

SBML Test Suite

Lucian Smith
libSBML team; Caltech

What it is

- A way to test simulators to ensure they produce the simulation results they're supposed to.
- A collection of models, rules, and expected results.

What it isn't

- A way to test SBML features that don't affect simulation results.
- A great way to test translators (though you can hack it).

Interaction:

http://sbml.org/Software/SBML_Test_Suite

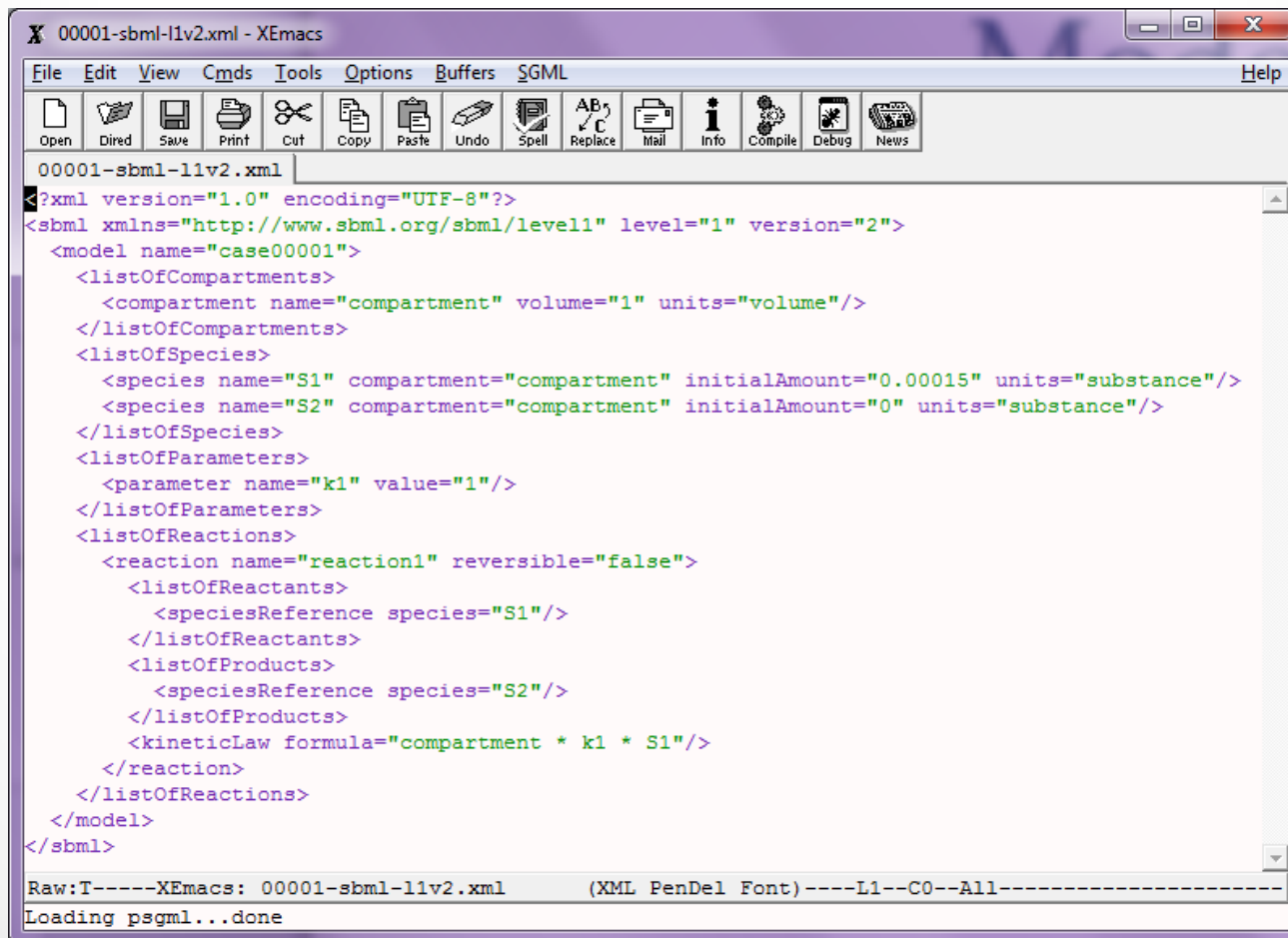
- Use the Online Test Suite at
http://sbml.org/Facilities/Online_SBML_Test_Suite
- Download the tests from
<https://sourceforge.net/projects/sbml/files/test-suite/>
- Get the SVN from
<https://sbml.svn.sourceforge.net/svnroot/sbml/trunk/test-suite>
- Use Frank's Comparison Tool
<https://compare-results.sf.net/>

Structure

- `semantic/00001/` through `01123/`
- Each test contains:
 - Models for appropriate levels(SBML L1v2 \rightarrow L3v1)
 - A description (`-model.html`)
 - A settings file (`-settings.txt`)
 - A SED-ML version of the settings file (`-sedml.xml`)
 - A plot of the results (`-plot.jpg`)
 - (in SVN): The numeric results (`-results.csv`)

Models

- Range from the simple...



