

SBML status update

Michael Hucka, Ph.D.

*Control and Dynamical Systems
Division of Engineering and Applied Science
California Institute of Technology
Pasadena, CA, USA*



Outline

1. Status of the SBML specifications
2. Status of the SBML Level 3 packages
3. Status of the SBML Development Process
4. Status of the SBML Test Suite
5. Status of the SBML libraries

SBML = Systems Biology Markup Language

Format for representing quantitative models

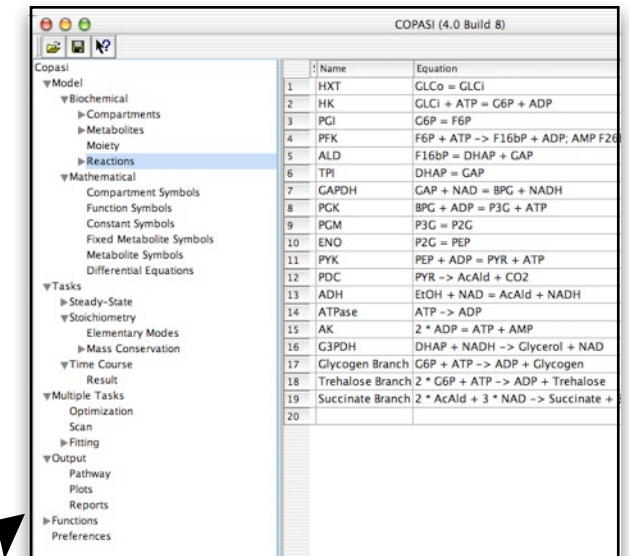
- Defines object model + rules for its use
 - Serialized to XML

Neutral with respect to modeling framework

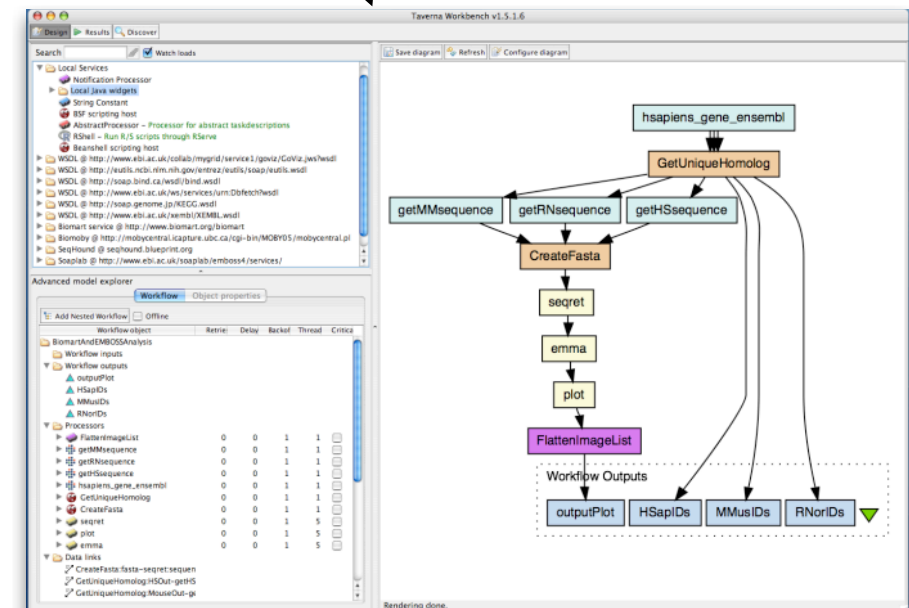
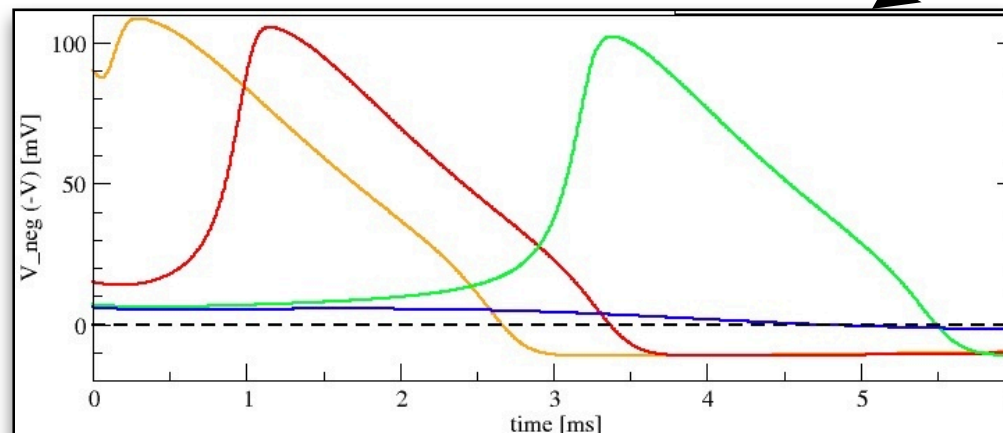
- ODE vs. stochastic vs. ...

