SBML Team facilities & software

Frank T. Bergmann
Sarah Keating
On behalf of the SBML Team
ONLINE VALIDATOR
http://sbml.org/Facilities/Validator
Options for Direct or Scheduled Validation

**Direct Validation**
- Intended for (smallish) models, or inexpensive validation runs (i.e., not for unit validation on models with thousands of reactions)
- 60 second timeout
- Yields results directly

**Scheduled Validation**
- Intended for all cases where validation is expected to take longer
- After scheduling you will receive a validation identifier that you can use later to retrieve the results
- Results will be kept for 24 hours
Customizable Error List

You can customize the way your results are displayed:

- Traditional
- Grouped By Category
- Grouped By Severity
- Grouped By Error

You can also choose to display / hide the SBML output:

- Display on Error
- Display Always
- Always Hide

Or choose how detailed the error messages should be (for non-traditional display only):

- Detailed
- Short messages

Apply
## Error Categories

### Units consistency

Category of errors that can occur while validating the units of measurement on quantities in a model.

<table>
<thead>
<tr>
<th>ErrorId</th>
<th>Meaning</th>
<th>L1V1</th>
<th>L1V2</th>
<th>L2V1</th>
<th>L2V2</th>
<th>L2V3</th>
<th>L2V4</th>
<th>L3V1</th>
</tr>
</thead>
<tbody>
<tr>
<td>10501</td>
<td>Units of arguments to function call do not match function's definition</td>
<td>Warning</td>
<td>Warning</td>
<td>Warning</td>
<td>Error</td>
<td>Error</td>
<td>Warning</td>
<td>Warning</td>
</tr>
<tr>
<td>10503</td>
<td>Inconsistent <code>&lt;kineticLaw&gt;</code> units</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>Warning</td>
</tr>
<tr>
<td>10511</td>
<td>Mismatched units in assignment rule for compartment</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Warning</td>
<td>Warning</td>
</tr>
<tr>
<td>10512</td>
<td>Mismatched units in assignment rule for species</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Warning</td>
<td>Warning</td>
</tr>
<tr>
<td>10513</td>
<td>Mismatched units in assignment rule for parameter</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Warning</td>
<td>Warning</td>
</tr>
<tr>
<td>10514</td>
<td>Mismatched units in assignment rule for stoichiometry</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>Warning</td>
</tr>
<tr>
<td>10521</td>
<td>Mismatched units in initial assignment to compartment</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>Error</td>
<td>Error</td>
<td>Warning</td>
<td>Warning</td>
</tr>
<tr>
<td>10522</td>
<td>Mismatched units in initial assignment to species</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>Error</td>
<td>Error</td>
<td>Warning</td>
<td>Warning</td>
</tr>
<tr>
<td>10523</td>
<td>Mismatched units in initial assignment to parameter</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>Error</td>
<td>Error</td>
<td>Warning</td>
<td>Warning</td>
</tr>
<tr>
<td>10524</td>
<td>Mismatched units in initial assignment to stoichiometry</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>Warning</td>
</tr>
<tr>
<td>10531</td>
<td>Mismatched units in rate rule for compartment</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Warning</td>
<td>Warning</td>
</tr>
<tr>
<td>10532</td>
<td>Mismatched units in rate rule for species</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Warning</td>
<td>Warning</td>
</tr>
<tr>
<td>10533</td>
<td>Mismatched units in rate rule for parameter</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Warning</td>
<td>Warning</td>
</tr>
<tr>
<td>10534</td>
<td>Mismatched units in rate rule for</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>Warning</td>
</tr>
</tbody>
</table>
REST API

• Invoke the validator by POST ing a file to it, or passing in a URL (with an API Key)

```
curl -F file=@filename.xml
   -F output=text
   -F offcheck=u,p,s
http://sbml.org/validator/
```