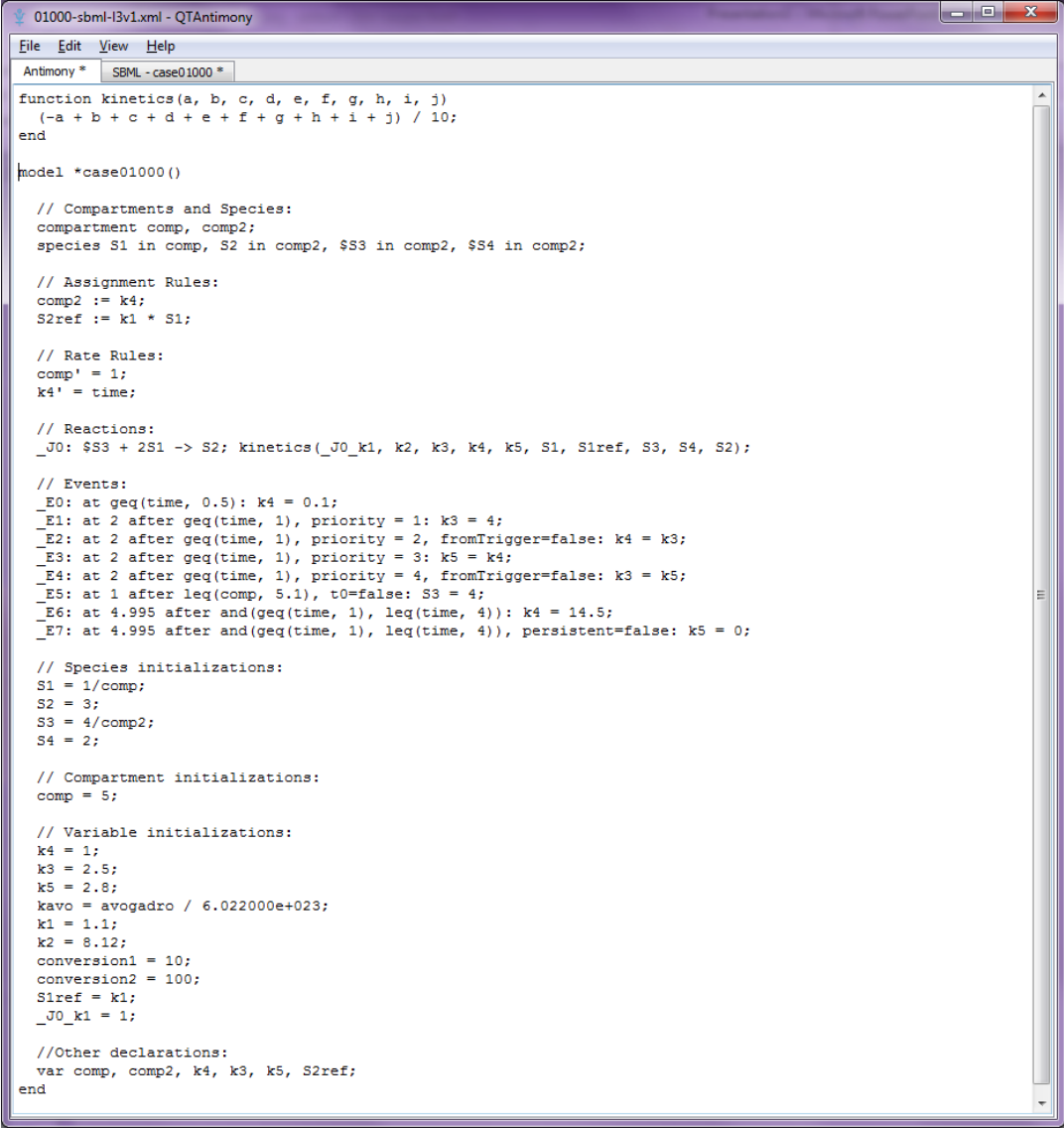
The screenshot shows an XEmacs editor window titled "00001-sbml-l1v2.xml - XEmacs". The menu bar includes File, Edit, View, Cmds, Tools, Options, Buffers, SGML, and Help. The toolbar contains icons for Open, Dired, Save, Print, Cut, Copy, Paste, Undo, Spell, Replace, Mail, Info, Compile, Debug, and News. The main text area displays the following XML code:

```
00001-sbml-l1v2.xml
<?xml version="1.0" encoding="UTF-8"?>
<sbml xmlns="http://www.sbml.org/sbml/level1" level="1" version="2">
  <model name="case00001">
    <listOfCompartments>
      <compartment name="compartment" volume="1" units="volume"/>
    </listOfCompartments>
    <listOfSpecies>
      <species name="S1" compartment="compartment" initialAmount="0.00015" units="substance"/>
      <species name="S2" compartment="compartment" initialAmount="0" units="substance"/>
    </listOfSpecies>
    <listOfParameters>
      <parameter name="k1" value="1"/>
    </listOfParameters>
    <listOfReactions>
      <reaction name="reaction1" reversible="false">
        <listOfReactants>
          <speciesReference species="S1"/>
        </listOfReactants>
        <listOfProducts>
          <speciesReference species="S2"/>
        </listOfProducts>
        <kineticLaw formula="compartment * k1 * S1"/>
      </reaction>
    </listOfReactions>
  </model>
</sbml>
```

