SBML and related resources for computational systems biology

Dr. Michael Hucka *(on behalf of many people)*
California Institute of Technology
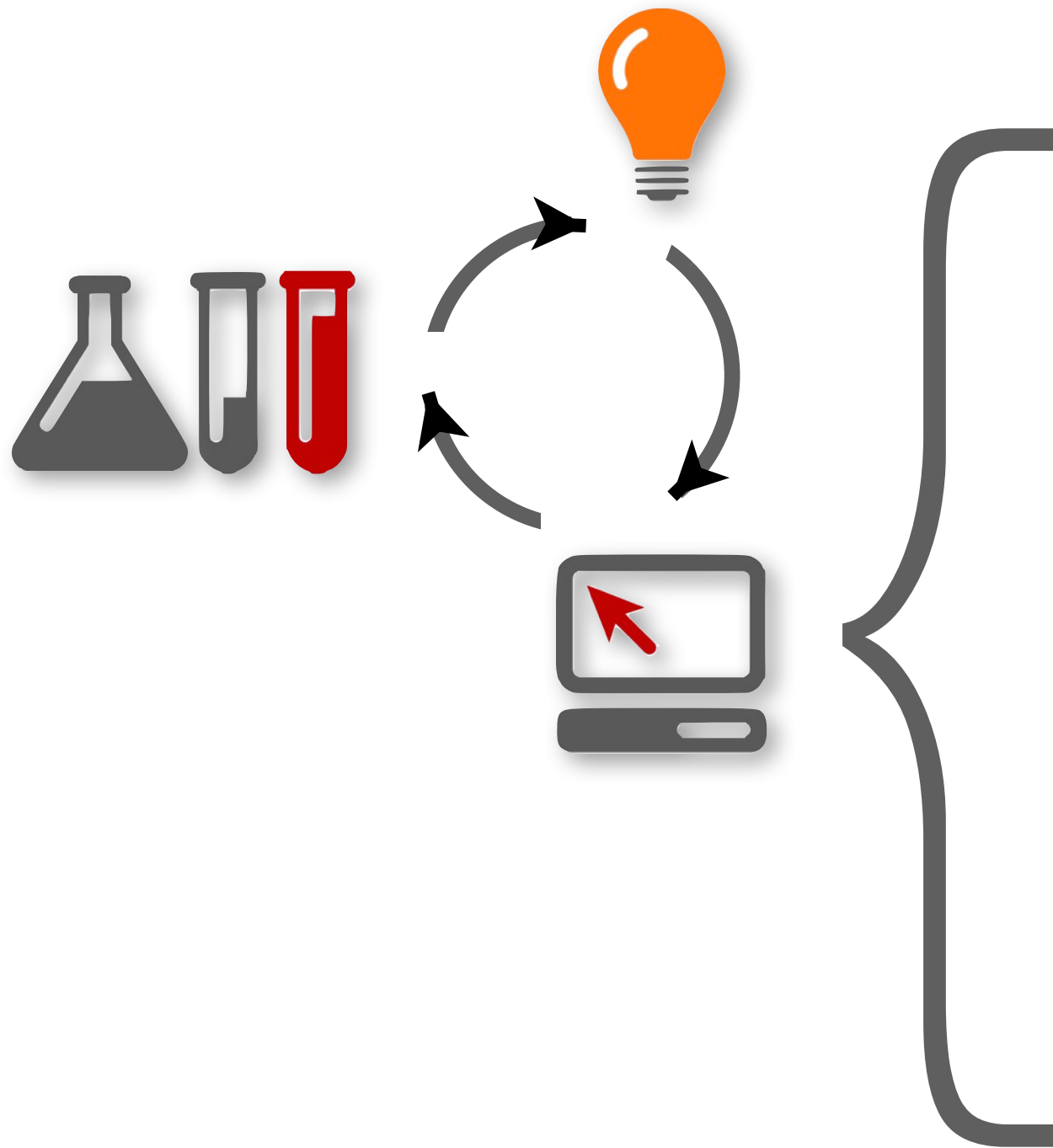*Pasadena, California, USA*

Email: *mhucka@caltech.edu*   *Twitter:* *@mhucka*

Outline

- SBML (Systems Biology Markup Language)