REST API

• Invoke the validator by POST ing a file to it, or passing in a URL (with an API Key)

curl -F url=<URL>
  -F apikey=<KEY>
  -F output=text
  -F offcheck=u,p,s
http://sbml.org/validator/

http://sbml.org/Facilities/Documentation/Validator_Web_API
REST API

- Invoke the validator by POSTing a file to it, or:

```bash
$ curl -F url=http://sbml.org/validator/api/sample-02.xml -
File: sample-02.xml
```

Options:
- Units consistency checking: off
- Identifier consistency checking: on
- MathML consistency checking: on
- SBO consistency checking: on
- Overdetermined model checking: on
- Modeling practices checking: on
- Overall SBML consistency checking: on

Results: The web request could not be authenticated.

http://sbml.org/validator/

http://sbml.org/Facilities/Documentation/Validator_Web_API
REST API

User information

Name: Frank Bergmann
Email Address: fbergman@caltech.edu
API Key: 1f234c22-7162-43e3-8715-3a10686dc33c
Software (optional): SBML Validator

Log out Change Password

Return to the SBML Validator front page.

http://sbml.org/Facilities/Documentation/Validator_Web_API
Web Service

String convertSBML(String sbmlModel, int targetLevel, int targetVersion)
String convertSBMLtoHTML(String content)
String expandFunctionDefinitions(String sbmlModel)
String expandInitialAssignments(String sbmlModel)
String getLibSBMLVersion()
String validateSBML(String sbmlModel, boolean withUnits, boolean withIdentifiers, boolean withMathML, boolean withSBO, boolean withOverdetermined, boolean withModelingPractices, boolean withGeneral)
String validateSBMLBuffer(byte[] sbmlContent, String type, boolean withUnits, boolean withIdentifiers, boolean withMathML, boolean withSBO, boolean withOverdetermined, boolean withModelingPractices, boolean withGeneral)
String validateSBMLBufferWithoutUnitChecks(byte[] sbmlContent, String type)
String validateSBMLWithoutUnitChecks(String sbmlModel)

http://sbml-validator.caltech.edu:8888/validator_ws/services/LibSBML?wsdl
SOFTWARE INDEX
SBML Software Tools Survey

http://sbml.org/SBML_Software_Guide/SBML_Software_Details_Entry_Questionnaire
Updated: Software Matrix

---

**SBML Software Matrix**

This matrix provides an at-a-glance summary of software known to us to provide some degree of support for reading, writing, or otherwise working with SBML. The columns' meanings are explained below. For a list of longer descriptions grouped into themes, please see our SBML Software Summary page. Please use the survey form to notify us about omissions, updates and suggestions.

<table>
<thead>
<tr>
<th>Capabilities</th>
<th>Frameworks</th>
<th>API</th>
<th>Dep.</th>
<th>Platforms</th>
<th>SBML</th>
<th>Availabil.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **ABC-SysBio**
- **acslXtreme**
- **ALC**
- **Antimony**
- **Arcadia**
- **Asmparts**
- **Athena**
- **AutoSBW**
- **AVIS**
- **BANSA**
- **BASIS**
- **BetaWB**
- **Bifurcation Discovery Tool**
- **BIOG**
- **BiNoM**
- **BiNoM Cytoscape Plugin**
- **BioSketchPad**

This matrix was generated on 2011-07-19 (total number of tools: 230)

Updated: Software Summary

SBML Software Summary

This page lists software known to us to provide some degree of support for reading, writing, or otherwise working with SBML. For an at-a-glance matrix summarizing key features of these software packages, please see our SBML Software Matrix page. Please use the survey form to notify us about additions and suggestions.

Note that several of the ODE/DAE-based simulators also include some form of stochastic simulation capability, and vice-versa. Also most of the model simulation, development, and analysis tools listed elsewhere on this page include some form of visualization.

Analysis software

- **ABC-SysBio** — ABC-SysBio implements likelihood free parameter inference and model selection in dynamical systems. It is designed to work with both stochastic and deterministic models written in Systems Biology Markup Language (SBML). ABC-SysBio is a Python package that combines three algorithms: ABC rejection sampler, ABC SMC for parameter inference and ABC SMC for model selection.

- **AutoSBW** — A frontend around AUTO to simplify bifurcation analysis.