The status bar at the bottom shows "Raw: T-----XEmacs: 00001-sbml-l1v2.xml (XML PenDel Font)-----L1--C0--A11-----" and "Loading psgml...done".

Models

- ...to the complex



```
01000-sbml-l3v1.xml - QTAntimony
File Edit View Help
Antimony * SBML - case01000 *

function kinetics(a, b, c, d, e, f, g, h, i, j)
  (-a + b + c + d + e + f + g + h + i + j) / 10;
end

model *case01000()

  // Compartments and Species:
  compartment comp, comp2;
  species S1 in comp, S2 in comp2, S3 in comp2, S4 in comp2;

  // Assignment Rules:
  comp2 := k4;
  S2ref := k1 * S1;

  // Rate Rules:
  comp' = 1;
  k4' = time;

  // Reactions:
  _J0: S3 + 2S1 -> S2; kinetics(_J0_k1, k2, k3, k4, k5, S1, S1ref, S3, S4, S2);

  // Events:
  _E0: at geq(time, 0.5): k4 = 0.1;
  _E1: at 2 after geq(time, 1), priority = 1: k3 = 4;
  _E2: at 2 after geq(time, 1), priority = 2, fromTrigger=false: k4 = k3;
  _E3: at 2 after geq(time, 1), priority = 3: k5 = k4;
  _E4: at 2 after geq(time, 1), priority = 4, fromTrigger=false: k3 = k5;
  _E5: at 1 after leq(comp, 5.1), t0=false: S3 = 4;
  _E6: at 4.995 after and(geq(time, 1), leq(time, 4)): k4 = 14.5;
  _E7: at 4.995 after and(geq(time, 1), leq(time, 4)), persistent=false: k5 = 0;

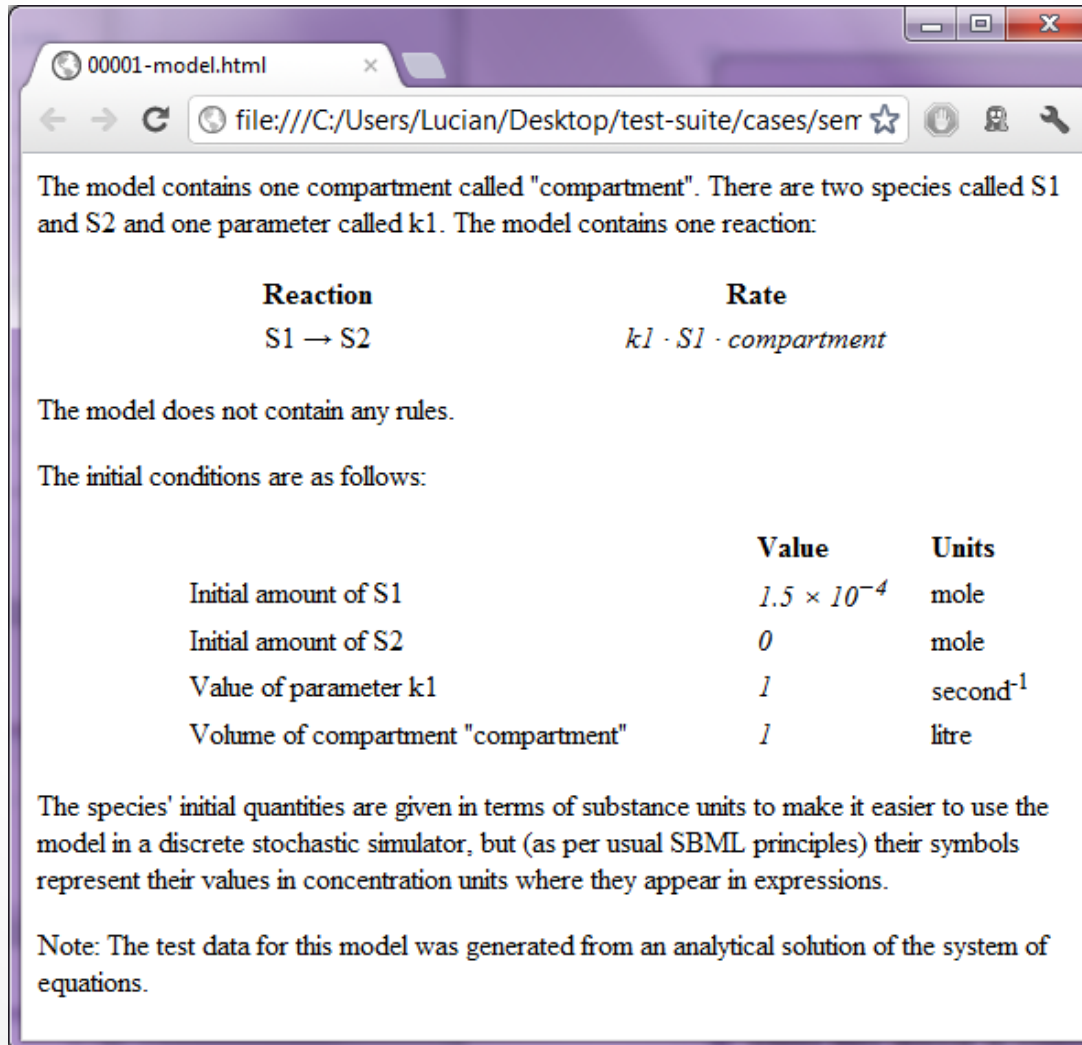
  // Species initializations:
  S1 = 1/comp;
  S2 = 3;
  S3 = 4/comp2;
  S4 = 2;

  // Compartment initializations:
  comp = 5;

  // Variable initializations:
  k4 = 1;
  k3 = 2.5;
  k5 = 2.8;
  kavo = avogadro / 6.022000e+023;
  k1 = 1.1;
  k2 = 8.12;
  conversion1 = 10;
  conversion2 = 100;
  S1ref = k1;
  _J0_k1 = 1;

  //Other declarations:
  var comp, comp2, k4, k3, k5, S2ref;
end
```

