MathSBML: A Mathematica Package For Systems Biology

Bruce E. Shapiro (1,2), Michael Hucka (1), and Andrew Finney (1,3)

(1) Control and Dynamical Systems, California Institute of Technology, Pasadena, CA, USA; (2) Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, USA; (3) Science and Technology Research Institute, University of Hertfordshire, Hatfield, UK

SBML: A Tool-Neutral Model Definition Format

SBML - the Systems Biology Markup Language - is a computer-readable format for representing models of biochemical reaction networks, applicable to:

- metabolic networks,
- cell-signaling pathways,
- genomic regulatory networks, and
- other modeling problems in system sbiology.

SBML is based on XML, a standard medium for representing and transporting data that is widely supported on the Internet as well as in computational biology and bioinformatics.

Because SBML is tool-independent, it enables

- use of multiple simulation and analysis tools in a single research project without rewriting models for each tool;
- publication of models in peer-reviewed journals - other researchers can download and use your model even if they use a different software environment;
- survival of models - they can outlive the software used to create them, making your work still useful even if a particular simulation package is no longer supported;

SBML is well on its way to becoming the *lingua franca* of computational systems biology. SBML is currently supported by over 60 tools including the following:

BALSA
BASIS
BioCharon
biocyc2SBML
BioGrid
BioNetGen
BioSketchPad
Dashboard
BioSpreadsheet
BioUML
BSTLab
CADLIVE
CellDesigner
Cellerator
Cellware
COPASI
Cytoscape
DBsolve
Dizzy
E-CELL
ecellJ
ESS
FluxAnalyzer
Gepasi
INSILICOdiscovery
Jarnac
JDesigner
JigCell
JSIM
JWS*
Karyote*
KEGG2SBML
Kinsolver*
libSBML
LION TargetEngine
MathSBML
MesoRD
MetaboLogica
MMT2
Modesto
Moleculizer
MOMA*
Monod
NetBuilder
PathArt
PathScout
PaVESy
PathwayBuilder
ProcessDB*
PySCeS*
SBW
SCIpath
Sigmoid*
SigPath
SigTran
Simpathica
SimWiz
StochSim
STOCKS
Trelis
VirtualCell
VLXSuite
WinSCAMP

*under development

Model Schematic

```
<?xml version="1.0" encoding="UTF-8"?>
<sbml xmlns="http://www.sbml.org/sbml/level2"
  level="2" version="1">
 <model id="ex1">
  <listOfFunctionDefinitions>
   ...
  </listOfFunctionDefinitions>
  <listOfUnitDefinitions>
   ...
  </listOfUnitDefinitions>
  <listOfCompartments>
   ...
  </listOfCompartments>
  <listOfSpecies>
   ...
   </listOfSpecies>
  <listOfParameters>
   ...
  </listOfParameters>
  <listOfRules>
   ...
  </listOfRules>
  <listOfReactions>
   ...
  </listOfReactions>
  <listOfEvents>
   ...
  </listOfEvents>
 </model>
</sbml>
```

MathSBML: A Package for Manipulating SBML Files

MathSBML is an open-source *Mathematica* package that facilitates working with SBML models. Features include:

- Import of SBML models to Mathematica
- Simulation and plotting of SBML models
- Simulation of models with events
- Simulation of differential-algebraic systems
- Export of models to SBML, HTML, Berkeley Madonna, XPP, Fortran formats
- Complete API for creating and modifying models
- Tabular model display and export to HTML
- Ability to use any Mathematica capability
- Open source / freely downloadable (LGPL)

Model Interoperability

With MathSBML investigators can explore SBML models with the full range of *Mathematica* features. *Mathematica* is one of several platforms widely used by biological modelers and is available in many academic and commercial environments (e.g., over 500 US colleges and universities have site licenses).

MathSBML provides full model interoperability with this environment as well as a candidate reference implementation of SBML. Support of other general-purpose languages, including C/C++, Java, Python, Perl, LISP and MATLAB have been developed separately using libSBML, a platform-independent library that is not part of MathSBML.

MathSBML supports SBML Level 1, Versions 1 and 2, and SBML Level 2. The MathSBML Model Editor supports SBML Level 2.