- A sample of SBML software resources

- A sample of related resources and efforts

- COMBINE

- Acknowledgments

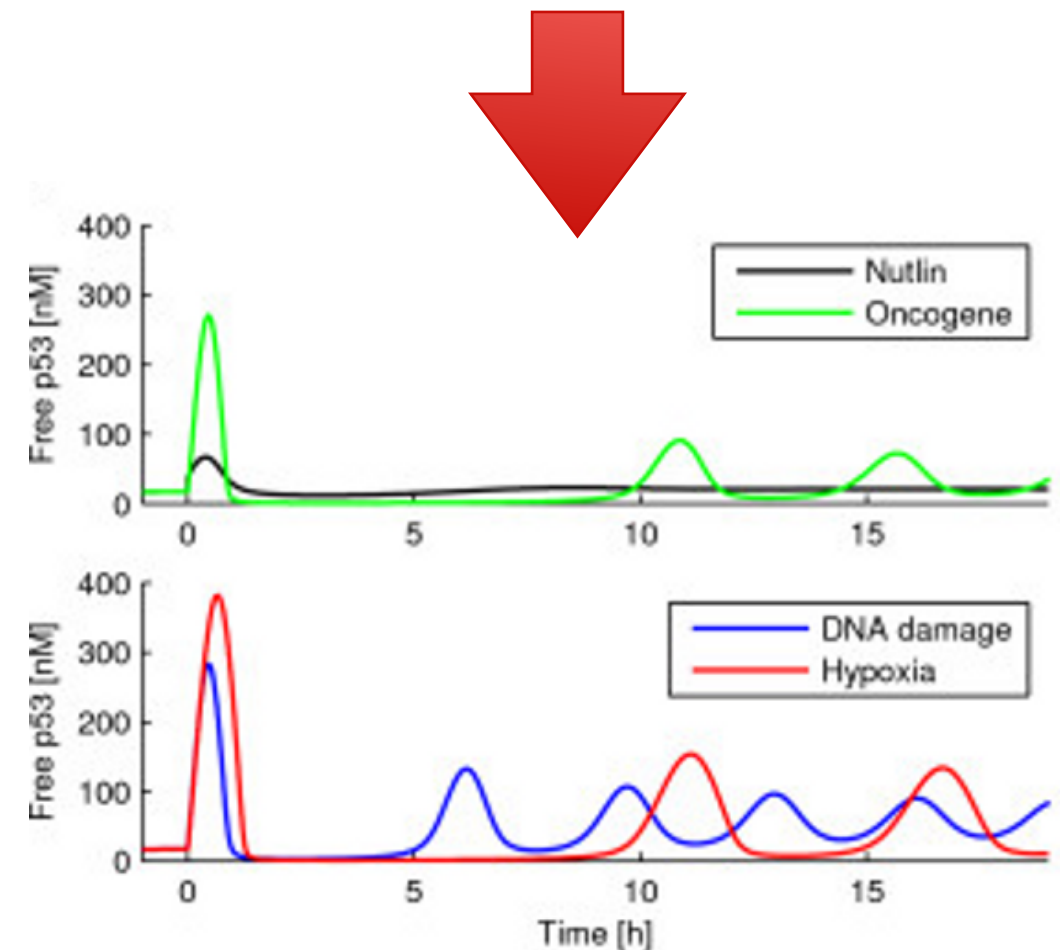# Introduction to SBML

Background context: formal models

$$\frac{d[p]}{dt} = \sigma - \alpha[p] - k_f[m][p] + k_b[c] + \gamma[c]$$

$$\frac{d[m_m]}{dt} = k_t[p]^2 - \beta m_m$$

$$\frac{d[m]}{dt} = k_{tl}[m_m] - k_f[p][m] + k_b[c] + \delta[c] - \gamma[m]$$

$$\frac{d[c]}{dt} = k_f[p][m] - k_b[c] - \delta[c] - \gamma[c]$$

Many software tools for modeling and simulation are available

ABC–SysBio	CellNetAnalyzer	Karyote*	PaVESy	SBW: Auto Layout
acslXtreme	CellNOpt	KEGGconverter	PAYAO	sbw: javasim
ALC	Cellware	KEGGtranslator	PET	sbw: stochastic simulator
AMIGO	CLEML	Kineticon	PhysioLab Modeler	SCIpath
Antimony	CL–SBML	Kinsolver	PINT	SED–ML Web Tools
APMonitor	COBRA	libAnnotationSBML	PK–Sim / MoBi	semanticSBML
Arcadia	CompuCell3D	libRoadRunner	PNK	SensSB
Asmparts	ConsensusPathDB	libSBML	PottersWheel	SGMP
Athena	COPASI	libSBMLSim	PRISM	Sigmoid*
AutoSBW	CRdata	libStruct	ProcessDB	SIGNALIGN
AVIS	CycSim	MASS Toolbox	ProMoT	SignaLink
BALSA	CySBML	MatCont	PROTON	SigPath
BASIS	Cytoscape	MathSBML	pybrn	SigTran
BetaWB	Cyto–Sim	Medicel	PyDSTool	SIMBA
Bifurcation Discovery Tool	DBSolve	MEMOSys	PySB	SimBiology
BiGG	DEDiscover	MesoRD	PySCeS	Simpathica
BiNoM	Dizzy	Meta–All	RANGE	SimPheny*
BiNoM Cytoscape Plugin	DOTcvpSB	Metaboflux	RAVEN	Simulate3D
Bio Sketch Pad	E–CELL	MetaCrop	Reactome	Simulation Core Library
BioBayes	ecellJ	MetaFluxNet	ReMatch	Simulation Tool
BIOCHAM	EPE	Metannogen	RMBNToolbox	SimWiz
BioCharon	ESS	Metatool	roadRunner	SloppyCell
BioCyc	Facile	MetExplore	RSBML	SmartCell
BioGRID	FAME	MetNetMaker	SABIO–RK	Snoopy
Biological Networks	FASIMU	MIRIAM Resources	Saint	SOSlib
BioMet Toolbox	FBASBW	MMT2	SBFC	SPDBS
BioModels Database	FERN	modelMaGe	SBML Harvester	SRS
BioModels Importer	FluxBalance	ModeRator	SBML Layout	STEPS
BioNessie	Fluxor	Modesto	SBML Reaction Finder	StochKit
BioNetGen	Genetdes	Moleculizer	SBML Translators	StochPy
BioPARKIN	Genetic Network Analyzer	MonaLisa	SBML2APM	StochSim
BioPathwise	Gepasi	Monod	SBML2BioPax	STOCKS
BioPAX2SBML	Gillespie2	MOOSE	SBML2LaTeX	SurreyFBA
BioRica	GINsim	MuVal (Multi–valued logic)	SBML2NEURON	SyBiL
BioSens	GNAT	Narrator	SBML2Octave	SYCAMORE
BioSPICE Dashboard	GNU MCSim	nemo	SBML2SMW	SynBioSS
BioSpreadsheet	GRENDEL	NetBuilder'	SBML2TikZ	Systrip
BioSyS	HSMB	NetPath	SBML2XPP	TERANODE Suite
BioTapestry	HybridSBML	NetPro	SBMLEditor	The Cell Collective
BioUML	iBioSim	Odefy	SBML–PET–MPI	Tide
BoolNet	IBRENA	Omix	SBMLR	TinkerCell
braincirc	Insilico Discovery	ONDEX	SBML–SAT	Trelis
BRENDA	insilicoIDE	optflux	SBML–shorthand	UTKornTools
BSTLab	iPathways	Oscill8	SBMLSim	VANTED
ByoDyn	JACOBIAN	PANTHER Pathway	SBMLsqueezer	Vcell
CADLIVE	Jacobian Viewer	PathArt	sbmltidy	WebCell
Cain	Jarnac	Pathway Access	SBMLToolbox	WinSCAMP
CARMEN	JarnacLite	Pathway Analyser	SBMM assistant	Wolfram SystemModeler
Cell Illustrator				xCellerator
CellDesigner				Xholon
Cellerator				XPPAUT
CellMC				
CellML2SBML				

Research often involves the use of more than one tool

Need flexible way to exchange results between tools (and researchers)

SBML = Systems Biology Markup Language

Format for **representing** models of biological processes

- Data structures + principles + serialization to XML

- (Mostly) Declarative, not procedural—not a scripting language

- For **software** to read/write, **not humans**

(Mostly) **neutral** with respect to modeling framework

- E.g., ODE, stochastic systems, etc.

Does **not** store experimental data

Does **not** store simulation descriptions

Does define how annotations may be written within an SBML file

SBML is a file format based on XML

```xml
<listOfSpecies>
    <species compartment="cytosol" id="ES" initialAmount="0"     name="ES"/>
    <species compartment="cytosol" id="P"  initialAmount="0"     name="P"/>
    <species compartment="cytosol" id="S"  initialAmount="1e-20" name="S"/>
    <species compartment="cytosol" id="E"  initialAmount="5e-21" name="E"/>
</listOfSpecies>
<listOfReactions>
    <reaction id="veq">
        <listOfReactants>
            <speciesReference species="E"/>
            <speciesReference species="S"/>
        </listOfReactants>
        <listOfProducts>
            <speciesReference species="ES"/>
        </listOfProducts>
        <kineticLaw>
            <math xmlns="http://www.w3.org/1998/Math/MathML">
                <apply>
                    <times/>
                    <ci>cytosol</ci>
                    <apply>
                        <minus/>
                        <apply>
                            <times/>
                            <ci>kon</ci>
                            <ci>E</ci>
                            <ci>S</ci>
                        </apply>
                        <apply>
                            <times/>
                            <ci>koff</ci>
                            <ci>ES</ci>
                        </apply>
```
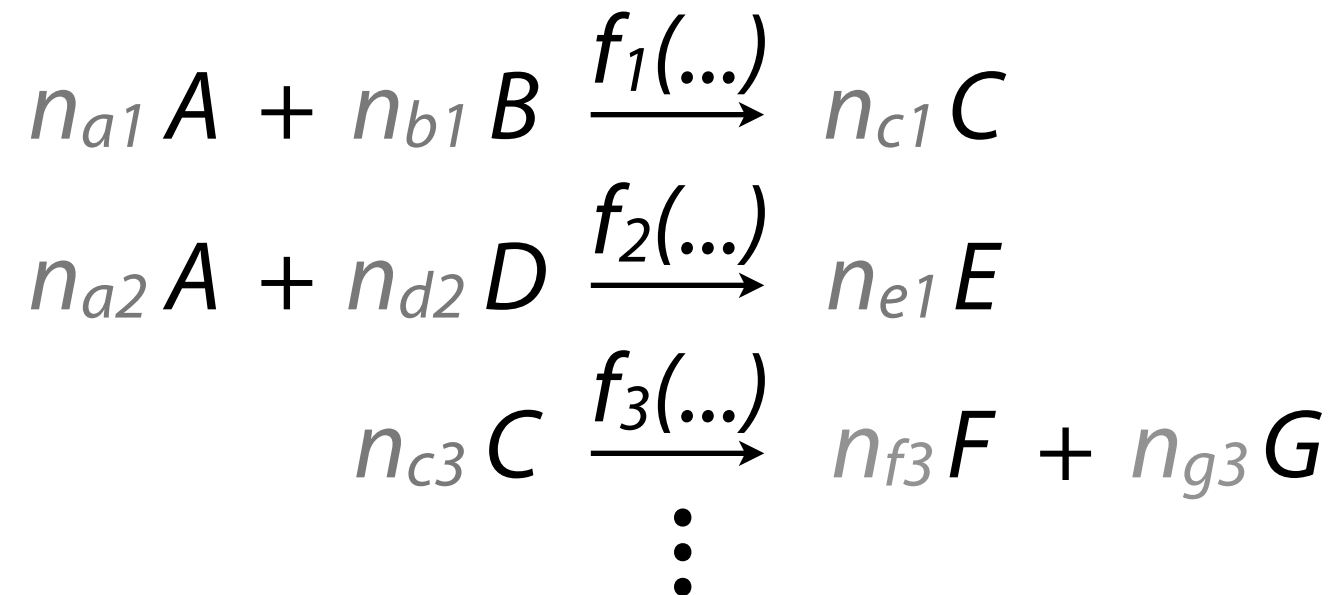
```
<listOfSpecies>
    <species compartment="cytosol" id="ES" initialAmount="0"     name="ES"/>
    <species compartment="cytosol" id="P"  initialAmount="0"     name="P"/>
    <species compartment="cytosol" id="S"  initialAmount="1e-20" name="S"/>
    <species compartment="cytosol" id="E"  initialAmount="5e-21" name="E"/>
</listOfSpecies>
<listOfReactions>
    <reaction id="veq">
        <listOfReactants>
            <speciesReference species="E"/>
            <speciesReference species="S"/>
        </listOfReactants>
        <listOfProducts>
            <speciesReference species="ES"/>
        </listOfProducts>
        <kineticLaw>
```