Model description file



The screenshot shows a web browser window with the address bar displaying "file:///C:/Users/Lucian/Desktop/test-suite/cases/serr". The page content describes a model with one compartment, two species (S1, S2), one parameter (k1), and one reaction. The reaction is S1 → S2 with a rate expression $k1 \cdot S1 \cdot compartment$. The model does not contain any rules. The initial conditions are listed in a table below.

The model contains one compartment called "compartment". There are two species called S1 and S2 and one parameter called k1. The model contains one reaction:

Reaction	Rate
$S1 \rightarrow S2$	$k1 \cdot S1 \cdot compartment$

The model does not contain any rules.

The initial conditions are as follows:

	Value	Units
Initial amount of S1	1.5×10^{-4}	mole
Initial amount of S2	0	mole
Value of parameter k1	1	second ⁻¹
Volume of compartment "compartment"	1	litre

The species' initial quantities are given in terms of substance units to make it easier to use the model in a discrete stochastic simulator, but (as per usual SBML principles) their symbols represent their values in concentration units where they appear in expressions.

Note: The test data for this model was generated from an analytical solution of the system of equations.

Settings file

start: 0

duration: 5

steps: 50

variables: S1, S2

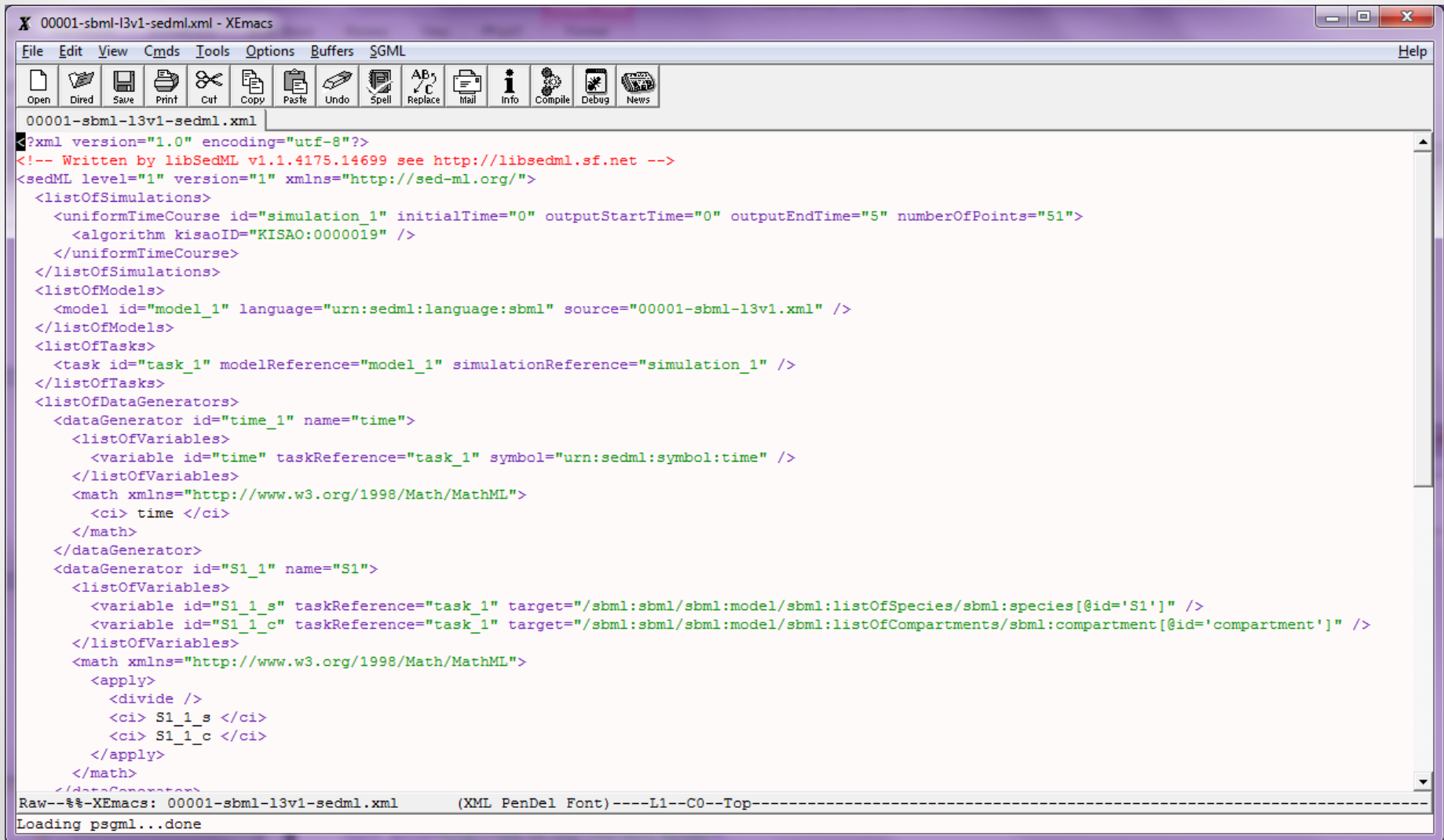
absolute: 1.000000e-007

relative: 0.0001

amount: S1, S2

concentration:

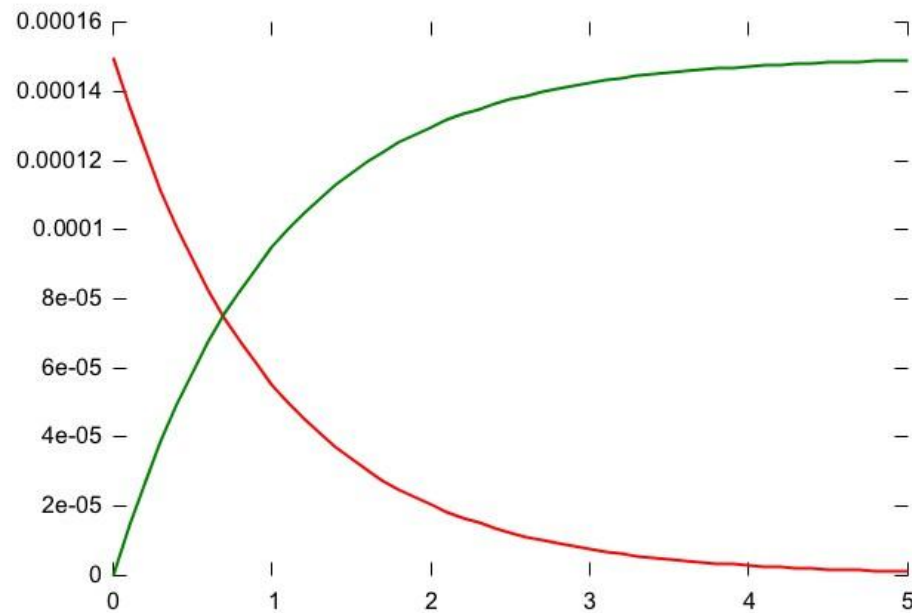
The SED-ML file



The screenshot shows an XEmacs editor window titled "00001-sbml-l3v1-sedml.xml - XEmacs". The menu bar includes File, Edit, View, Cmds, Tools, Options, Buffers, SGML, and Help. The toolbar contains icons for Open, Dired, Save, Print, Cut, Copy, Paste, Undo, Spell, Replace, Mail, Info, Compile, Debug, and News. The main text area displays the XML content of the SED-ML file, which is color-coded. The XML starts with a declaration of version "1.0" and encoding "utf-8". It includes a comment about the version and a link to the libSedML project. The root element is <sedML>, which contains a list of simulations, a list of models, a list of tasks, a list of data generators, and a list of variables. The data generators include a time generator and a species generator. The species generator defines two variables, S1_1_s and S1_1_c, which are used in a mathematical expression. The status bar at the bottom shows "Raw--XEmacs: 00001-sbml-l3v1-sedml.xml (XML PenDel Font)----L1--C0--Top-----" and "Loading psgml...done".

```
00001-sbml-l3v1-sedml.xml
<?xml version="1.0" encoding="utf-8"?>
<!-- Written by libSedML v1.1.4175.14699 see http://libsedml.sf.net -->
<sedML level="1" version="1" xmlns="http://sed-ml.org">
  <listOfSimulations>
    <uniformTimeCourse id="simulation_1" initialTime="0" outputStartTime="0" outputEndTime="5" numberOfPoints="51">
      <algorithm kisaoID="KISAO:0000019" />
    </uniformTimeCourse>
  </listOfSimulations>
  <listOfModels>
    <model id="model_1" language="urn:sedml:language:sbml" source="00001-sbml-l3v1.xml" />
  </listOfModels>
  <listOfTasks>
    <task id="task_1" modelReference="model_1" simulationReference="simulation_1" />
  </listOfTasks>
  <listOfDataGenerators>
    <dataGenerator id="time_1" name="time">
      <listOfVariables>
        <variable id="time" taskReference="task_1" symbol="urn:sedml:symbol:time" />
      </listOfVariables>
      <math xmlns="http://www.w3.org/1998/Math/MathML">
        <ci> time </ci>
      </math>
    </dataGenerator>
    <dataGenerator id="S1_1" name="S1">
      <listOfVariables>
        <variable id="S1_1_s" taskReference="task_1" target="/sbml:sbml/sbml:model/sbml:listOfSpecies/sbml:species[@id='S1']" />
        <variable id="S1_1_c" taskReference="task_1" target="/sbml:sbml/sbml:model/sbml:listOfCompartments/sbml:compartment[@id='compartment']" />
