  der(Theta) = 2*pi;
13  p(x) = c(x) + L(x)*cos(Theta)*cos(Phi) - L(y)*sin(Theta)*sin(Phi);
14  p(y) = c(y) + L(y)*sin(Theta)*cos(Phi) + L(x)*cos(Theta)*sin(Phi);
15 end EllipseSweep;
```

Provide explicit path to classes (or use import)

- Dymola error message 1:

Error: Use of undeclared variable pi

In class DebuggingLectureHandout.SyntaxErrors.EllipseSweepAllErrors.

In file: C:/Users/cparedis/Documents/Dymola/DebuggingLectureHandout backup/package.mo

Errors found in: der(Theta) = 2*pi

Modelica Text: line 12

Syntax Error Example: What's Wrong Here?

```
1  model EllipseSweep "Draw an ellipse"
2   //equations from http://en.wikipedia.org/wiki/Ellipse
3   import SI = Modelica.SIunits;
4   constant Integer X = 1 "Index of x coordinate";
5   constant Integer Y = 2 "Index of y coordinate";
6   parameter SI.Length L[2] = {1, 2} "axis lengths {semimajor, semiminor}";
7   parameter SI.Position[2] c = {0,0} "center position {x, y}";
8   parameter SI.Angle Phi=Modelica.Constants.pi/4 "Rotation angle";
9   SI.Position p[2] "Position of point";
10  SI.Angle Theta(start = 0, fixed = true) "Sweep angle";
11 equation
12  der(Theta) = 2*Modelica.Constants.pi;
13  p(x) = c(x) + L(x)*cos(Theta)*cos(Phi) - L(y)*sin(Theta)*sin(Phi);
14  p(y) = c(y) + L(y)*sin(Theta)*cos(Phi) + L(x)*cos(Theta)*sin(Phi);
15 end EllipseSweep;
```

- Dymola error message 1:

Error: subscripting with () instead of [] for variable c.

Warning: Found function c without explicit declaration in (model DebuggingLectureHandout.SyntaxErrors.EllipseSweepAllErrors).

Syntax Error Example: What's Wrong Here?

```
1  model EllipseSweep "Draw an ellipse"
2   //equations from http://en.wikipedia.org/wiki/Ellipse
3   import SI = Modelica.SIunits;
4   constant Integer X = 1 "Index of x coordinate";
5   constant Integer Y = 2 "Index of y coordinate";
6   parameter SI.Length L[2] = {1, 2} "axis lengths {semimajor, semiminor}";
7   parame                                    er position {x, y}";
8   parame                                    s.pi/4 "Rotation angle";
9   SI.Position p[2] "Position of point";
10  SI.Angle Theta(start = 0, fixed = true) "Sweep angle";
11 equation
12  der(Theta) = 2*Modelica.Constants.pi;
13  p(x) = c(x) + L(x)*cos(Theta)*cos(Phi) - L(y)*sin(Theta)*sin(Phi);
14  p(y) = c(y) + L(y)*sin(Theta)*cos(Phi) + L(x)*cos(Theta)*sin(Phi);
15 end EllipseSweep;
```

Use [ ] for array and vector subscripting, not ( )

- Dymola error message 1:

Error: subscripting with () instead of [] for variable c.

Warning: Found function c without explicit declaration in (model DebuggingLectureHandout.SyntaxErrors.EllipseSweepAllErrors).

Syntax Error Example: What's Wrong Here?

```
1  model EllipseSweep "Draw an ellipse"
2   //equations from http://en.wikipedia.org/wiki/Ellipse
3   import SI = Modelica.SIunits;
4   constant Integer X = 1 "Index of x coordinate";
5   constant Integer Y = 2 "Index of y coordinate";
6   parameter SI.Length L[2] = {1, 2} "axis lengths {semimajor, semiminor}";
7   parameter SI.Position[2] c = {0,0} "center position {x, y}";
8   parameter SI.Angle Phi=Modelica.Constants.pi/4 "Rotation angle";
9   SI.Position p[2] "Position of point";
10  SI.Angle Theta(start = 0, fixed = true) "Sweep angle";
11 equation
12  der(Theta) = 2*Modelica.Constants.pi;
13  p[x] = c[x] + L[x]*cos(Theta)*cos(Phi) – L[y]*sin(Theta)*sin(Phi);
14  p[y] = c[y] + L[y]*sin(Theta)*cos(Phi) + L[x]*cos(Theta)*sin(Phi);
15 end EllipseSweep;
```

- Dymola error message 1:

Error: Use of undeclared variable x

In class DebuggingLectureHandout.SyntaxErrors.EllipseSweepAllErrors.

In file: C:/Users/cparedis/Documents/Dymola/DebuggingLectureHandout backup/package.mo

Syntax Error Example: What's Wrong Here?

```
1  model EllipseSweep "Draw an ellipse"
2   //equations from http://en.wikipedia.org/wiki/Ellipse
3   import SI = Modelica.SIunits;
4   constant Integer X = 1 "Index of x coordinate";
5   constant Integer Y = 2 "Index of y coordinate";
6   parameter SI.Length L[2] = {1, 2} "axis lengths {semimajor, semiminor}";
7   p                                 "center position {x, y}";
8   parameter SI.Angle Phi=Modelica.constants.pi/4 "Rotation angle";
9   SI.Position p[2] "Position of point";
10  SI.Angle Theta(start = 0, fixed = true) "Sweep angle";
11 equation
12  der(Theta) = 2*Modelica.Constants.pi;
13  p[x] = c[x] + L[x]*cos(Theta)*cos(Phi) - L[y]*sin(Theta)*sin(Phi);
14  p[y] = c[y] + L[y]*sin(Theta)*cos(Phi) + L[x]*cos(Theta)*sin(Phi);
15 end EllipseSweep;
```

Modelica is case sensitive!

- Dymola error message 1:

Error: Use of undeclared variable x

In class DebuggingLectureHandout.SyntaxErrors.EllipseSweepAllErrors.

In file: C:/Users/cparedis/Documents/Dymola/DebuggingLectureHandout backup/package.mo

Syntax Error Example: What's Wrong Here?