MathSBML will run on any platform that has *Mathematica* 4.1 or higher installed. The solution of differential-algebraic systems (SBML models that have algebraic rules) requires *Mathematica* 5.0 or higher; purely differential systems (SBML without algebraic rules) can be solved on *Mathematica* 4.1.

The Model Builder: An SBML API

MathSBML contains a simple model editor, allowing users to create SBML models compatible with other simulators, as well as a *Mathematica* text-command based API that can be used to produce arbitrarily complex SBML files. The model editor contains a suite of commands to add, modify, or remove single SBML objects (such as a reaction, chemical species, or equation) from the current model. The model may be either created de-novo or read from a file. After building the model, the user can test it by running simulations, continue to modify it, or write the results as an SBML file, in any order.

An Example

The following example illustrates a growing cell with a mitotic oscillator [Goldbeter, 1991] in which cell division is initiated when a variable passes a threshold. Cell division is indicated by defining a variable "Mass" that divides in half when both of the following tests are true at the same time: M > 0.7 and Mass > 0.6. The API commands to build and save the model to a file are shown in the box to the right. To read this file into the simulator, one could enter

```
s = SBMLRead["gmo.xml"];
```

A file on disk can be translated to html via

```
SBMLWrite[inputfile->"gmo.xml", outputfile->
"gmo.html", format-> "HTML"];
```

The output is illustrated in the large figure below. Other output formats include XPP, Berkeley Madonna and Fortran.

When building a model, the user can save it to a file, or test it with the simulator, at any time. For example,