Don't work with it directly! Let software do it.

```
                <apply>
                    <times/>
                    <ci>cytosol</ci>
                    <apply>
                        <minus/>
                        <apply>
                            <times/>
                            <ci>kon</ci>
                            <ci>E</ci>
                            <ci>S</ci>
                        </apply>
                        <apply>
                            <times/>
                            <ci>koff</ci>
                            <ci>ES</ci>
                        </apply>
```

The *process* is central

- Literally called "**reaction**" (not necessarily biochemical)

- Participants are pools of entities of the same kind ("**species**")

$$n_{a1}\,A + n_{b1}\,B \xrightarrow{f_1(...)} n_{c1}\,C$$

$$n_{a2}\,A + n_{d2}\,D \xrightarrow{f_2(...)} n_{e1}\,E$$

$$n_{c3}\,C \xrightarrow{f_3(...)} n_{f3}\,F + n_{g3}\,G$$

$$\vdots$$

- Species are located in containers ("**compartments**")

  - *Core* SBML assumes well-mixed compartments (but see Level 3)

Models can further include:

- Explicit math expressions

- Discontinuous **events**

- Unit definitions

- **Annotations**

- Other constants & variables

Core SBML concepts are fairly simple

Example of model type

Typical ODE models (e.g., cell differentiation)

Conductance-based models (e.g., Hodgin-Huxley)

Typically do not use SBML "reaction" construct, but instead use "rate rules" construct

Neural models (e.g., spiking neurons)

Typically use "events" for discontinuous changes

Pharmacokinetic/dynamics models

"Species" are not required to be biochemical entities

Infectious diseases

Example model

BioModels Database model
#BIOMD0000000451

BioModels Database model
#BIOMD0000000020

BioModels Database model
#BIOMD0000000127

BioModels Database model
#BIOMD0000000234

BioModels Database model
#MODEL1008060001

List originally by Nicolas Le Novére

Core SBML constructs support many types of models

A community-driven global reconstruction of human metabolism

Ines Thiele[1,2,37], Neil Swainston[3,4,37], [
Maike K Aurich[1], Hulda Haraldsdotti[
Stefan G Thorleifsson[1], Rasmus Agren[
Paul Dobson[12], Warwick B Dunn[3,13],
Daniel Jameson[3,4], Neema Jamshidi[7],
Nicolas Le Novère[17,18], Naglis Malys[3,
Evgeni Selkov, Sr[23], Martin I Sigurdsson[
Anatoly Sorokin[21,27], Johannes H G M
Hans V Westerhoff[3,28,33,34,

Multiple
describe
of huma
improve
we pred
experim
analyzin
models,
future b

An und
the phen
where n
human d

Sear

Quantitative Biology > Molecular Networks

Striking a balance with Recon 2.1

Kieran Smallbone

Recon 2 is a highly curated reconstruction of the human metabolic network. Whilst the network is state of the art, it has shortcomings, including the presence of unbalanced reactions involving generic metabolites. By replacing these generic molecules with each of their specific instances, we can ensure full elemental balancing, in turn allowing constraint-based analyses to be performed. The resultant model, called Recon 2.1, is an order of magnitude larger than the original

A consensus yeast metabolic network reconstruction obtained from a community approach to systems biology

Markus J Herrgård[1,19,20], Neil Swainston[2,3,20], Paul Dobson[3,4], Warwick B Dunn[3,4], K Yalçin Arga[5], Mikko Arvas[6], Nils Blüthgen[3,7], Simon Borger[8], Roeland Costenoble[9], Matthias Heinemann[9], Michael Hucka[10]

SBML enables large, collaborative, model-building efforts

Accepted by dozens of journals *

100's of software tools available today

- 260+ listed in SBML Software Guide †

1000's of models available

- … in public databases

- … as supplementary data to papers

- … in private repositories

http://sbml.org

* http://sbml.org/Documents/Publications_known_to_accept_submissions_in_SBML_format

† http://sbml.org/SBML_Software_Guide

Many models and software resources are available today

SBML Editors

Andreas Dräger

Sarah Keating
Coordinator
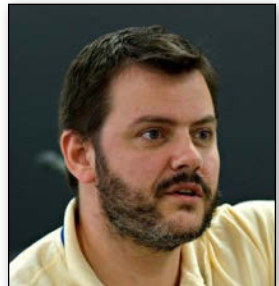
Nicolas Le Novère

Brett Olivier

Lucian Smith
Secretary

Dagmar Waltemath

Current　　**Past**

Frank Bergmann

Hamid Bolouri

Andrew Finney

Stefan Hoops

Michael Hucka

Chris Myers

Sven Sahle

Herbert Sauro

James Schaff

Dagmar Waltemath

Darren Wilkinson

SBML Level 1	SBML Level 2	SBML Level 3
predefined math functions	user-defined functions	user-defined functions
text-string math notation	MathML subset	MathML subset
reserved namespaces for annotations	no reserved namespaces for annotations	no reserved namespaces for annotations
no controlled annotation scheme	RDF-based controlled annotation scheme	RDF-based controlled annotation scheme
no discrete events	discrete events	discrete events
default values defined	default values defined	no default values
monolithic	monolithic	**modular & extensible**

Level 3 packages add constructs on top of **SBML Level 3 Core**

Level 3 package	What it supports	Status
Hierarchical model composition	Models composed from other models/parts	✔
Flux balance constraints	Constraint-based (a.k.a. steady-state) models	✔
Qualitative models	Petri net, Boolean, and similar model types	✔
Graph layout	Storing layouts of network diagrams	✔
Groups	Grouping elements for conceptual purposes	✔
Arrays	Arrays of components	🕐
Distributions	Statistical distributions of values	🕐
Multicomponent/state species	Rule-based descriptions of entities with features	🕐
Spatial	Nonhomogeneous spatial models	🕐
Graph rendering	Storing graphical symbols used in Layout diagrams	🕐
Dynamic structures	Creation/destruction of entities during simulation	🕐
Extended Math	Additional MathML constructs	🕐

Level 3 package	What it supports	Status
Hierarchical model composition	Models composed from other models/parts	✔
Flux balance constraints	Constraint-based (a.k.a. steady-state) models	✔
Qualitative models	Petri net, Boolean, and similar model types	✔
Graph layout	Storing layouts of network diagrams	✔
Groups	Grouping elements for conceptual purposes	✔
Arrays	Arrays of componen...	**Implementations being tested**
Distributions	Statistical distributi...	**Implementations being tested**
Multicomponent/state species	Rule-based descript...	**Implementations being tested**
Spatial	Nonhomogeneous...	**Implementations being tested**
Graph rendering	Storing graphical sy...	**Implementations being tested**
Dynamic structures	Creation/destruction of entities during simulation	🕐
Extended Math	Additional MathML constructs	🕐

SBML Level 3 package: Spatial Processes ("spatial")

Main components:

- Coordinate systems

- Patches of spatial geometries, called **domains**

- Mapping of SBML compartments, species, & parameters to domains

- Definition of **molecular transport** mechanisms (advection, diffusion, boundary conditions)

- Mapping of molecular transport mechanisms to domains

Developed mostly by Jim Schaff & Anu Lakshminarayana (VCell), with recent input and involvement from Devin Sullivan (U. Pittsburgh), Lucian Smith and Frank Bergmann

- Beta implementation for libSBML available today

http://sbml.org/Documents/Specifications/SBML_Level_3/Packages/spatial

Draft implementations available in several tools already

SBML Level 3 *Spatial* package
draft specification

Virtual Cell

COPASI

MCell and CellBlender

SBML & JSBML Team: Mike Hucka, Sarah Keating, Frank Bergmann, Lucian Smith, Andrew Finney, Herbert Sauro, Hamid Bolouri, Ben Bornstein, Maria Schilstra, Jo Matthews, Bruce Shapiro, Linda Taddeo, Akira Funahashi, Akiya Juraku, Ben Kovitz, Nicolas Rodriguez, Andreas Dräger, Alex Thomas

SBML Editors: Mike Hucka, Frank Bergmann, Andreas Dräger, Sarah Keating, Nicolas Le Novère, Chris Myers, Lucian Smith, Stefan Hoops, Brett Olivier, Sven Sahle, James Schaff, Dagmar Waltemath, Darren Wilkinson

SBML Package authors (so far): Duncan Berenguier, Frank Bergmann, Claudine Chaouiya, Andreas Dräger, Andrew Finney, Ralph Gauges, Colin Gillespie, Martin Ginkel, Tomás Helikar, Stefan Hoops, Sarah Keating, Anu Lakshminarayana, Nicolas Le Novère, Wolfram Liebermeister, Martin Meier-Schellerscheim, Stuart Moodie, Ion Moraru, Chris Myers, Aurélien Naldi, Brett Olivier, Ursula Rost, Lucian Smith, Sven Sahle, James Schaff, Devin P. Sullivan, Denis Thieffry, Martijn P. van Iersel, Leandro Watanabe, Katja Wengler, Darren Wilkinson, Maciej Swat (EBI), Fengkai Zhang

Many people contributed to the development of SBML

Outline

- SBML (Systems Biology Markup Language)
- A sample of SBML software resources
- A sample of related resources and efforts
- COMBINE
- Acknowledgments

API libraries for supporting SBML

libSBML

- Written in portable C++

  - Linux, Mac, Windows

- APIs for C, C++, C#, Java, JavaScript, MATLAB, Octave, Perl, PHP, Python, R, Ruby

- Reads, writes, validates SBML

- Many other features: e.g., unit checking & conversion

JSBML

- Pure Java

- API very similar to libSBML, but more Java-ish

- Reads, writes, manipulates SBML

- Additional Java-relevant APIs such as listeners

Both are free, open-source under LGPL

http://sbml.org/Software/libSBML

http://sbml.org/Software/JSBML

The Online SBML Validator checks SBML files

Reports whether an SBML file conforms to the SBML specifications

- Helpful when 3rd-party software complains, or fails to load a file

Offers browser-based user interface *and* RESTful network API

Recent changes:

- Support for **draft SBML Level 3 packages**

    - Traditionally used libSBML for officially released packages

    - Now *also* uses separate **RELAX NG schemas** for *draft* packages

- Offers to send files to **other services** for additional analyses

- Moved to faster hardware

Software updates by Frank Bergmann; RNG schemas by Sarah Keating

SBML.org — The Systems Biology Markup Language

News   Documents   Downloads   Forums   Facilities   Community   Events   About

Google Site Search

Parent pages: SBML.org / Facilities / Online SBML Validator

This tests the syntax and internal consistency of an SBML file. Passing this validator does not *guarantee* correctness, but it's the best automated assessment available.

Use the form below to upload your SBML content. You can also access this service via a **network API**. By using any part of this service, you agree to the terms of the **privacy notice**.

Sign in or **Register** with this service.

You can retrieve the result of a previously-scheduled validation run by entering its key here:

Submit

(E.g., **7bf66904-7d67-49bb-9127-bee77a8f96ba**)

| **Upload File** | **Submit URL** | **Paste SBML** |

Browse   Clear Queue   **Validate now**

Select an SBML file located on your computer. The file can be uncompressed, or compressed using **zip**, **gzip** or **bzip2**.   **Schedule for Validation**

Validation options:
- ☐ Check consistency of measurement units associated with quantities (SBML L2V4 rules 105nn)
- ☑ Check correctness and consistency of identifiers used for model entities (**SBML L2V4 rules 103nn**)
- ☑ Check syntax of MathML mathematical expressions (**SBML L2V4 rules 102nn**)
- ☑ Check validity of **SBO identifiers** ⧉ (if any) used in the model (SBML L2V4 rules 107nn)
- ☑ Perform static analysis of whether the model is **overdetermined**
- ☑ Perform additional checks for recommended good **modeling practices**
- ☑ Perform all other general SBML consistency checks (**SBML L2V4 rules 2nnnn; highly recommended**)
- ☐ Ignore SBML Level 3 packages that are not officially finalized

This validator supports all specifications of SBML through **Level 3 Version 1 Core** (Release 1), as well as all **Level 3 packages** for which specifications exist. **Limitations**: (1) For SBML Level 3 packages that have not been finalized, validation is limited to XML syntax *only*; consequently, a model may have other errors in its use of Level 3 package constructs yet still pass this validator. (2) SBML `<annotation>` element content is always only checked for proper XML syntax; the *content* of annotations is *not* validated; consequently, a model may have errors in its annotations yet still pass

http://sbml.org/Facilities/Validator

SBML.org

The Systems Biology Markup Language

News   Documents   Downloads   Forums   **Facilities**   Community   Events   About   Google Site Search

Parent pages: SBML.org / Facilities / Online SBML Validator

This tests the syntax and internal consistency of an SBML file. Passing this validator does not *guarantee* correctness, but it's the best automated assessment available.

Use the form below to upload your SBML content. You can also access this service via a **network API**. By using any part of this service, you agree to the terms of the **privacy notice**.

Sign in or **Register** with this service.

You can retrieve the result of a previously-scheduled validation run by entering its key here:

[   ]   **Submit**

(E.g., **7bf66904-7d67-49bb-9127-bee77a8f96ba**)

| **Upload File** | **Submit URL** | **Paste SBML** |

Browse   **Clear Queue**   **Validate now**

Select an SBML file located on your computer. The file can be uncompressed, or compressed using **zip**, **gzip** or **bzip2**.