```
1  model EllipseSweep "Draw an ellipse"
2   //equations from http://en.wikipedia.org/wiki/Ellipse
3   import SI = Modelica.SIunits;
4   constant Integer x = 1 "Index of x coordinate";
5   constant Integer y = 2 "Index of y coordinate";
6   parameter SI.Length L[2] = {1, 2} "axis lengths {semimajor, semiminor}"
7   parameter SI.Position[2] c = {0,0} "center position {x, y}";
8   parameter SI.Angle Phi=Modelica.Constants.pi/4 "Rotation angle";
9   SI.Position p[2] "Position of point";
10  SI.Angle Theta(start = 0, fixed = true) "Sweep angle";
11 equation
12  der(Theta) = 2*Modelica.Constants.pi;
13  p[x] = c[x] + L[x]*cos(Theta)*cos(Phi) – L[y]*sin(Theta)*sin(Phi);
14  p[y] = c[y] + L[y]*sin(Theta)*cos(Phi) + L[x]*cos(Theta)*sin(Phi);
15 end EllipseSweep;
```

- Dymola error message 1:

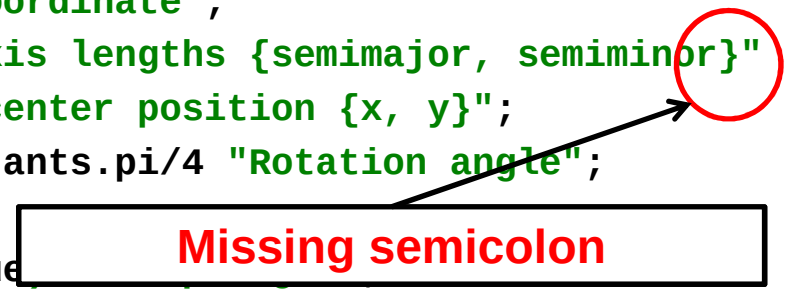
Error: declaration window line 7 column 3, syntax error at "parameter" missing { EndOfFile EXTEN DS CONSTRAINEDBY ANNOTATION) ; , }

Error: declaration window line 7 column 35, syntax error at "0" missing { OPERATOR IDENT }

ERROR: 15 errors were found

Syntax Error Example: What's Wrong Here?

```
1  model EllipseSweep "Draw an ellipse"
2   //equations from http://en.wikipedia.org/wiki/Ellipse
3   import SI = Modelica.SIunits;
4   constant Integer x = 1 "Index of x coordinate";
5   constant Integer y = 2 "Index of y coordinate";
6   parameter SI.Length L[2] = {1, 2} "axis lengths {semimajor, semiminor}"
7   parameter SI.Position[2] c = {0,0} "center position {x, y}";
8   parameter SI.Angle Phi=Modelica.Constants.pi/4 "Rotation angle";
9   SI.Position p[2] "Position of point";
10  SI.Angle Theta(start = 0, fixed = true
11 equation
12  der(Theta) = 2*Modelica.Constants.pi;
13  p[x] = c[x] + L[x]*cos(Theta)*cos(Phi) - L[y]*sin(Theta)*sin(Phi);
14  p[y] = c[y] + L[y]*sin(Theta)*cos(Phi) + L[x]*cos(Theta)*sin(Phi);
15 end EllipseSweep;
```



- Dymola error message 1:

Error: declaration window line 7 column 3, syntax error at "parameter" missing { EndOfFile EXTEN DS CONSTRAINEDBY ANNOTATION) ; , }

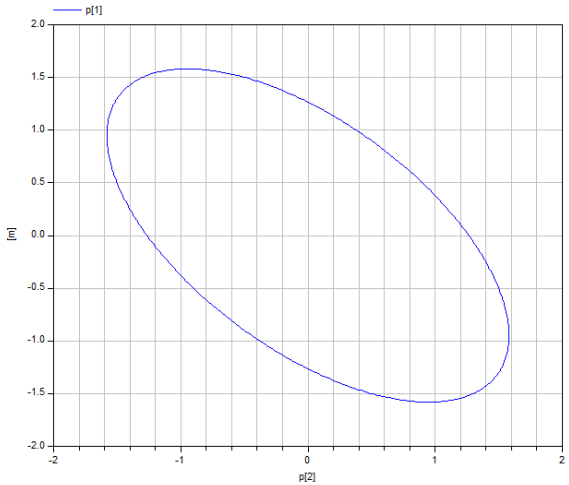
Error: declaration window line 7 column 35, syntax error at "0" missing { OPERATOR IDENT }

ERROR: 15 errors were found

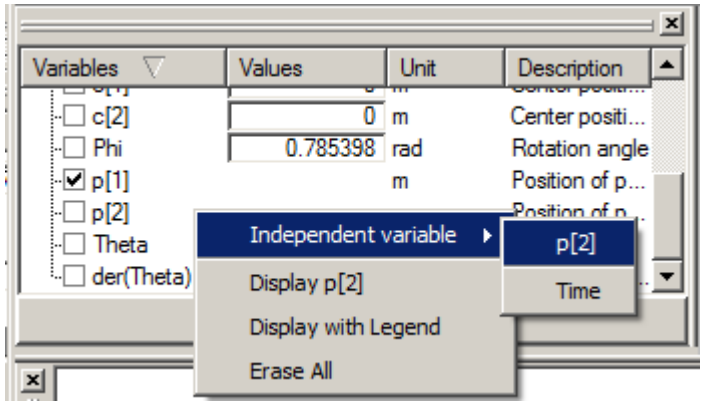
Syntax Error Example: Finally Correct!

```
1  model EllipseSweep "Draw an ellipse"
2   //equations from http://en.wikipedia.org/wiki/Ellipse
3   import SI = Modelica.SIunits;
4   constant Integer x = 1 "Index of x coordinate";
5   constant Integer y = 2 "Index of y coordinate";
6   parameter SI.Length L[2] = {1, 2} "axis lengths {semimajor, semiminor}";
7   parameter SI.Position[2] c = {0,0} "center position {x, y}";
8   parameter SI.Angle Phi=Modelica.Constants.pi/4 "Rotation angle";
9   SI.Position p[2] "Position of point";
10  SI.Angle Theta(start = 0, fixed = true) "Sweep angle";
11 equation
12  der(Theta) = 2*Modelica.Constants.pi;
13  p[x] = c[x] + L[x]*cos(Theta)*cos(Phi) – L[y]*sin(Theta)*sin(Phi);
14  p[y] = c[y] + L[y]*sin(Theta)*cos(Phi) + L[x]*cos(Theta)*sin(Phi);
15 end EllipseSweep;
```

Syntax Error Example: Finally Correct!