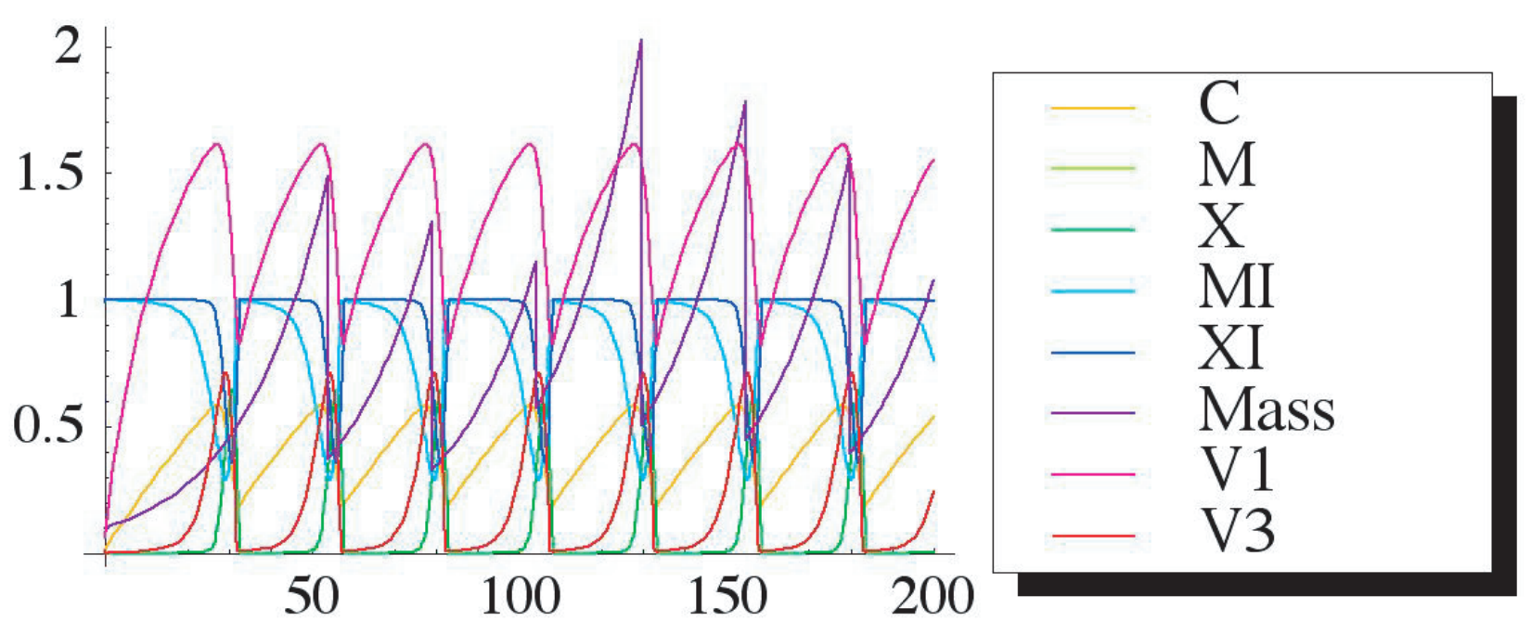
```
s = loadSimulator[];
```

returns a data structure (called s, in this case) that is compatible with the simulator; experienced Mathematica users can manipulate this data structure directly.

To run a simulation for, say, 200 time units, and plot all model variables, one would enter

```
n = SBMLNDSolve[s, 200];
plt = SBMLPlot[n];
```

The resulting screen plot is shown below.



API Commands to Build the Model

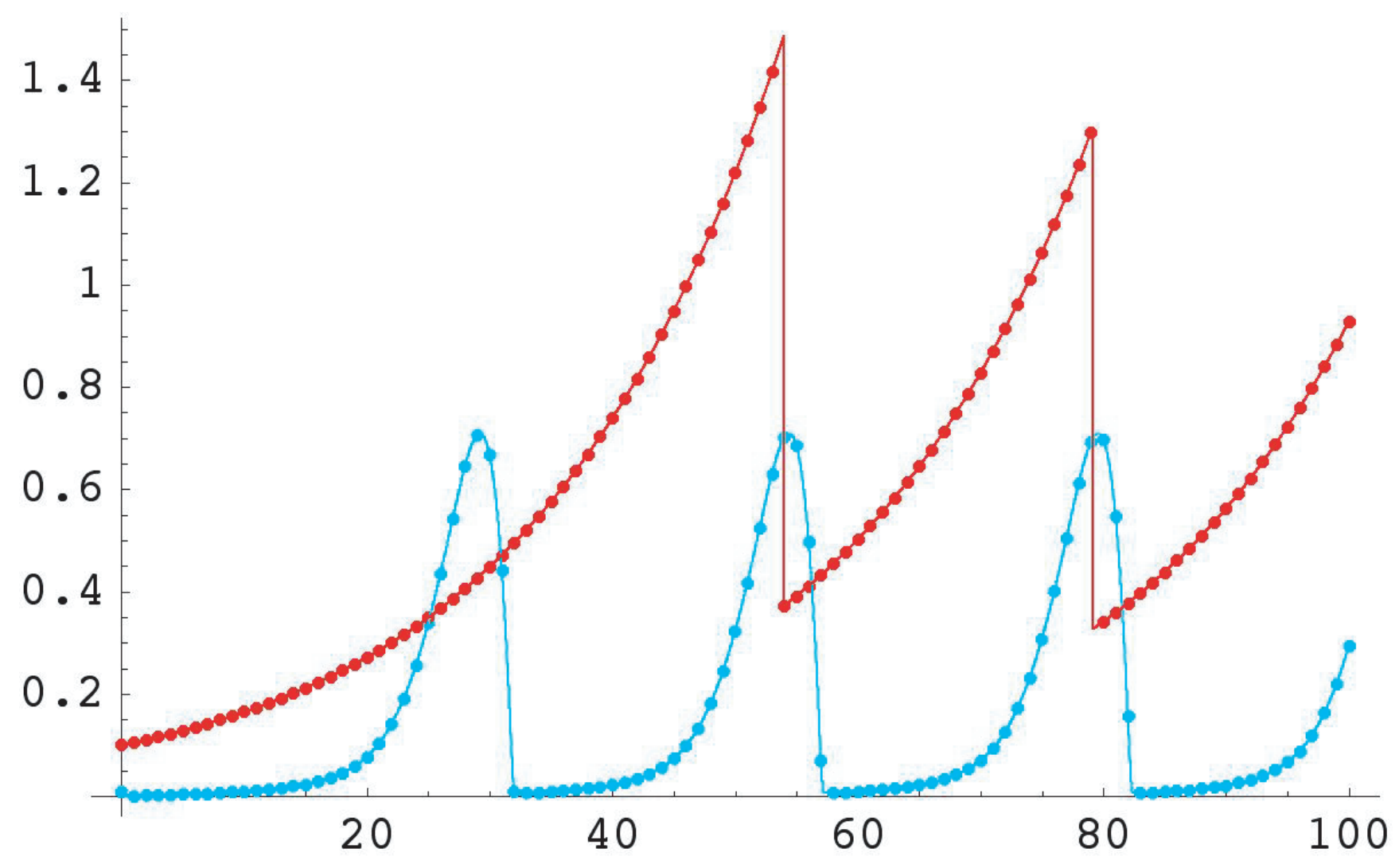
```
<<mathsbml.m;
newModel["GMO"];
addCompartment["cell"];
addFunction[id->mitosis,arguments->{m,mpf},math->
  ((m>0.6)&&(mpf>0.7))];
addSpecies[C,initialConcentration->0.01];
addSpecies[M,initialConcentration->0.01];
addSpecies[X,initialConcentration->0.01];
addSpecies[MI,initialConcentration->0.99];
addSpecies[XI,initialConcentration->0.99];
addParameter[Mass,value->0.1,constant->False];
addParameter[mu,value->0.05];
addParameter[V1,constant->False];
addParameter[V3,constant->False];