Schedule for Validation

Validation options:
- ☐ Check consistency of measurement units associated with quantities (**SBML L2V4 rules 105nn**)
- ☑ Check correctness and consistency of identifiers used for model entities (**SBML L2V4 rules 103nn**)
- ☑ Check syntax of MathML mathematical expressions (**SBML L2V4 rules 102nn**)
- ☑ Check validity of **SBO identifiers** (if any) used in the model (**SBML L2V4 rules 107nn**)
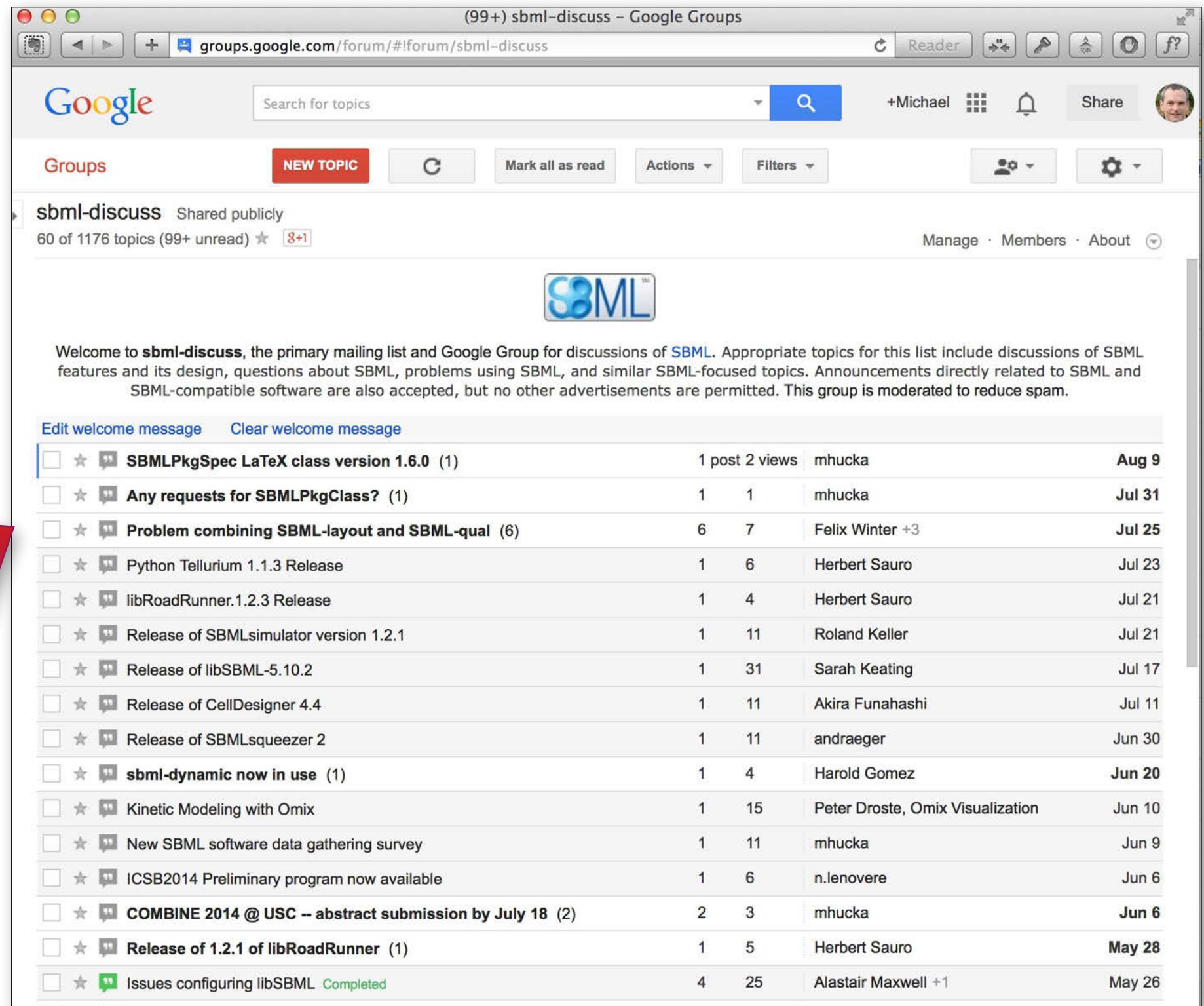- ☑ Perform static analysis of whether the model is **overdetermined**
- ☑ Perform additional checks for recommended good **modeling practices**
- ☑ Perform all other general SBML consistency checks (**SBML L2V4 rules 2nnnn; highly recommended**)
- ☐ Ignore SBML Level 3 packages that are not officially finalized

This validator supports all specifications of SBML through **Level 3 Version 1 Core** (Release 1), as well as all **Level 3 packages** for which specifications exist. **Limitations**: (1) For SBML Level 3 packages that have not been finalized, validation is limited to XML syntax *only*; consequently, a model may have other errors in its use of Level 3 package constructs yet still pass this validator. (2) SBML `<annotation>` element content is always only checked for proper XML syntax; the *content* of annotations is *not* validated; consequently, a model may have errors in its annotations yet still pass

http://sbml.org/Facilities/Validator

MOCCASIN is a MATLAB converter

Model ODE Converter for Creating Automated SBML INteroperability

Converts certain basic forms of MATLAB ODE models into SBML form

Written in Python – does **not** need or rely on MATLAB

- Parses MATLAB into abstract syntax tree using custom parser

- Uses heuristics to resolve ambiguous constructs (arrays vs functions)

- **Infers reactions** from ODEs using algorithm by Fages, Gay & Soliman

  - "Inferring reaction systems from ordinary differential equations", *Theoretical Computer Science*, 599:64–78, 2015

  - Uses BIOCHAM web service

Developed by Michael Hucka, Sarah Keating, Harold Gómez

Simple example

```matlab
% Various parameter settings.  The specifics here are unimportant; this
% is just an example of a real input file.
%
tspan  = [0 300];
xinit  = [0; 0];
a      = 0.01 * 60;
b      = 0.0058 * 60;
c      = 0.006 * 60;
d      = 0.000192 * 60;


% A call to a MATLAB ODE solver
%
[t, x] = ode45(@f, tspan, xinit);


% A function that defines the ODEs of the model.
%
function dx = f(t, x)
  dx = [a - b * x(1); c * x(1) - d * x(2)];
end
```

Simple example

```matlab
% Various parameter settings.  The specifics here are unimportant; this
% is just an example of a real input file.
%
tspan  = [0 300];
xinit  = [0; 0];
a      = 0.01 * 60;
b      = 0.0058 * 60;
c      = 0.006 * 60;
d      = 0.000192 * 60;


% A call to a MATLAB ODE solver
%
[t, x] = ode45(@f, tspan, xinit);


% A function that defines the ODEs of the model.
%
function dx = f(t, x)
  dx = [a - b * x(1); c * x(1) - d * x(2)];
end
```

Currently ignored

Simple example

```
% Various parameter settings.  The specifics here are unimportant; this
% is just an example of a real input file.
%
tspan  = [0 300];
xinit  = [0; 0];
a      = 0.01 * 60;
b      = 0.0058 * 60;
c      = 0.006 * 60;
d      = 0.000192 * 60;


% A call to a MATLAB ODE solver
%
[t, x] = ode45(@f, tspan, xinit);


% A function that defines the ODEs of the model.
%
function dx = f(t, x)
  dx = [a - b * x(1); c * x(1) - d * x(2)];
end
```

Currently ignored

Used for initial values

Simple example

```matlab
% Various parameter settings.  The specifics here are unimportant; this
% is just an example of a real input file.
%
tspan  = [0 300];
xinit  = [0; 0];
a      = 0.01 * 60;
b      = 0.0058 * 60;
c      = 0.006 * 60;
d      = 0.000192 * 60;


% A call to a MATLAB ODE solver
%
[t, x] = ode45(@f, tspan, xinit);


% A function that defines the ODEs of the model.
%
function dx = f(t, x)
  dx = [a - b * x(1); c * x(1) - d * x(2)];
end
```

Currently ignored

Used for initial values

SBML parameters & initial assignments

Simple example

```
% Various parameter settings.  The specifics here are unimportant; this
% is just an example of a real input file.
%
tspan  = [0 300];
xinit  = [0; 0];
a      = 0.01 * 60;
b      = 0.0058 * 60;
c      = 0.006 * 60;
d      = 0.000192 * 60;

% A call to
%
[t, x] = ode45(@f, tspan, xinit);

% A function that defines the ODEs of the model.
%
function dx = f(t, x)
  dx = [a - b * x(1); c * x(1) - d * x(2)];
end
```

Currently ignored

Used for initial values

SBML parameters & initial assignments

Used to name the SBML species

Simple example

```
% Various parameter settings.  The specifics here are unimportant; this
% is just an example of a real input file.
%
tspan  = [0 300];
xinit  = [0; 0];
a      = 0.01 * 60;
b      = 0.0058 * 60;
c      = 0.006 * 60;
d      = 0.000192 * 60;

% A call to
%
[t, x] = ode45(@f, tspan, xinit);

% A function that defines the ODEs of the model.
%
function dx = f(t, x)
  dx = [a - b * x(1); c * x(1) - d * x(2)];
end
```

Currently ignored

Used for initial values

SBML parameters & initial assignments

Used to name the SBML species

ODEs for the species – used to infer reaction equations

Upcoming for MOCCASIN

Provide way for users to identify—

- Compartments

- Preferred names for species/parameters/compartments

- Units

Interpret and support more MATLAB constructs

- E.g.: if/then/else, load, switch, reshape, linspace, etc.