- Quick Tip: In simulation results browser, right click on variable and choose “Independent variable”

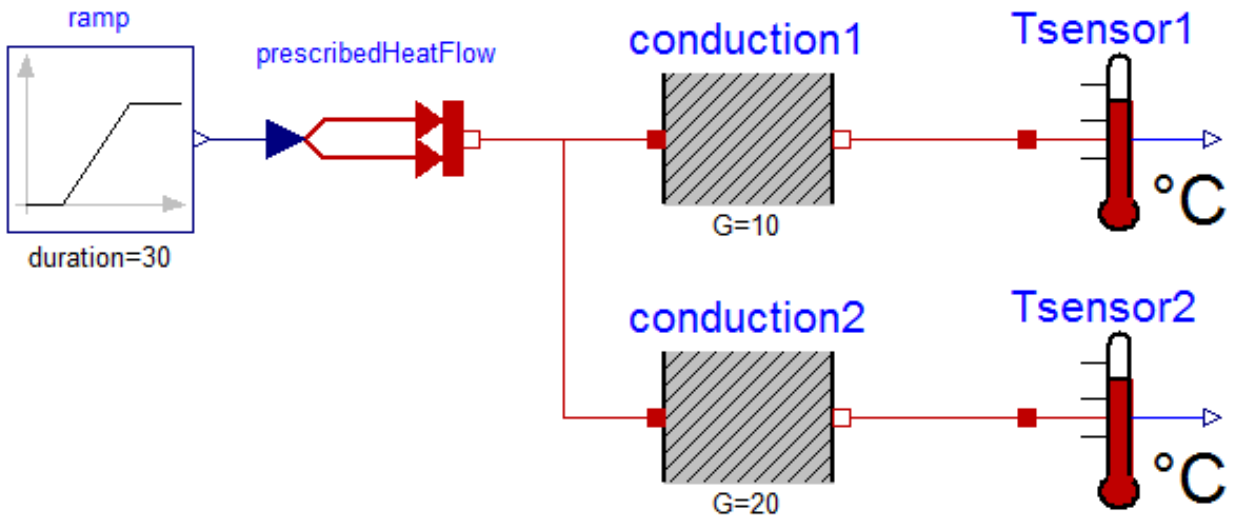


Common Mistakes: Under / Over Specified Systems

- How to detect?
  - When checking or translating, Dymola reports error
  - Long listing of variables/equations is an indication of where the problem may exist – Read it carefully!!
- When using models from the Modelica libraries
  - Did you connect components correctly?
  - Did you include a reference (e.g., electrical ground)?
- When using your own models
  - Do each of the component models have the appropriate number of equations?
  - Do the connectors have an equal number of through and across variables? (If not, problems may occur when the model contains loops)

Example: What's Wrong Here?

DebuggingLectureHandout.MysteryError1.ThermalNetwork



- Dymola error message (simulation tab):

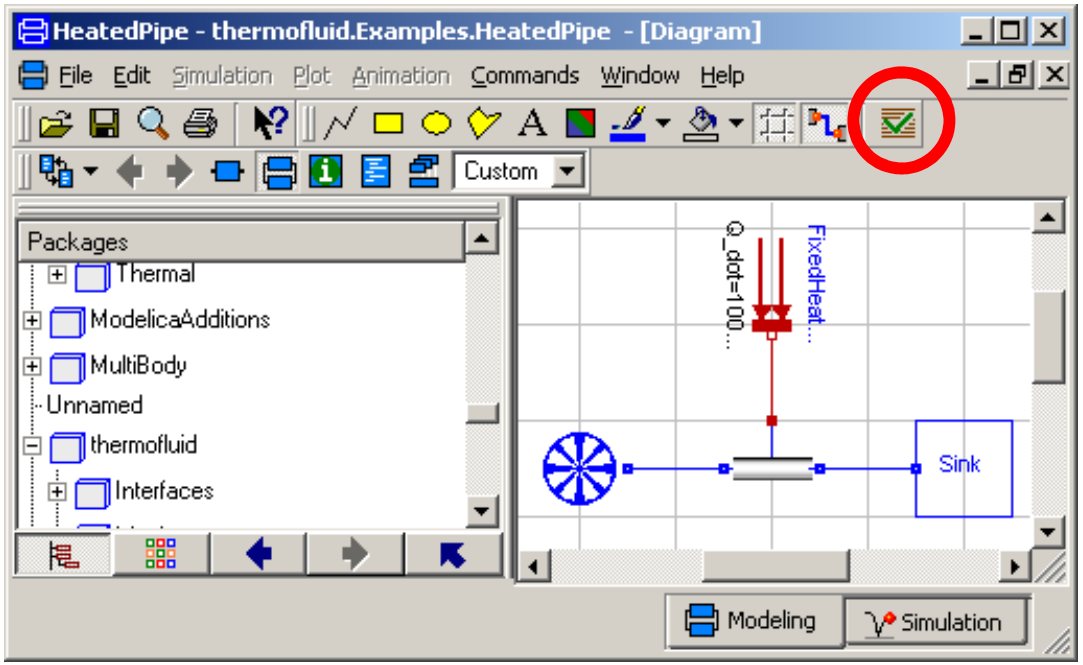
The following error was detected at time: 0
Error: Singular inconsistent scalar system for conduction1.port_a.T = (prescribedHeatFlow.Q_flow*(1-prescribedHeatFlow.alpha*prescribedHeatFlow.T_ref))/(-prescribedHeatFlow.Q_flow*prescribedHeatFlow.alpha) = 300/-0