addParameter[VM1,value->3];
addParameter[VM3,value->1];
addParameter[Kc,value->0.5];
addRule[type->"RateRule",variable->Mass,
  math->mu*Mass];
addRule[type->"AssignmentRule",variable->V1,
  math-> C*VM1/(Kc+C)];
addRule[type->"AlgebraicRule",math->M*VM3-V3];
addReaction[products->{C},kineticLaw->(vi-kd*C),
  parameters->{vi->0.025,kd>0.01},
  reversible-> False];
addReaction[reactants->{C},modifiers->{X},
  kineticLaw ->vd*X*C/(Kd+C),parameters->{vd->0.25,
  Kd-> .02},reversible-> False];
addReaction[MI->M,kineticLaw->
  (V1*MI/(K1+MI)),reversible-> False,parameters->{
  K1-> .005}];
addReaction[M->MI,kineticLaw->V2*M/(K2+M),
  parameters->{V2-> 1.5, K2->.005},
  reversible->False];
addReaction[XI->X,kineticLaw->V3*XI/(K3+XI),
  parameters->{K3->.005 }, reversible->False];
addReaction[X->XI,kineticLaw->V4*X/(K4+X),
  parameters->{K4->.005, V4->.5},reversible->False];
add Event["CellDivision",trigger->mitosis[Mass,M],
  eventAssignment->{Mass->(Mass/2)}];

createModel["gmo.xml"];
```