- Deconstruct arrays so can convert to core SBML

  - Idea: translate to Numpy, then use Numpy functions to simplify

Support models split across multiple files

Implement the Fages et al. algorithm in MOCCASIN itself

http://sbml.org/Software/MOCCASIN

Discussion groups, Twitter feed, G+, RSS – see SBML.org

Discussion groups, Twitter feed, G+, RSS – see SBML.org

Discussion groups, Twitter feed, G+, RSS – see SBML.org

Discussion groups, Twitter feed, G+, RSS – see SBML.org

Outline

- SBML (Systems Biology Markup Language)
- A sample of SBML software resources
- **A sample of related resources and efforts**
- COMBINE
- Acknowledgments

BioModels Database

BioModels Database

Stores & serves quantitative models of biological interest

- **Free**, public resource

- Models must be described in **peer-reviewed publication**(s)

Hundreds of models are **curated by hand**

Imports & exports models in several formats
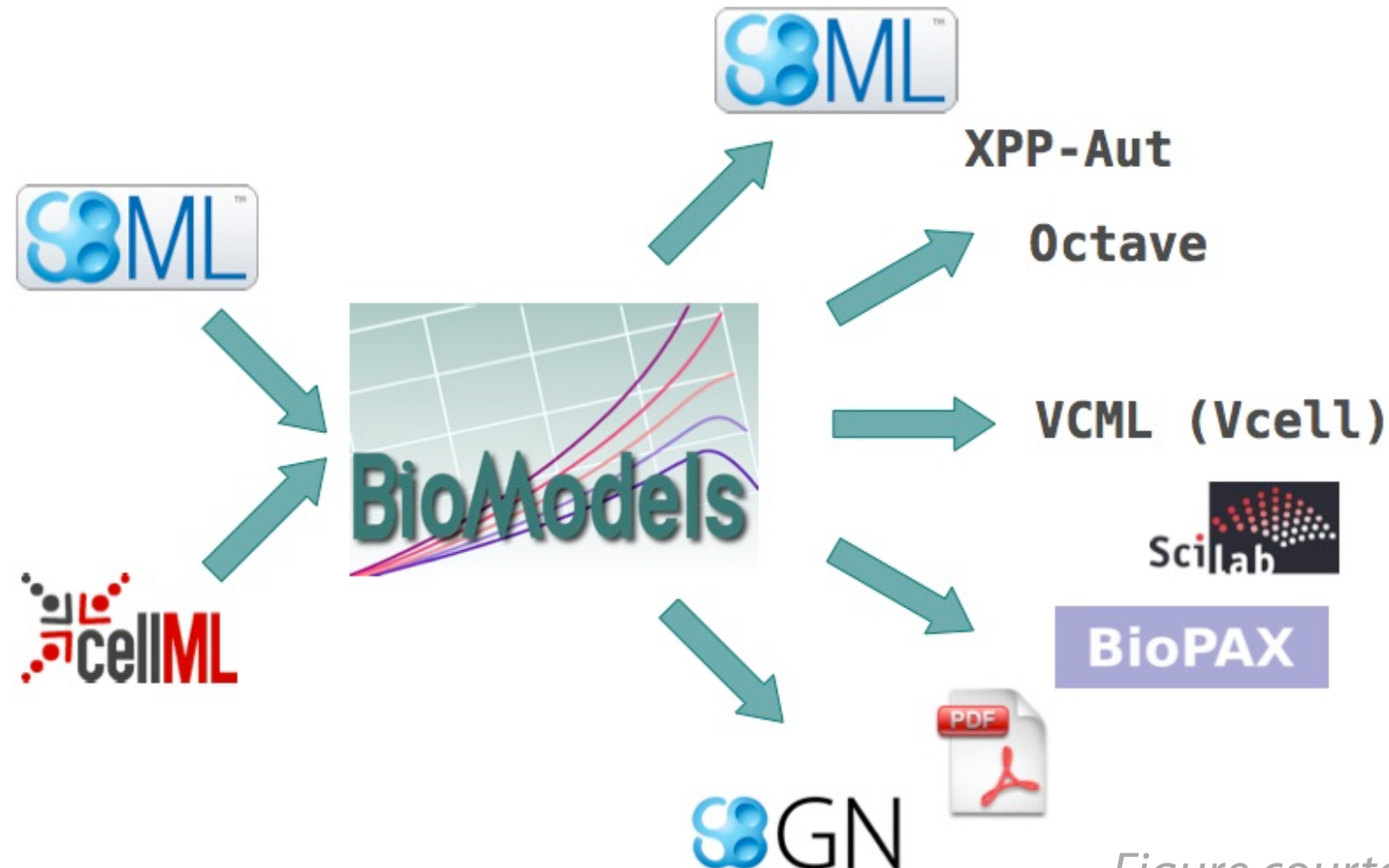
Figure courtesy of Camille Laibe

BioModels Database models are annotated by humans

Visit http://biomodels.net and submit your models!

SED-ML

Decroly & Goldbeter, PNAS, 1982

```
<sbml xmlns="http://www.sbml.org/sbml/level2/version4" xmlns:rdf="http://www.w3.org/
biomodels.net/biology-qualifiers/" xmlns:bqmodel="http://biomodels.net/model-qualific
  <model metaid="metaid_decroly82" id="decroly82" name="Decroly1982_Enzymatic_Oscilla
  <listOfUnitDefinitions>
    ...
  </listOfUnitDefinitions>
  <listOfCompartments>
    <compartment metaid="meta_cell" id="cell" name="cell" size="1">
    </compartment>
  </listOfCompartments>
  <listOfSpecies>
    <species metaid="_462445" id="alpha" name="alpha" compartment="cell" initialConc
    <species metaid="_462448" id="beta" name="beta" compartment="cell" initialConcen
    <species metaid="_462451" id="gamma" name="gamma" compartment="cell" initialConc
  </listOfSpecies>
  <listOfReactions>
    <reaction metaid="_462452" id="r1" reversible="false" sboTerm="SBO:0000176">
      <listOfProducts>
        <speciesReference species="alpha"/>
      </listOfProducts>
      <kineticLaw>
        <math xmlns="http://www.w3.org/1998/Math/MathML">
          <ci> v_Km1 </ci>
        </math>
        <listOfParameters>
          <parameter metaid="_462462" id="v_Km1" value="0.45" units="per_sec" sboTer
        </listOfParameters>
      </kineticLaw>
    </reaction>
    <reaction metaid="_462455" id="r2" reversible="false" sboTerm="SBO:0000176">
      <listOfReactants>
        <speciesReference species="alpha"/>
      </listOfReactants>
      <listOfProducts>
        <speciesReference species="beta" stoichiometry="50"/>
      </listOfProducts>
      <kineticLaw>
        <math xmlns="http://www.w3.org/1998/Math/MathML">
          <apply>
```
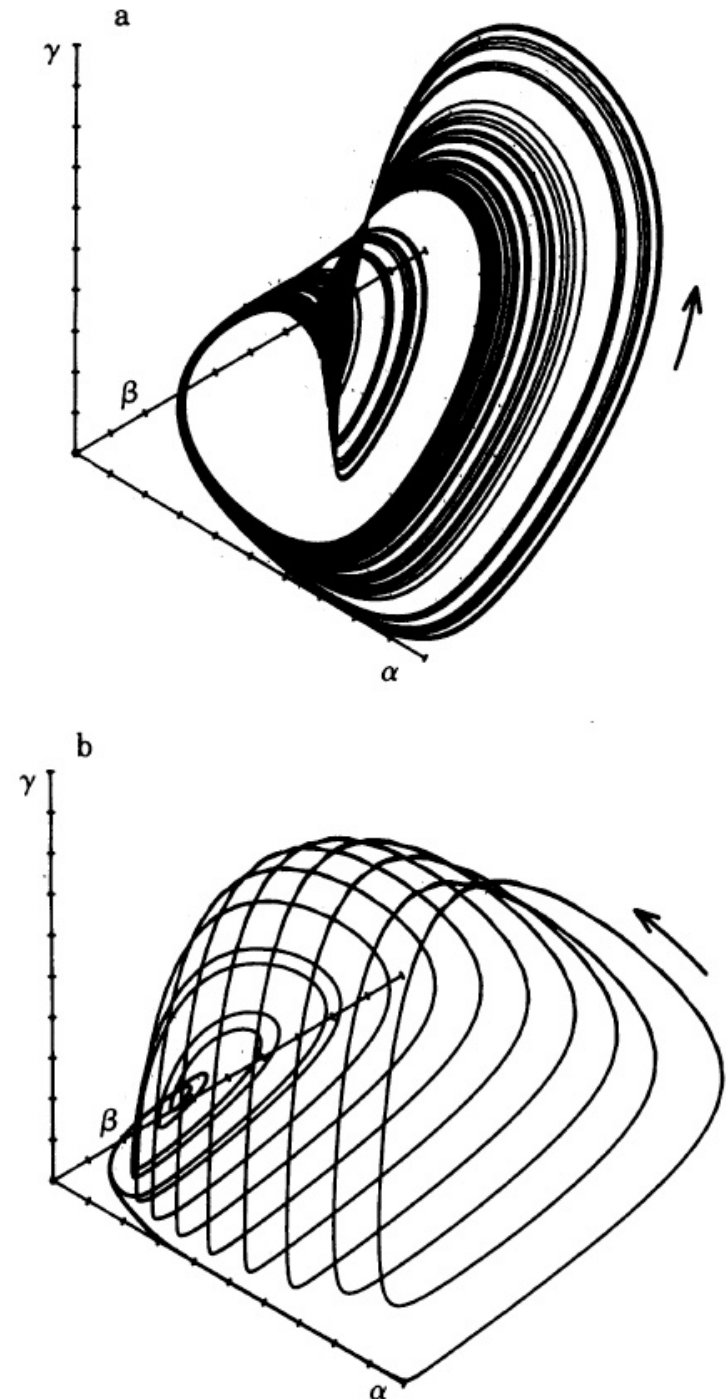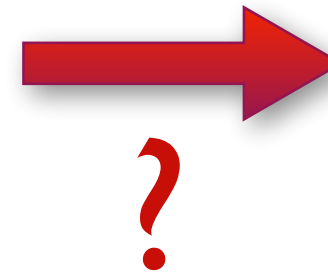
BIOMD0000000319 in BioModels Database