Tips and Best Practices

- Review Modelica Reference for usage and syntax of keywords and operators
- Duplicate and modify existing models from Modelica Standard Library or elsewhere (don’t start from scratch unless necessary)
- Pay attention to the translation warnings (not just the errors)

Tips and Best Practices: Checking in Modeling Mode

- Check models while in modeling mode
  - Quick check for syntax:
    - » Ctrl + L
  - Quick check for syntax and squareness:



Tips and Best Practices: Stopping the Simulation

Stop the simulation

- When experiment is complete

```
when vehicle.body.s >= 100 then
   terminate("Race over");
end when;
```

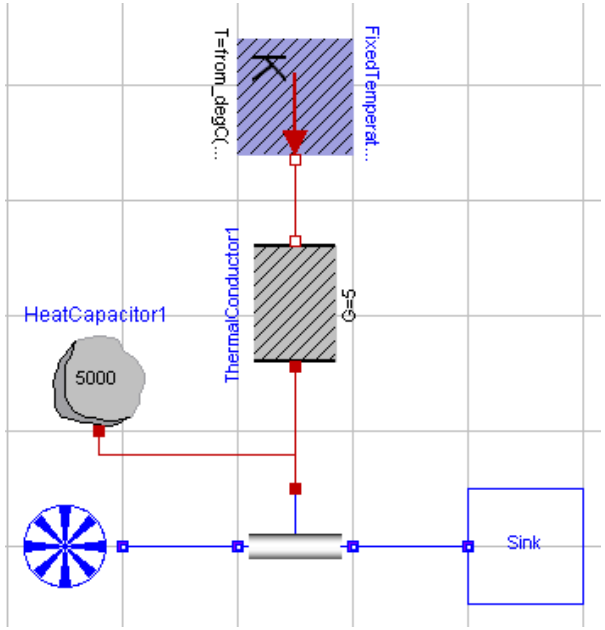
- When variable goes outside its valid range

```
assert(medium.T >= 273.15 and medium.T <=
  423.15, "Temperature outside valid range of
  medium");
```

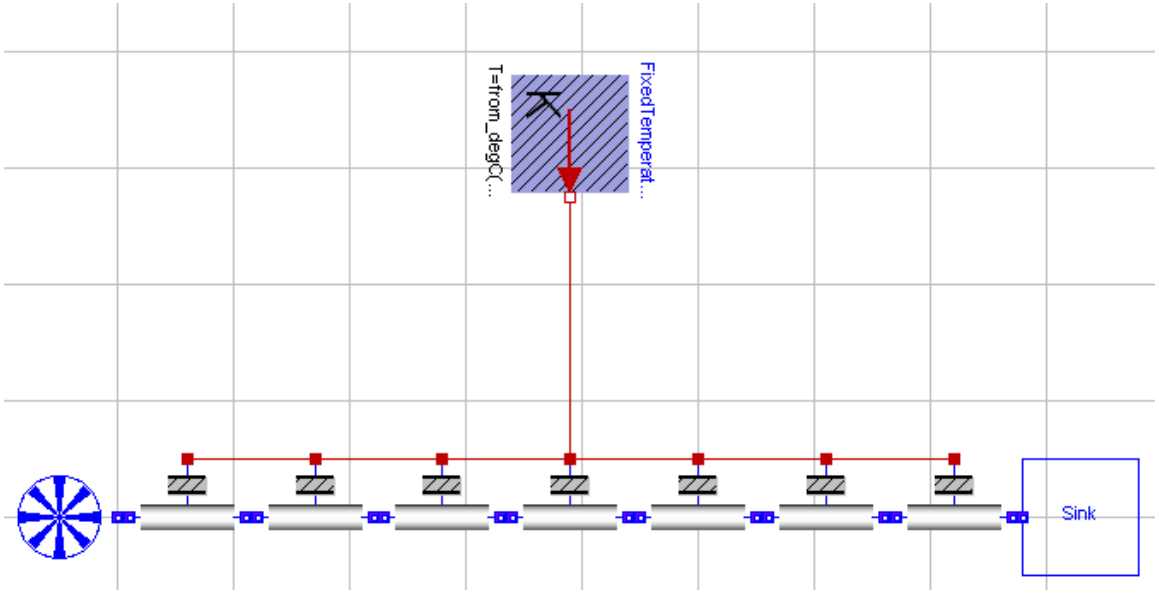
- Note: By default in Dymola, the solver doesn't stop when a variable's min or max attribute is exceeded.