      </listOfVariables>
      <math xmlns="http://www.w3.org/1998/Math/MathML">
        <apply>
          <divide />
          <ci> S1_1_s </ci>
          <ci> S1_1_c </ci>
        </apply>
      </math>
    </dataGenerator>
  </listOfDataGenerators>
  <listOfVariables>
    <variable id="S1_1_s" taskReference="task_1" target="/sbml:sbml/sbml:model/sbml:listOfSpecies/sbml:species[@id='S1']" />
    <variable id="S1_1_c" taskReference="task_1" target="/sbml:sbml/sbml:model/sbml:listOfCompartments/sbml:compartment[@id='compartment']" />
  </listOfVariables>
  <math xmlns="http://www.w3.org/1998/Math/MathML">
    <apply>
      <divide />
      <ci> S1_1_s </ci>
      <ci> S1_1_c </ci>
    </apply>
  </math>
</sedML>
Raw--XEmacs: 00001-sbml-l3v1-sedml.xml (XML PenDel Font)----L1--C0--Top-----
Loading psgml...done
```

The results plot



Legend
S1 — S2 —

Results data (SVN only)

time,S1,S2

0,0.00015,0

0.1,0.0001357256127053939,1.427438729460607e-005

0.2,0.0001228096129616973,2.719038703830272e-005

0.3,0.0001111227331022577,3.887726689774233e-005

0.4,0.0001005480069053459,4.945199309465411e-005

0.5,9.097959895689501e-005,5.902040104310499e-005

0.6,8.232174541410396e-005,6.767825458589604e-005

0.7,7.448779556871142e-005,7.551220443128858e-005

0.8,6.739934461758323e-005,8.260065538241677e-005

0.9,6.098544896108986e-005,8.901455103891014e-005

1,5.518191617571635e-005,9.481808382428365e-005

1.1,4.993066255471193e-005,0.0001000693374452881

Model description file (SVN only)

category:	Test
synopsis:	Basic single forward reaction with two species in one compartment
componentTags:	Compartment, Species, Reaction, Parameter
testTags:	Amount
testType:	TimeCourse
levels:	1.2, 2.1, 2.2, 2.3, 2.4, 3.1
generatedBy:	Analytic

The model contains one compartment called "compartment". There are two species called S1 and S2 and one parameter called k1. The model contains one reaction:

Filtering tests

Online SBML Test Suite

sbml.org/test-suite/web/selecttests.jsp

Parent pages: [SBML.org](#) / [Facilities](#) / [Online SBML Test Suite](#)

Step 1: Select SBML Tests

The first step in using the online Test Suite is to obtain a set of test models using the form on this page. Once you have answered the questions below and downloaded a set of tests, you must then record the results of simulating each model in your application and put the results in files according to the guidelines in [step 2](#).