- **Bifurcation Discovery Tool** — The Bifurcation Discovery Tool uses a genetic algorithm to search for Hopf bifurcations, turning points, and bistable switches. The user can select parameters to be searched, admissible parameter ranges, and the nature of the bifurcation to be sought. The tool returns parameter values for the model for which the particular behavior is observed.

http://sbml.org/SBML_Software_Guide/SBML_Software_Summary
New: Software Showcase

SBML Software Showcase

The following slideshow of SBML-compatible software systems contains the subset of packages for which we were provided screenshots by the software's authors. For a complete listing of SBML software, see the SBML Software Matrix, and for a list of longer descriptions grouped into themes, please see our SBML Software Summary page.

JarnaCLite

JarnaCLite allows to edit SBML through a simple script based format. It is integrated with SBK, so that a model created with JarnaCLite can be quickly simulated and analyzed with a variety of tools. JarnaCLite is available on all platforms, and as translator also online. It is released under the BSD license.

http://sbml.org/SBML_Software_Guide/SBML_Software_Showcase
In Development: Software Index

http://sysbioapps.dyndns.org/SoftwareMatrix
SBML TOOLBOX 4.0.1
SBMLToolbox 4.0.1
SBML Toolbox 4.0.1

- import and export removed to libSBML
- improved simulation
  (passes 450 tests from SBML Test-suite)
- improved creation of SBML models
- removed GUI support
  (fully octave compatible)
- improved documentation
SBMLToolbox 4.0 API Manual

This manual describes the application programming interface (API) of SBMLToolbox, an open-source (GPL) MATLAB/Octave toolbox for writing and manipulating content in the Systems Biology Markup Language (SBML). This version of SBMLToolbox supports all releases of SBML up through Level 3 Version 1 Core Base Level 1. For more information about SBML, please visit http://sbml.org or the Internet.

All functions can be used in both the MATLAB and Octave environments.

SBMLToolbox 4.0 API Manual

ACCESSMODEL

The AccessModel folder contains a number of functions that derive information from the MATLAB SBML structures.

Function: ...

array = DetermineSpeciesRoleInReaction(SBMLSpecies, SBMLReaction)

Usage:

array = DetermineSpeciesRoleInReaction(SBMLSpecies, SBMLReaction)

Takes

1. SBMLSpecies, an SBML species structure
2. SBMLReaction, an SBML reaction structure

Returns

1. an array with five elements [isProduct, isReactant, isModifier, positionInProductList, positionInReactantList] indicating whether the species is a product, reactant or modifier and their position in the list of products/reactants

or

1. array = 0 if the species is NOT part of the reaction

EXAMPLE:

y = DetermineSpeciesRoleInReaction(a, z)

= 0

= [1, 0, 0, 2, 0] if z is product number 2 in reaction

= [0, 1, 0, 0, 1] if z is a reactant number 1 in reaction

= [0, 0, 1, 0, 0] if z is a modifier in reaction

= [1, 1, 0, 1, 2] if z is product number 1 and reactant number 2

[names, values] = GetARParameters(SBMLModel)

Takes

1. SBMLModel, an SBML Model structure

Returns

1. an array of strings representing the identifiers of all parameters (both global and embedded) within

2. an array of the values of each parameter

NOTE: the value returned will be in (order)

- determined from assignmentRules/initialAssignments where appropriate
- the attribute value for the given parameter
- NaN, if the value is not specified in any way within the model

http://sbml.org/Software/SBMLToolbox/SBMLToolbox_4.0_API_Manual
Acknowledgements

• Arsen Batagov
• Ben Bornstein
• Will Bryant
• Bill Denney
• Andrew Finney

• Thomas Grotkjær
• Pieter Pareit
• Kieran Smallbone
• Sumant Turlapati
SBML Team

Frank Bergmann
Caltech, USA

Lucian Smith
U. of Washington,
USA

Sarah Keating
EMBL-EBI, UK

Mike Hucka
Caltech, USA

Linda Taddeo
Caltech, USA

Nicolas Rodriguez
EMBL-EBI, UK