Tips and Best Practices: Test and Retest

- Test your models at all levels



Test in a simple model first…



Before moving on to a more complex model

- Suggestion: Create an “Examples” package

Summary: Preventing Modeling Errors

- Common Mistakes
  - Syntax errors
  - Misconnected components
  - Over / under specified models
  - Over / under specified initial conditions

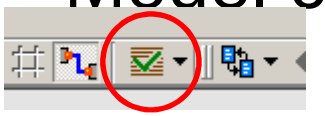
- Tips and Best Practices
  - Avoid starting models from scratch
  - Check while you are in modeling mode
  - Pay attention to warnings
  - Stop the simulation based on an event
  - Test your models at all levels

Objectives

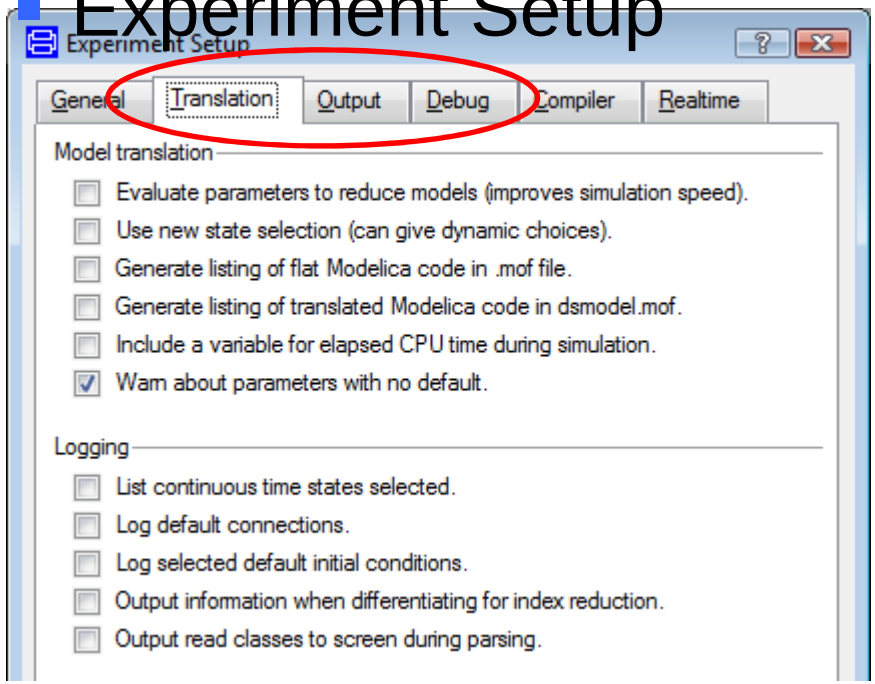
- Learn how to prevent modeling problems in Modelica
  - Common mistakes
  - Tips and best practices
- Learn how to debug Modelica models
  - Dymola's tools
  - Common bugs
    - » Symptoms
    - » Tests
    - » Remedies
- Practice debugging

Dymola's Debugging Tools

- Model check



- Experiment Setup



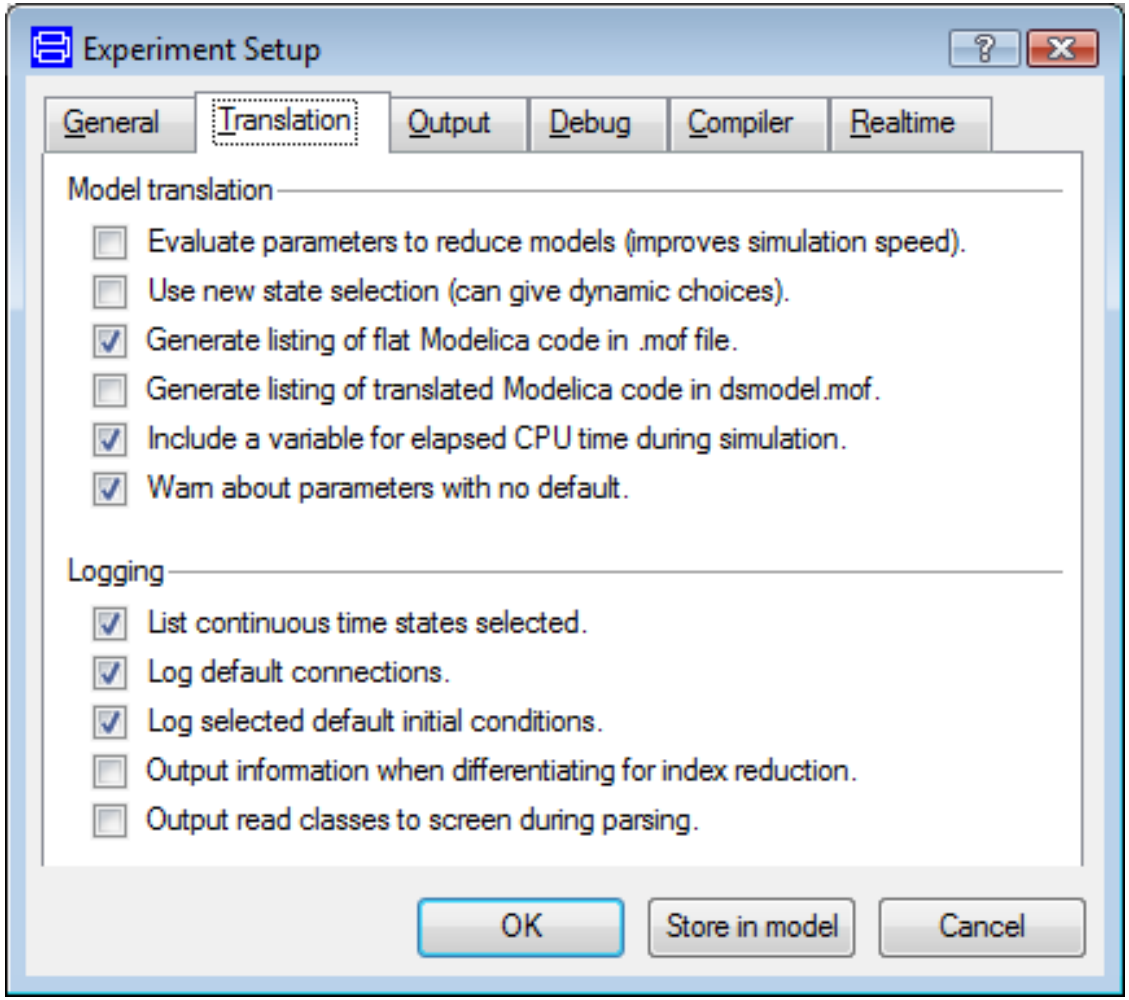
–Translation

–Output

–Debug