You can also skip this page and download the entire 980 cases (version of 2011-11-15) as a [zip archive](#).

(a) Select the SBML Level and Version to use: SBML Level 3 Version 1 Core

By default, you will be provided with all test cases *unless* you specifically *exclude* some. To exclude cases (perhaps because you already know the software you are testing doesn't support certain features), use the following checkboxes to select the SBML components or types of tests that you want *excluded*. Leave all boxes unchecked to get all possible tests.

Dependencies exist: when certain components are excluded, some tests must be excluded too, and remain so while the relevant components are selected for exclusion. In addition, not all components are available in all SBML Levels/Versions.

(b) Select the SBML component tags you would like to exclude:

<input type="checkbox"/> Assignment rules	<input type="checkbox"/> Events without delays	<input type="checkbox"/> <csymbol> for 'avogadro'
<input type="checkbox"/> Rate rules	<input type="checkbox"/> Events with delays	<input type="checkbox"/> Compartments
<input type="checkbox"/> Algebraic rules	<input type="checkbox"/> Priorities for event triggers	<input type="checkbox"/> Species
<input type="checkbox"/> Function definition	<input type="checkbox"/> <csymbol> for 'delay'	<input type="checkbox"/> Reactions
<input type="checkbox"/> Initial assignment	<input type="checkbox"/> <csymbol> for 'time'	<input type="checkbox"/> Parameters

(c) Select the test tags you would like to exclude: (Hover over items for short explanations)

<input type="checkbox"/> Initial entity values overridden	<input type="checkbox"/> Species using initial amounts	<input type="checkbox"/> Simultaneously scheduled events
<input type="checkbox"/> Varying parameter values	<input type="checkbox"/> Species using initial concentration	<input type="checkbox"/> Event uses trigger-time values
<input type="checkbox"/> Local parameters in reactions	<input type="checkbox"/> Species with 'hasOnlySubstanceUnits'	<input type="checkbox"/> Event uses assignment-time values
<input type="checkbox"/> Conversion factors	<input type="checkbox"/> Species as boundary conditions	<input type="checkbox"/> Persistent events
<input type="checkbox"/> 2-D compartments	<input type="checkbox"/> Species declared as constant	<input type="checkbox"/> Non-persistent events
<input type="checkbox"/> 1-D compartments	<input type="checkbox"/> 'Fast' reactions	<input type="checkbox"/> Event triggers at time = 0
<input type="checkbox"/> 0-D compartments	<input type="checkbox"/> Reversible reactions	
<input type="checkbox"/> Varying-size compartments	<input type="checkbox"/> Stoichiometries $\neq 1$	
<input type="checkbox"/> Compartments with size $\neq 1$	<input type="checkbox"/> Math using stoichiometries	
<input type="checkbox"/> Multiple compartments	<input type="checkbox"/> Math using constant stoichiometries	

When you are finished excluding tests, click the [Get test cases](#) button to download a zip archive of the test cases left:

[Get test cases](#)[Reset form](#)