Plots can be exported at any resolution and into any standard graphics format; other options allow one to plot specific variables, omit the lines in favor of interpolated points, or generate a table of data. For example, to plot values of model variables M and Mass at an interval of dt=1 and then draw a smooth line showing the complete plot:

```
p1 = SBMLListPlot[{GMO`M, GMO`Mass},
  {t, 0, 100, 1}, n, PlotJoined -> False,
  PlotStyle -> {PointSize[0.01]} ];
p2 = SBMLListPlot[{GMO`M, GMO`Mass},
  {t, 0, 100, .01}, n,
  PlotJoined-> True];
p3 = Show[{p1, p2}];
```

This plot is illustrated above to the right.



To generate a comma-separated value table of of the four variables C, M, X and Mass for $50 \le t \le 60$ at an interval of dt=1,

```
dataTable[{GMO`C, GMO`M, GMO`X, GMO`Mass}, {t, 50,
  60, 1}, n, format -> "CSV",file -> "data.csv"];
```

The contents of data.csv are illustrated:

```
time,GMO`C,GMO`M,GMO`X,GMO`Mass
50,0.5531960614058784,0.32297675469259335,0.008544577847197967,1.218249503953895
51,0.5694625611260513,0.41758093719068,0.01929209858863036,1.2807104916813272
52,0.5805031065678671,0.5239375719687274,0.05587927848922922,1.3463739227299842
53,0.5756801326474978,0.6291416198807009,0.15620661234571817,1.4154039898804824
54,0.5373906493972498,0.7013549355550314,0.3324825354021163,0.37199332637414895
55,0.4527248304099748,0.6847319906364808,0.5354327000099962,0.39106583205863366
56,0.330194470482806,0.4981814406885105,0.6415980510429778,0.4111162060234223
57,0.2178461614594566,0.06960985591427386,0.444781501835368 24,0.43219458464420035
58,0.19338295768478647,0.006216087295053213,0.0001306387689812787,0.4543536748467455
59,0.2163184042608536,0.007500819159890513,0.00007562049985154949,0.47764888704250336
60,0.23902241960847276,0.008987764654136041,0.00009088794848534929,0.502138469941423
```

Availability and Documentation

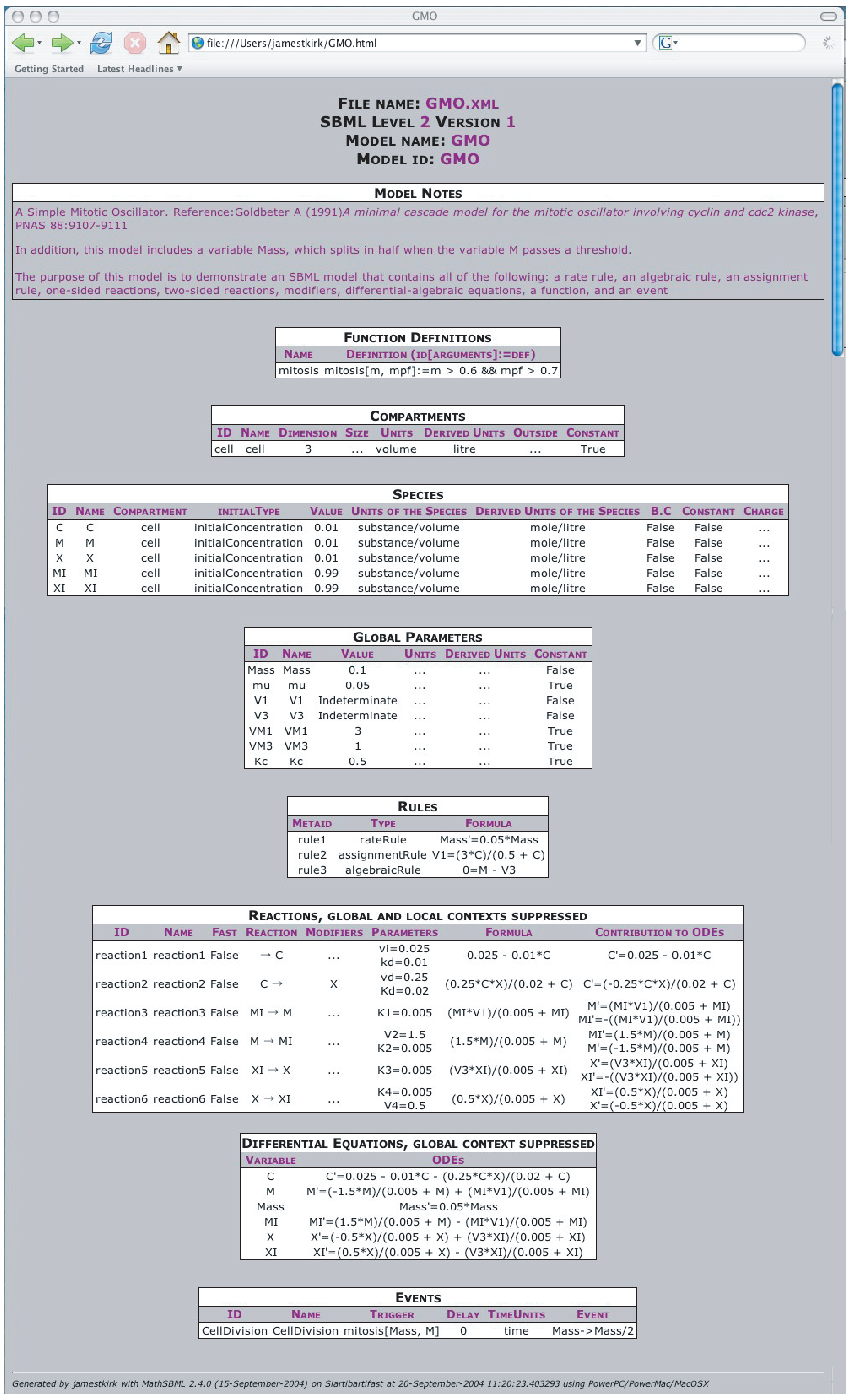
MathSBML is available free of charge from SourceForge under an LGPL license. It is open source and may be downloaded with a single click from our download site at:

http://sf.net/projects/sbml

Extensive documentation is included with the download, and is also available online at the SBML web site:

http://www.sbml.org/software/mathsbml/

For example, the site map contains links to documentation on all API function points, as illustrated in the figure below.



File name: GMO.xml
SBML Level 2 Version 1
Model name: GMO
Model ID: GMO

Model Notes

A Simple Mitotic Oscillator. Reference:Goldbeter A (1991)*A minimal cascade model for the mitotic oscillator involving cyclin and cdc2 kinase*, PNAS 88:9107-9111

In addition, this model includes a variable Mass, which splits in half when the variable M passes a threshold.

The purpose of this model is to demonstrate an SBML model that contains all of the following: a rate rule, an algebraic rule, an assignment rule, one-sided reactions, two-sided reactions, modifiers, differential-algebraic equations, a function, and an event

Function Definitions

Name	Definition (id[arguments]:=def)
mitosis	mitosis[m, mpf]:=m > 0.6 && mpf > 0.7

Compartments

ID	Name	Dimension	Size	Units	Derived Units	Outside	Constant
cell	cell	3	...	volume	litre	...	True

Species

ID	Name	Compartment	initialType	Value	Units of the Species	Derived Units of the Species	B.C	Constant	Charge
C	C	cell	initialConcentration	0.01	substance/volume	mole/litre	False	False	...
M	M	cell	initialConcentration	0.01	substance/volume	mole/litre	False	False	...
X	X	cell	initialConcentration	0.01	substance/volume	mole/litre	False	False	...
MI	MI	cell	initialConcentration	0.99	substance/volume	mole/litre	False	False	...
XI	XI	cell	initialConcentration	0.99	substance/volume	mole/litre	False	False	...

Global Parameters

ID	Name	Value	Units	Derived Units	Constant
Mass	Mass	0.1	...	...	False
mu	mu	0.05	...	...	True
V1	V1	Indeterminate	...	...	False
V3	V3	Indeterminate	...	...	False
VM1	VM1	3	...	...	True
VM3	VM3	1	...	...	True
Kc	Kc	0.5	...	...	True

Rules

Metaid	Type	Formula
rule1	rateRule	Mass'=0.05*Mass
rule2	assignmentRule	V1=(3*C)/(0.5 + C)
rule3	algebraicRule	0=M - V3

Reactions, global and local contexts suppressed

ID	Name	Fast	Reaction	Modifiers	Parameters	Formula	Contribution to ODEs
reaction1	reaction1	False	→ C	...	vi=0.025 kd=0.01	0.025 - 0.01*C	C'=0.025 - 0.01*C
reaction2	reaction2	False	C →	X	vd=0.25 Kd=0.02	(0.25*C*X)/(0.02 + C)	C'=(-0.25*C*X)/(0.02 + C)
reaction3	reaction3	False	MI → M	...	K1=0.005	(MI*V1)/(0.005 + MI)	M'=(MI*V1)/(0.005 + MI) MI'=-((MI*V1)/(0.005 + MI))
reaction4	reaction4	False	M → MI	...	V2=1.5 K2=0.005	(1.5*M)/(0.005 + M)	MI'=(1.5*M)/(0.005 + M) M'=(-1.5*M)/(0.005 + M)
reaction5	reaction5	False	XI → X	...	K3=0.005	(V3*XI)/(0.005 + XI)	X'=(V3*XI)/(0.005 + XI) XI'=-((V3*XI)/(0.005 + XI))
reaction6	reaction6	False	X → XI	...	K4=0.005 V4=0.5	(0.5*X)/(0.005 + X)	XI'=(0.5*X)/(0.005 + X) X'=(-0.5*X)/(0.005 + X)