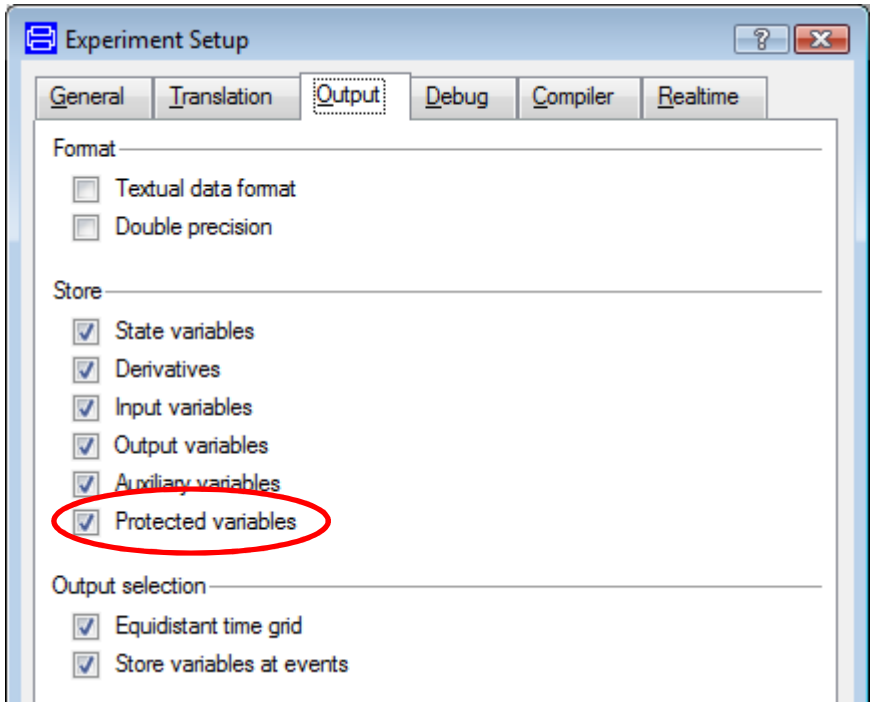
Experiment Setup → Translation

- Useful options:
  - Generate listing of translated Modelica code in dsmodel.mof.
  - Include a variable for elapsed CPU time during simulation.
  - List continuous time states selected.
  - Log default connections.
  - Log selected default initial conditions.



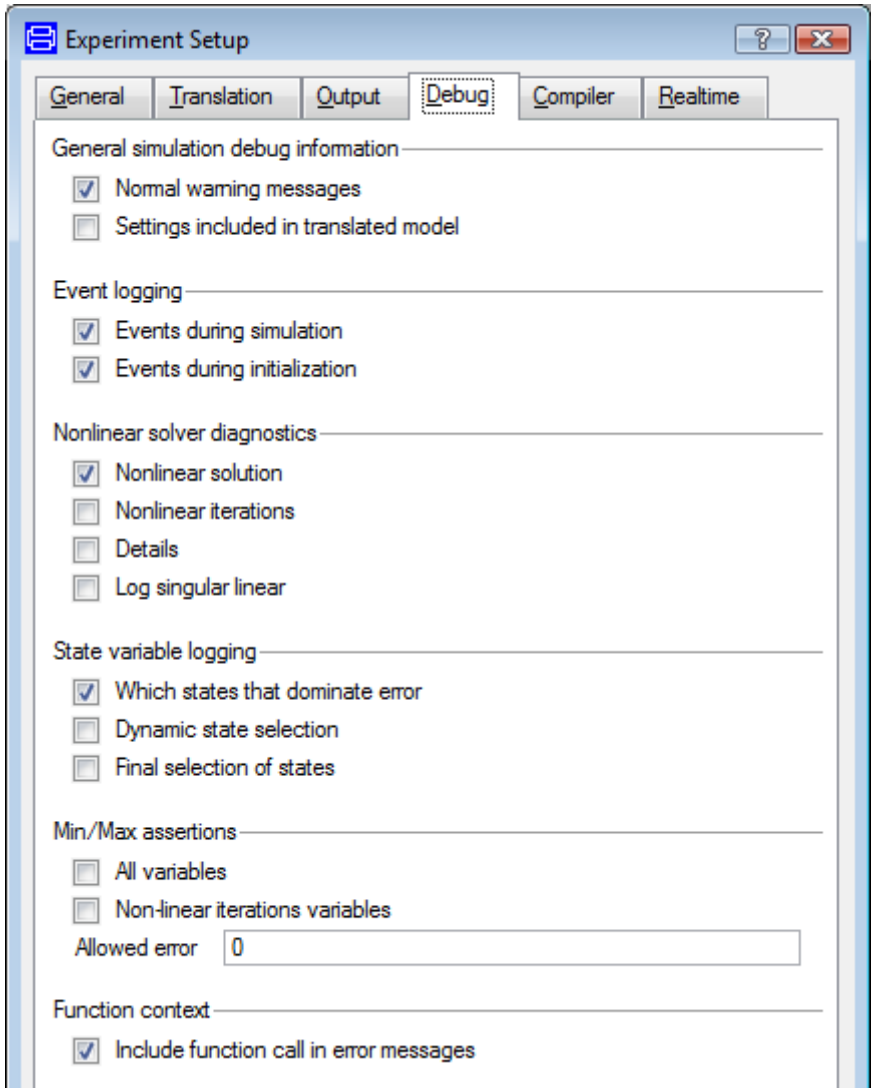
Experiment Setup → Output

- Useful options:
  - Store protected variables



Experiment Setup → Debug

- Useful options:
  - Events during simulation
  - Events during initialization
  - Nonlinear solution
  - Which states that dominate error
- Other options (with caution):
  - Nonlinear iterations & details
  - Min/Max assertions – All variables

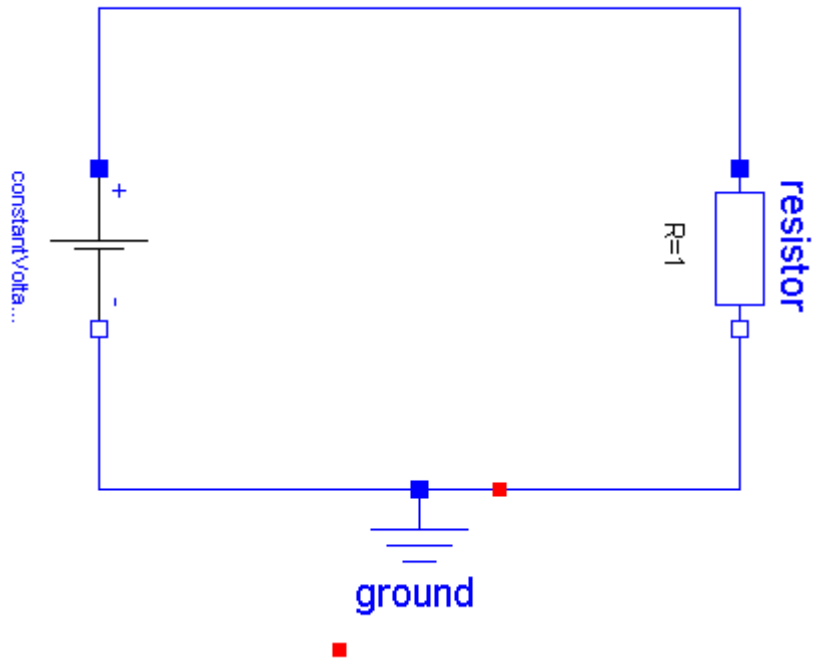


Objectives

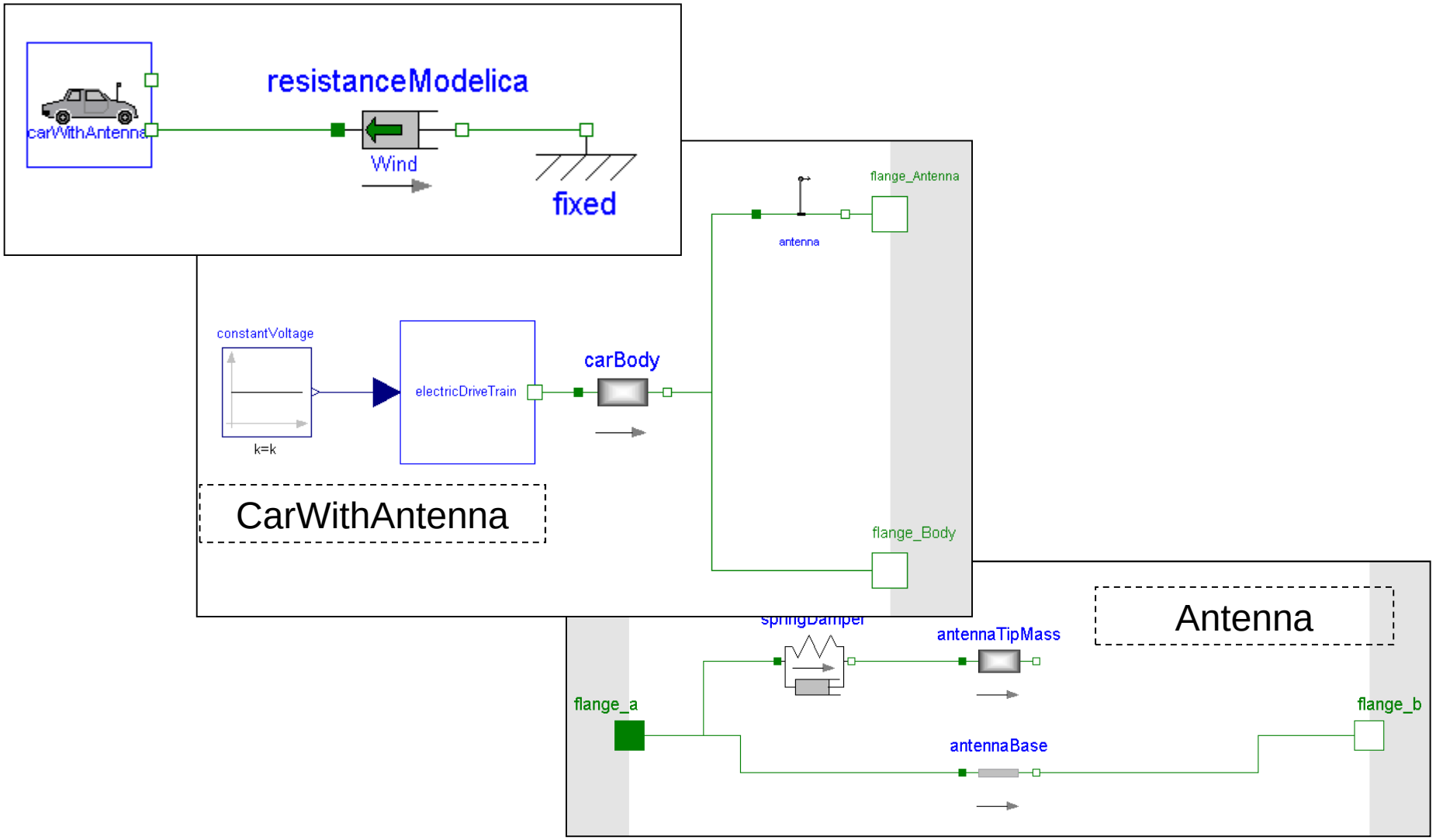
- Learn how to prevent modeling problems in Modelica
  - Common mistakes
  - Tips and best practices
- Learn how to debug Modelica models
  - Dymola's tools
  - Common bugs
    - » Symptoms
    - » Tests
    - » Remedies
- Practice debugging

Exercise 1: Why Won't the Model Translate?