?

FIG. 4. Trajectories in the phase space (α, β, γ) associated with chaos (a) and with complex periodic behavior (b). The curves correspond to the substrate evolution depicted in Fig. 2 c and d, respectively, and have been obtained by integration of the kinetic equations from $t = 0$–5,000 sec. The ranges of variation of α, β, and γ in a are $\alpha = 28.44$–50.6, $\beta = 50.05$–351.1, and $\gamma = 0.05$–2.28 and in b are $\alpha = 28.18$–190.5, $\beta = 0.14$–604.0, and $\gamma = 0.00014$–8.8.

# Another problem: software can't read figure legends

SED-ML = Simulation Experiment Description ML

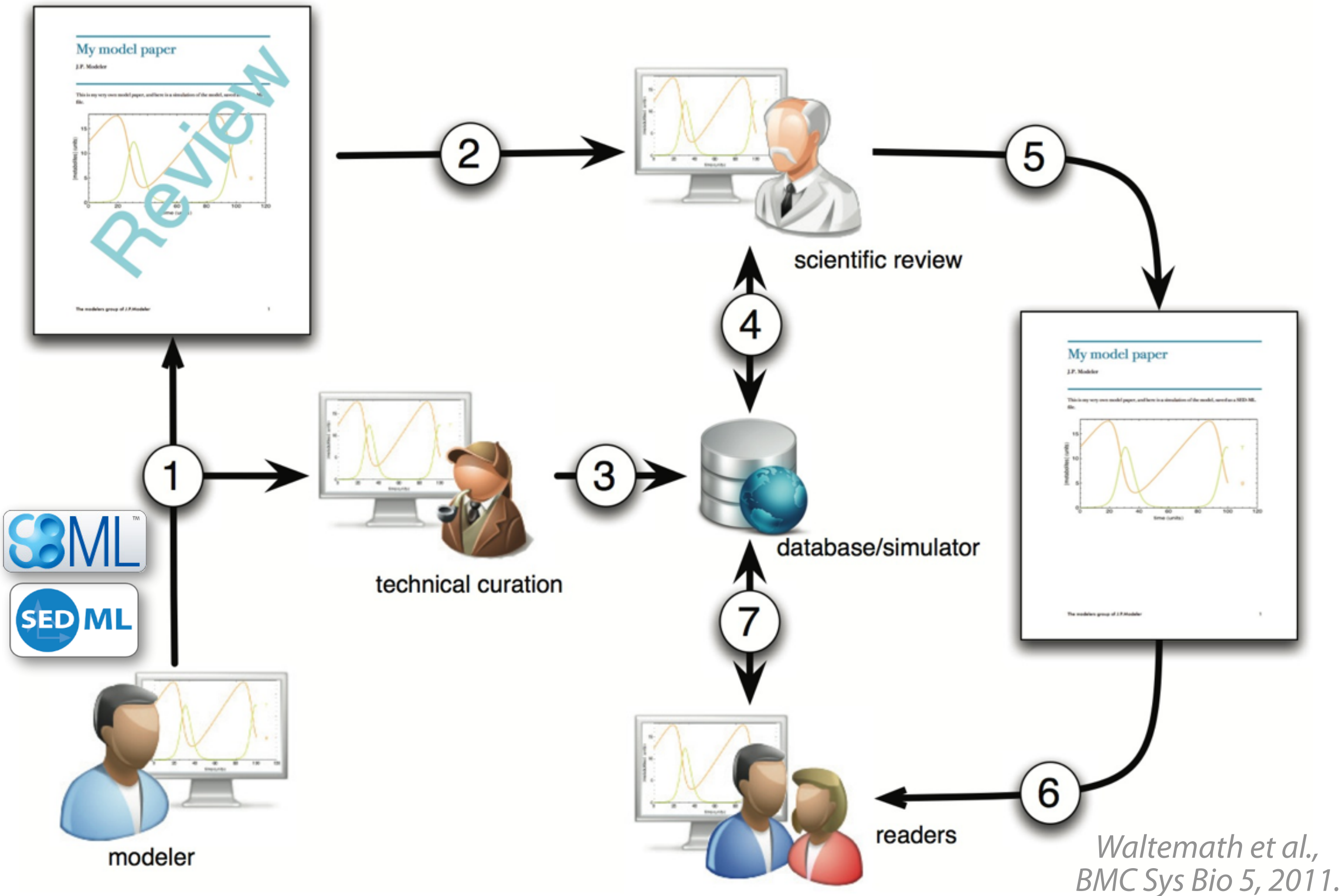**Application-independent** format

- Captures procedures, algorithms, parameter values

Can be used for

- Simulation experiments encoding parametrizations & perturbations

- Simulations using more than one model and/or method

- Data manipulations to produce plot(s)

http://sedml.org
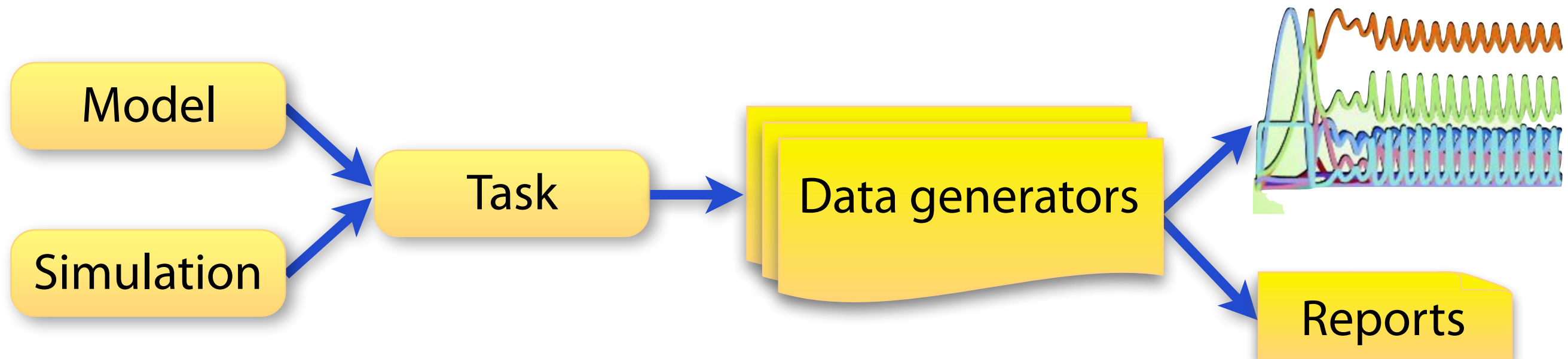
SED-ML improves reproducibility