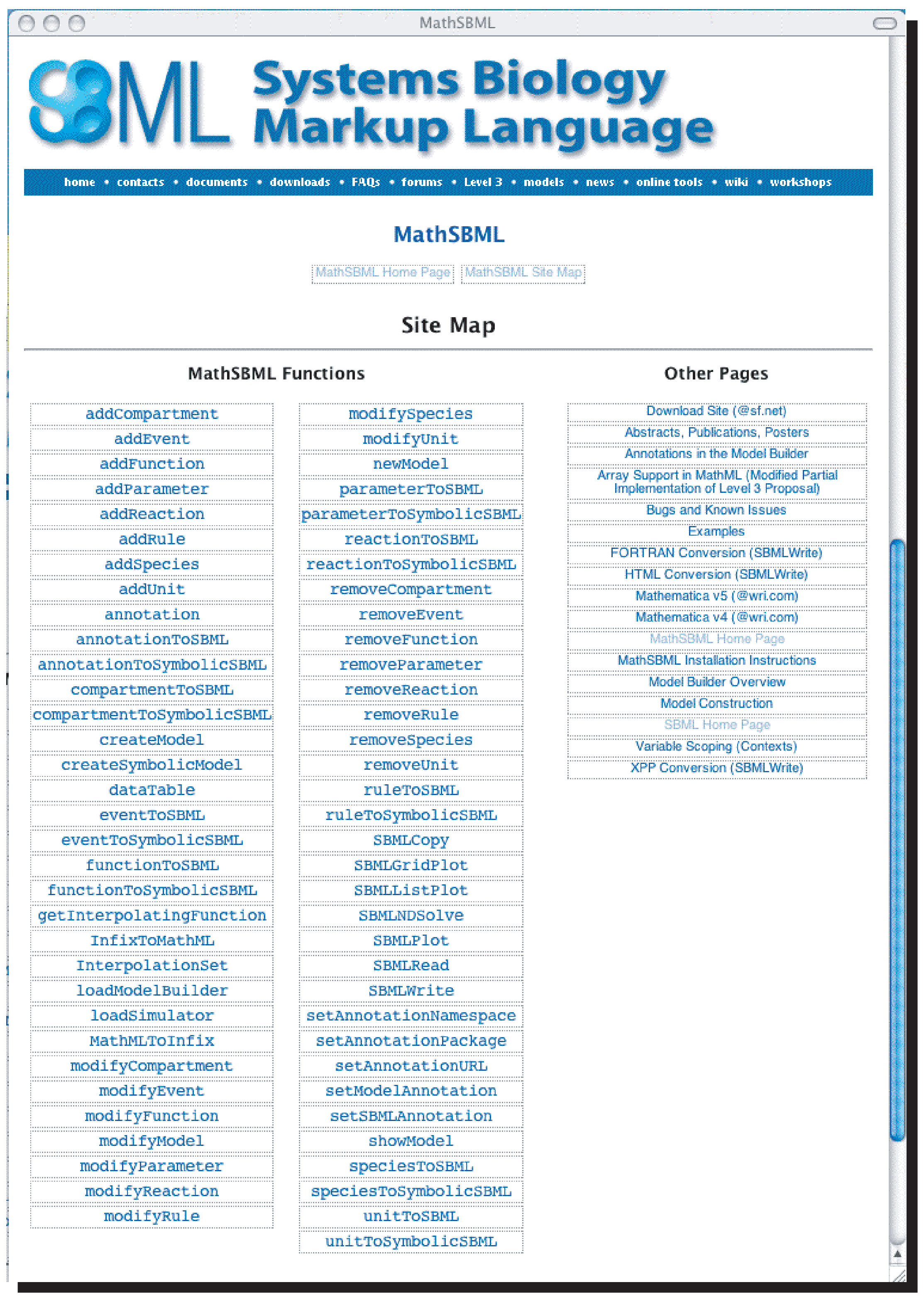
Differential Equations, global context suppressed