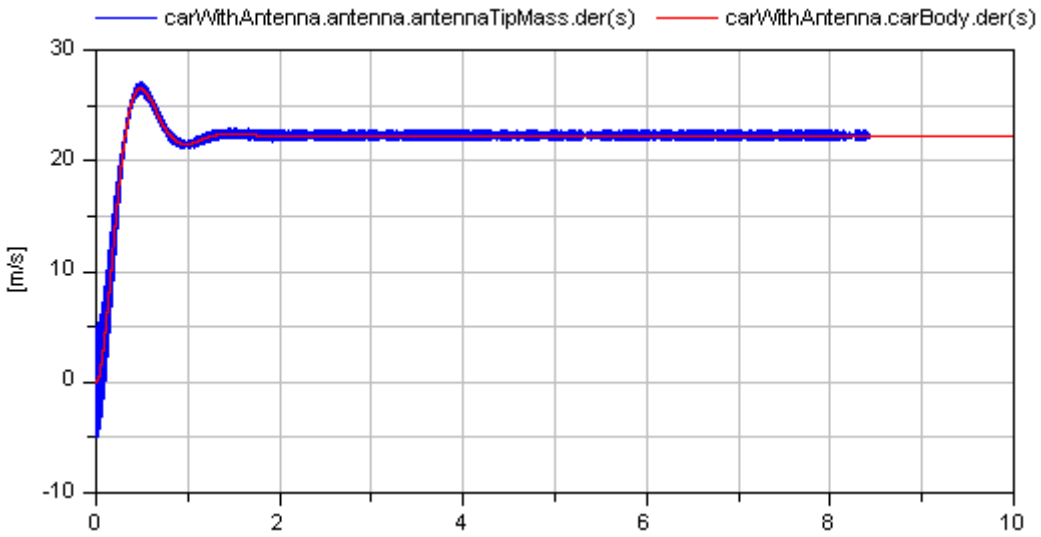
DebuggingLectureHandout.MysteryError2.BadCircuit

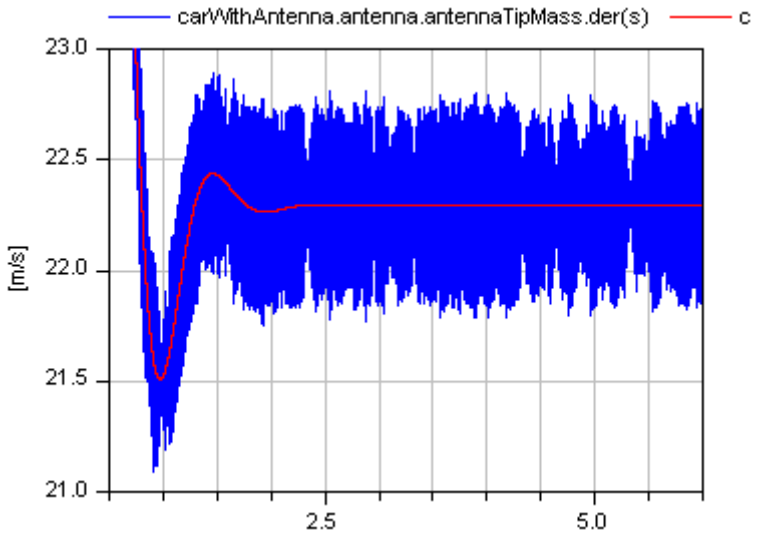


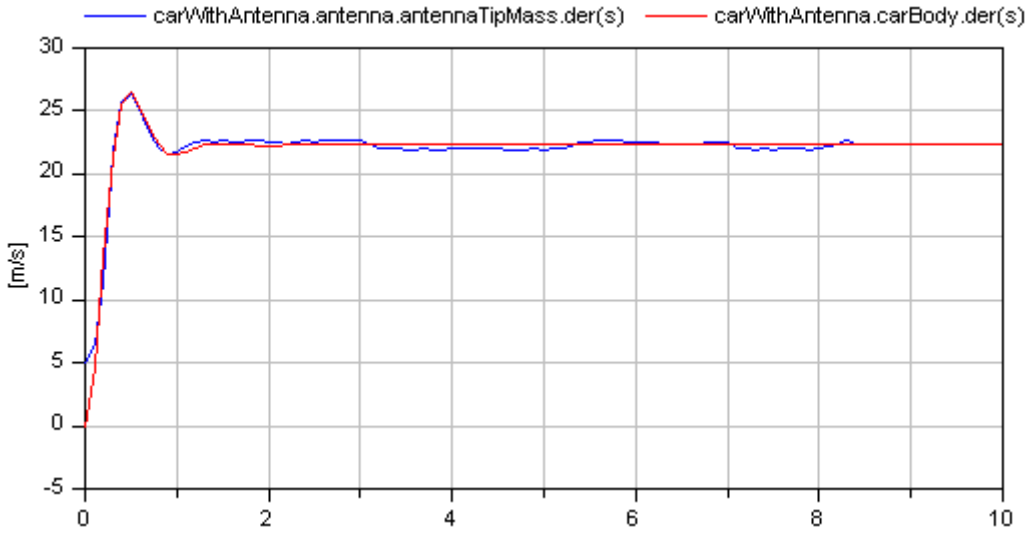
Demo: Stiffness

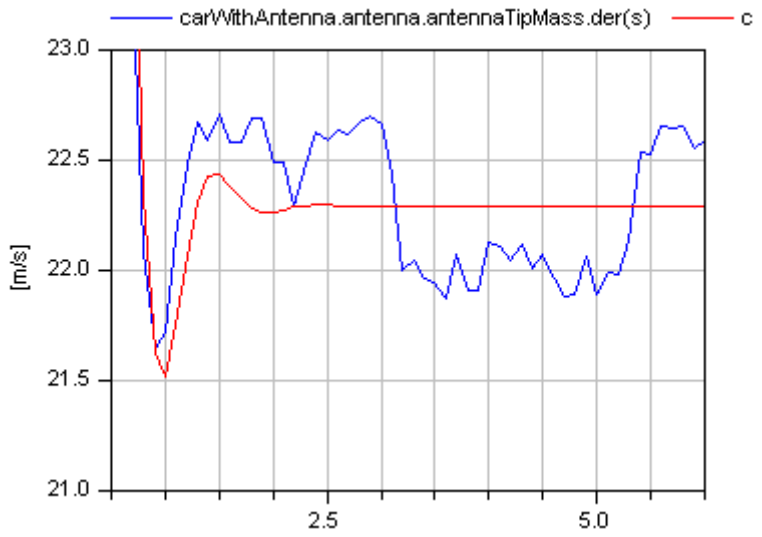


Demo: Stiffness

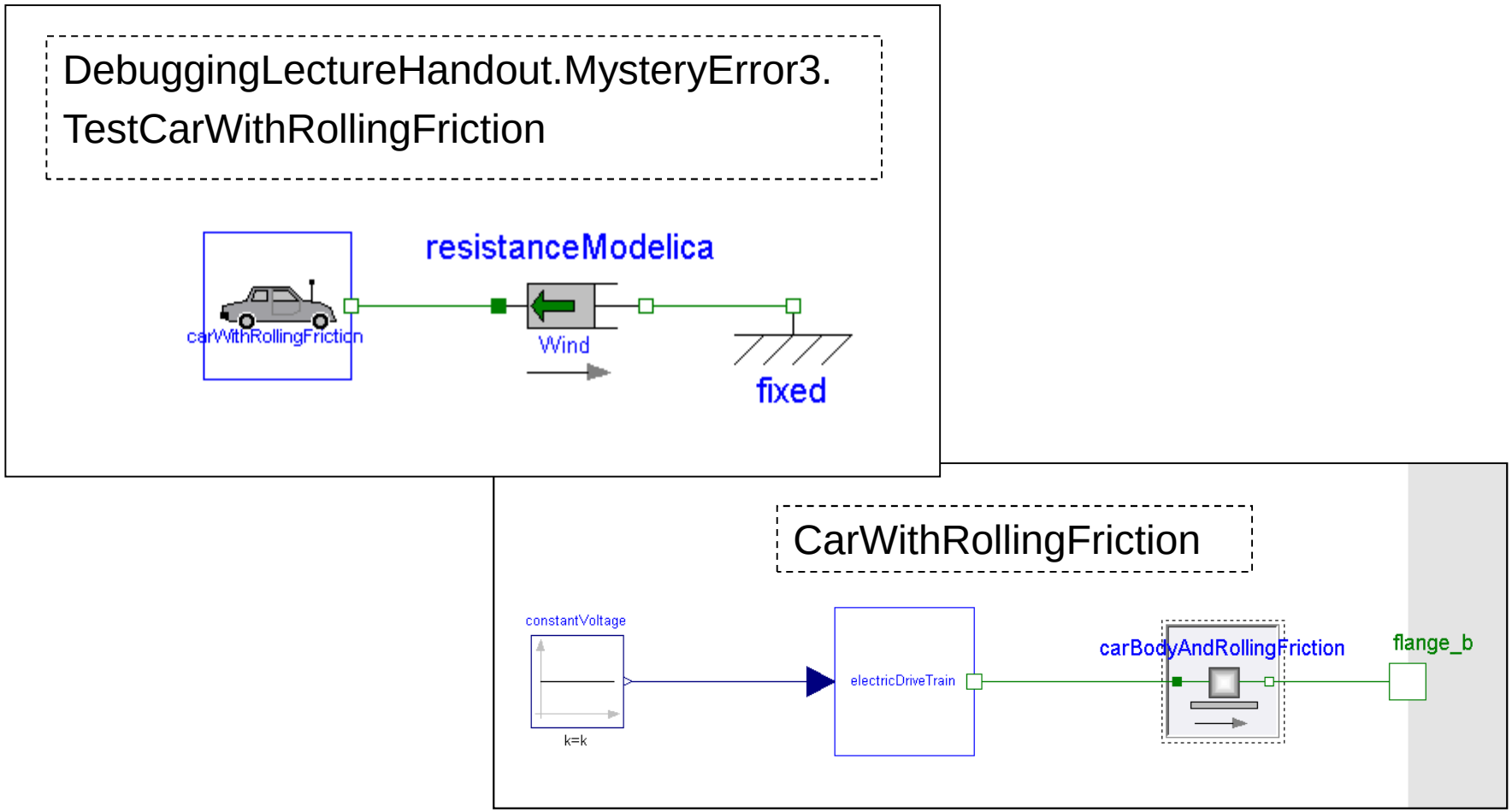








Exercise 2: Why Is the Simulation So Slow?



Scenario 1: Model Fails to Compile

- Symptoms:
  - Errors in Syntax or Translation tab of Messages window
- Possible explanations / remedies:
  - Syntax errors
  - Over / underspecified equations
- Tests:
  - Run model check
  - Build model up from its components, testing at each level

Scenario 2: Solver Fails at t = 0

- Symptoms:
  - Error message: “Integration terminated before reaching "StopTime" at T = 0”
- Possible explanations:
  - Solver may be having difficulty finding a consistent set of initial conditions
- Tests:
  - Experiment Setup → Translation: Log selected default initial conditions
  - Experiment Setup → Debug: Min / max assertions – All variables
- Possible remedies:
  - Add / adjust initial conditions to give reasonable values
    - » (start = […])
    - » (start = […], fixed = true)
    - » initial equation
  - Set limits
    - » (min = […], max = […])
    - » Use the types in Modelica.SIunits

Scenario 3: Solver Fails at t > 0

- Symptoms:
  - Error message: “Integration terminated before reaching "StopTime" at T = 0”
- Possible explanations:
  - Ill-posed mathematical operation (e.g., divide by zero or logarithm of zero)
  - Simulation may be outside the model’s valid range
- Tests:
  - Tools under Experiment Setup → Translation
  - Tools under Experiment Setup → Debug