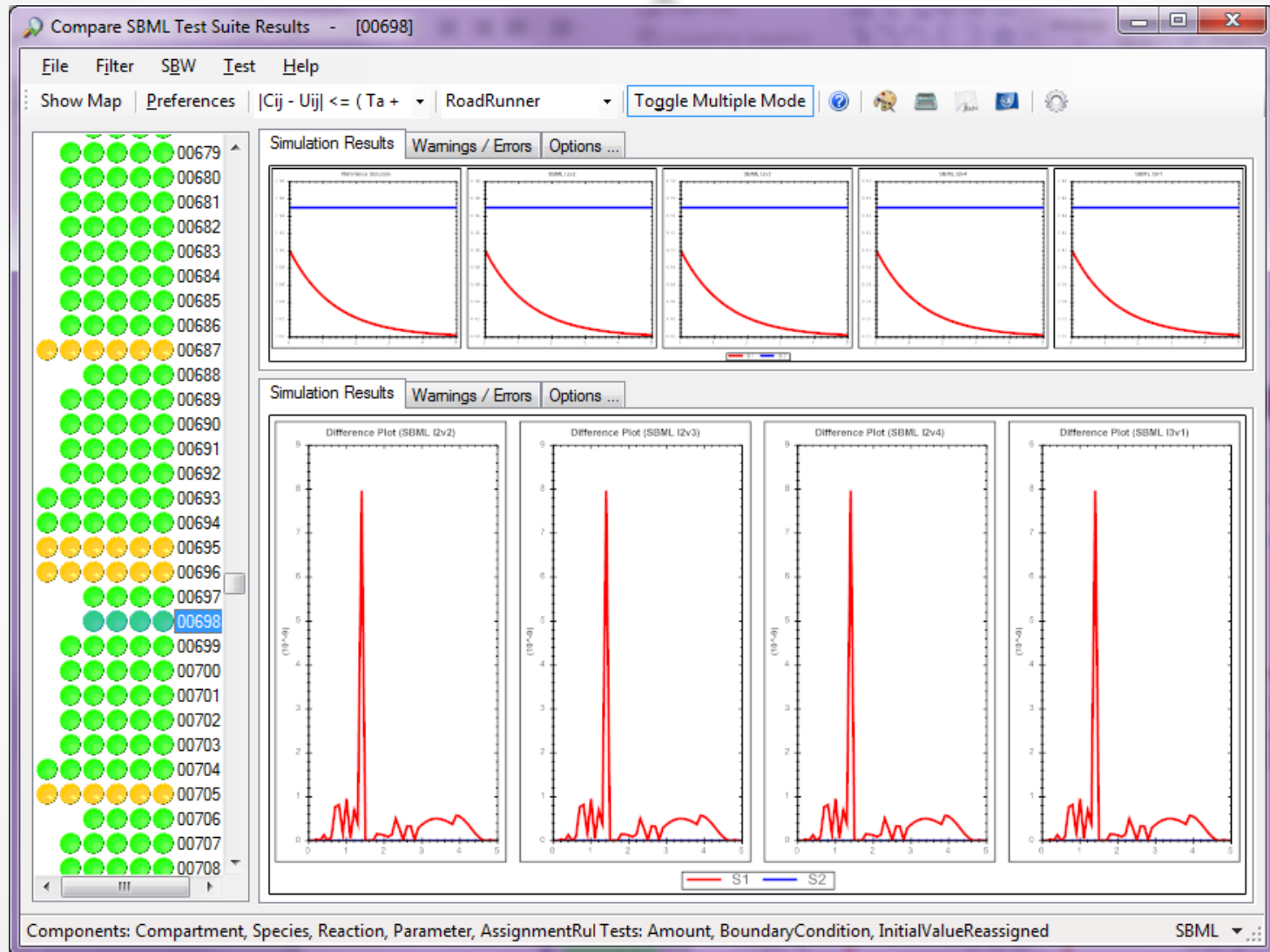
Component tags

- Basic:
 - AlgebraicRule
 - AssignmentRule
 - Compartment
 - FunctionDefinition
 - InitialAssignment
 - Parameter
 - RateRule
 - Reaction
 - Species
 - StoichiometryMath
- CSymbols:
 - CSymbolAvogadro
 - CSymbolDelay
 - CSymbolTime
- Events:
 - EventNoDelay
 - EventWithDelay
 - EventPriority

Test Tags

- Output
 - Amount
 - Concentration
- Reactions
 - ReversibleReaction
 - LocalParameters
 - FastReaction
 - NonUnityStoichiometry
 - AssignedConstantStoichiometry
 - AssignedVariableStoichiometry
 - SpeciesReferenceInMath
- Compartments
 - 0D-Compartment
 - MultiCompartment
 - NonUnityCompartment
 - NonConstantCompartment
- Species
 - BoundaryCondition
 - ConstantSpecies
 - HasOnlySubstanceUnits
- Events
 - EventsPersistent
 - EventsNotPersistent
 - EventT0Firing
 - EventUsesAssignmentTimeValues
 - EventUsesTriggerTimeValues
 - RandomEventExecution
- Changing math
 - ConversionFactors
 - InitialValueReassigned
 - NonConstantParameter

Frank's Comparison Tool



<https://compare-results.sf.net/>

Translators?

- Test round-tripping:
 - Obtain robust simulator that passes all tests.
 - Use your translator to round-trip models
 - Run simulator on round-tripped models
 - Compare results
 - Species→Parameters may be problematic.

What to do with results?

- Use the results to improve your software
- Tell us about it!

http://sbml.org/SBML_Software_Guide/

Questions?

Developing Tests

- Need a 'how can I break this?' mentality.
- Need to involve community
- 'Kitchen sink' vs. pinpointed tests

Future Plans/Ideas

- Put results file in other distributions.
- A more automated way for us to test others' software.
- A way to officially approve simulators on sbml.org
 - Political issues here!
- Analyze models for users to tell them what tags a simulator would need.