Variable	ODEs
C	C'=0.025 - 0.01*C - (0.25*C*X)/(0.02 + C)
M	M'=(-1.5*M)/(0.005 + M) + (MI*V1)/(0.005 + MI)
Mass	Mass'=0.05*Mass
MI	MI'=(1.5*M)/(0.005 + M) - (MI*V1)/(0.005 + MI)
X	X'=(-0.5*X)/(0.005 + X) + (V3*XI)/(0.005 + XI)
XI	XI'=(0.5*X)/(0.005 + X) - (V3*XI)/(0.005 + XI)

Events

ID	Name	Trigger	Delay	TimeUnits	Event
CellDivision	CellDivision	mitosis[Mass, M]	0	time	Mass->Mass/2

Generated by jamestkirk with MathSBML 2.4.0 (15-September-2004) on Slartibartifast at 20-September-2004 11:20:23.403293 using PowerPC/PowerMac/MacOSX


MathSBML
Systems Biology Markup Language
home • contacts • documents • downloads • FAQs • forums • Level 3 • models • news • online tools • wiki • workshops
MathSBML Home Page
MathSBML Site Map

MathSBML

Site Map

MathSBML Functions

addCompartment
addEvent
addFunction
addParameter
addReaction
addRule
addSpecies
addUnit
annotation
annotationToSBML
annotationToSymbolicSBML
compartmentToSBML
compartmentToSymbolicSBML
createModel
createSymbolicModel
dataTable
eventToSBML
eventToSymbolicSBML
functionToSBML
functionToSymbolicSBML
getInterpolatingFunction
InfixToMathML
InterpolationSet
loadModelBuilder
loadSimulator
MathMLToInfix
modifyCompartment
modifyEvent
modifyFunction
modifyModel
modifyParameter
modifyReaction
modifyRule
modifySpecies
modifyUnit
newModel
parameterToSBML
parameterToSymbolicSBML
reactionToSBML
reactionToSymbolicSBML
removeCompartment
removeEvent
removeFunction
removeParameter
removeReaction
removeRule
removeSpecies
removeUnit
ruleToSBML
ruleToSymbolicSBML
SBMLCopy
SBMLGridPlot
SBMLListPlot
SBMLNDSolve
SBMLPlot
SBMLRead
SBMLWrite
setAnnotationNamespace
setAnnotationPackage
setAnnotationURL
setModelAnnotation
setSBMLAnnotation
showModel
speciesToSBML
speciesToSymbolicSBML
unitToSBML
unitToSymbolicSBML

Other Pages

Download Site (@sf.net)
Abstracts, Publications, Posters
Annotations in the Model Builder
Array Support in MathML (Modified Partial Implementation of Level 3 Proposal)
Bugs and Known Issues
Examples
FORTRAN Conversion (SBMLWrite)
HTML Conversion (SBMLWrite)
Mathematica v5 (@wri.com)
Mathematica v4 (@wri.com)
MathSBML Home Page
MathSBML Installation Instructions
Model Builder Overview
Model Construction
SBML Home Page
Variable Scoping (Contexts)
XPP Conversion (SBMLWrite)

Acknowledgements

MathSBML was developed at the California Institute of Technology. Support has been provided by ERATO Kitano Project, the US National Institutes of Health, and the US National Science Foundation.

References

Finney A, Hucka M, "Systems biology markup language: Level 2 and beyond," *Biochem. Soc. Trans.* (2003) 31:1472-1473.

Goldbeter A "A minimal cascade model for the mitotic oscillator involving cyclin and cdc2 kinase," Proc Natl Acad Sci USA, 88:9107-9111 (1991).

Hucka M et al, "The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models", *Bioinformatics* 19: 524-531 (2003).

Hucka M et al, "Evolving a lingua franca and associated software infrastructure for computational systems biology: the Systems Biology Markup Language Project," IEE Systems Biology, 1:41-53 (2004).

Shapiro B et al, "MathSBML: a package for manipulating SBML-based biological models," Bioinformatics, doi:10.1093/bioinformatics/bth271 (2004).

http://sbml.org

JPL CL#04-0745