- Possible remedies:
  - Avoid physical assumptions which may be implicit in your model
  - Add assert() or terminate() statements

Scenario 4: Slow Simulation

- Symptoms:
  - May have this error: “WARNING: You have many state events. It might be due to chattering.”
  - May have this error: “Probably the communication interval is too large or the system is stiff.”
- Possible explanations:
  - Many state events (i.e., chattering)
  - Problem is stiff or model approaches singularity
  - Difficulty solving algebraic equations (large sets of nonlinear equations)
  - Dynamic state selection
- Tests:
  - Tools under Experiment Setup → Debug
  - In solver window (DOS-window):
    - » Press Ctrl-C to get a solver update
    - » Press Ctrl-C twice in a row and you go into the debugging mode:
      - c: continue
      - q: quit
      - l: log
- Possible remedies:
  - Reformulate the model equations
  - Reduce the model to suit the time range of interest
  - Static state selection (e.g. "Real x(stateSelect=StateSelect.prefer)")

Further Reading

- Kevin Davies and Chris Paredis, “Dymola Frequently Asked Questions,” http://www.srl.gatech.edu/education/ME6105/Homework/DymolaFAQ
- Dynasim AB, *Dymola – Dynamic Modeling Laboratory: User’s Manual,* version 5.3a, 2004
  - Especially the “Debugging Models” section, p. 182-192
  - Find the manual by clicking on Help→Documentation from the Dymola program
- Peter Fritzson, *Principles of Object-Oriented Modeling and Simulation with Modelica 2.1,* Wiley-IEEE Computer Society Press, 2003. (ISBN: 047147163)
  - Especially Section 18.3.2, “Deugging Equation-Based Models,” p. 693-695