COMBINE Archive

The problem

Multiple files usually comprise a single simulation experiment

- Model(s) file(s), possibly in multiple formats

- Simulation set-up (e.g., in SED-ML format)

- Parameter settings data files

- Diagrams (e.g., in SBGN format)

- Other files…

All the files need to be communicated together

- Opportunity to lose or mix up files during exchange & sharing

Open Modeling EXchange format (OMEX)

COMBINE Archive format =
single file that supports exchange
of all information necessary for any
modeling and simulation
experiment

- Not SBML-specific at all

- Not programming-language
  specific

- Not domain specific

OMEX = file format for COMBINE Archive

- ZIP file containing manifest file (in XML form) + other files

- Use of ZIP leverages many existing programming libraries
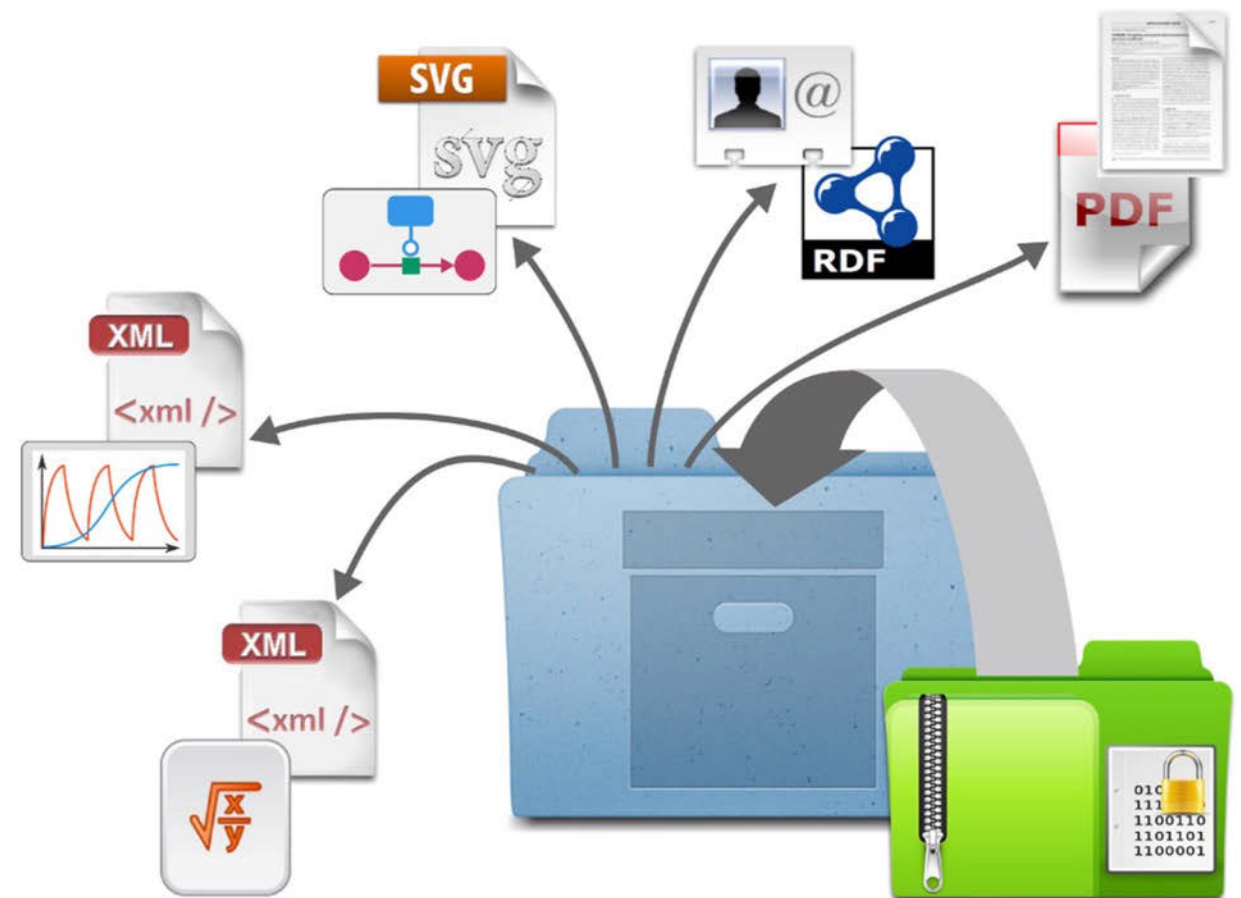
http://co.mbine.org/documents/archive

Outline

SBML (Systems Biology Markup Language)

A sample of SBML software resources

A sample of related resources and efforts

COMBINE

Acknowledgments

Motivations for the creation of COMBINE

Realizations about the state of affairs in late-2000's

- Many efforts overlapped, but lacked coordination

- Invented their own processes from scratch

- Many separate meetings meant more travel for many people

- Limited and fragile funding didn't support solid base

COMBINE = Computational Modeling in Biology Network

- Coordinate meetings

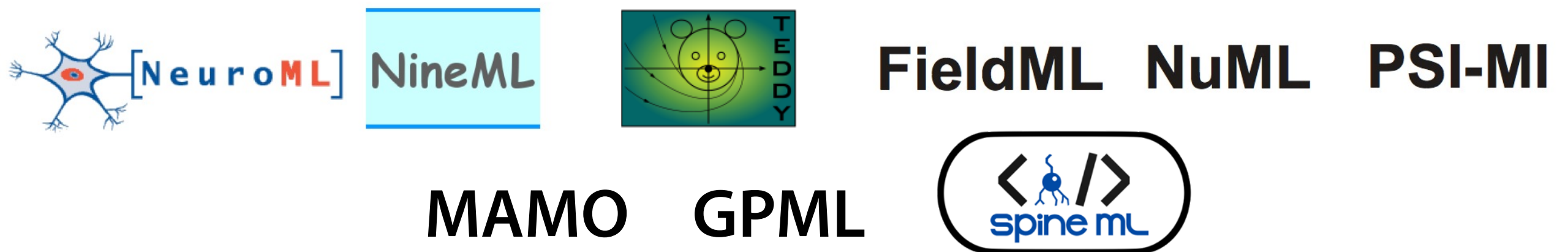- Coordinate standards development

- Develop common procedures & tools

- Provide a recognized voice

Standardization efforts represented in COMBINE today

Outline

- SBML (Systems Biology Markup Language)
- A sample of SBML software resources
- A sample of related resources and efforts
- COMBINE
- Acknowledgments

Funding and support

Attendees of COMBINE 2014, Los Angeles, California, USA

Huge thanks to everyone in the COMBINE community

Attendees of COMBINE 2015, Salt Lake City

COMBINE http://co.mbine.org

COMBINE Archive http://co.mbine.org/documents/archive

BioModels Database http://biomodels.net

SBML http://sbml.org

SED-ML http://sed-ml.org

URLs