A lingua franca for software

- Not procedural



	Name	Equation
1	HXT	GLCo = GLCI
2	HK	GLCI + ATP = G6P + ADP
3	PGI	G6P = F6P
4	PFK	F6P + ATP -> F16bP + ADP; AMP F2bP
5	ALD	F16bP = DHAP + GAP
6	TPI	DHAP = GAP
7	GAPDH	GAP + NAD = BPG + NADH
8	PKG	BPG + ADP = P3G + ATP
9	PGM	P3G = P2G
10	ENO	P2G = PEP
11	PYK	PEP + ADP = PYR + ATP
12	PDC	PYR -> AcAld + CO2
13	ADH	EtOH + NAD = AcAld + NADH
14	ATPase	ATP -> ADP
15	AK	2 * ADP = ATP + AMP
16	G3PDH	DHAP + NADH -> Glycerol + NAD
17	Glycogen Branch	G6P + ATP -> ADP + Glycogen
18	Trehalose Branch	2 * G6P + ATP -> ADP + Trehalose
19	Succinate Branch	2 * AcAld + 3 * NAD -> Succinate +
20		



For a list of tools, see the SBML Software Guide

SBML Software Guide - SBML.org

http://sbml.org/SBML_Software_Guide

Google

1/wos 2/jrn 3/uci 4/saf 5/time 6/insta 7/exif 8/cite 9/ig money sys fly web code data shop

SBML.org The Systems Biology Markup Language

News Documents Downloads Forums Facilities Community Events About

Google Site Search...

Parent pages: [SBML.org](#)

SBML Software Guide

SBML software and projects come in many varieties. Here we summarize all SBML-compatible systems known to us. The *matrix* provides an at-a-glance summary, whereas the *summary* provides longer descriptions of each software or project grouped by themes. Please [use the survey form](#) to notify us about additions and suggestions.

Number of software packages listed in the matrix today: 213

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. For an alternate set of longer descriptions grouped into themes, please see our [SBML Software Summary](#) page.

The contents of this table should be read in the following way:

- Capitalize summarizes the features that a package provides by itself (i.e., without involving another package) for working with SBML. "Editor" = creating/editing models, "Simulator" = performing time-series simulation of model "trajectories" = analyzing models (e.g., sensitivity analysis, flux balance analysis, etc.), "Compiler" = generating a detailed model, and "Utility" = providing other utility functions (e.g., translating SBML, outputting other formats).
- Framework summarizes the modeling frameworks supported by a package, regardless of whether the package also supports simulation or analysis using those same frameworks. "ODE" = ordinary differential equations, "SDE" = stochastic differential equations, "Hill" = piecewise differential equations, "Boolean" = discrete stochastic simulation, "Event" = discrete events, "Logic" = logic (e.g., Boolean) models, and "Other" = frameworks not listed here.
- API indicates whether a package exposes an application programming interface to other software systems; entries in this column are the programming languages for which that API is provided. "No" = not available.
- Dep. indicates dependencies on other software environments. "MAT" = Mathematica, "SPICE" = SciSPICE.
- Platform indicates the operating systems for which the software runs. "L" = Linux, "M" = Macintosh, "W" = Windows, "N" = not known/does not.
- SBML indicates whether a package allows importing or exporting SBML, or both. "Read" and "Write" of SBML are not indicated because it is impossible to determine this in all cases without exhaustive research.
- Avail. indicates the availability of the software. "Open source" indicates whether the source code is offered. "Academic user" indicates whether the software is free (F) or for-fee (C) to academic users. "Commercial user" indicates the same for non-academic use.

	Capabilities	Framework	API	Dep.	Platform	SBML	Avail.									
	Editor	Simulator	Compiler	Utility	ODE	SDE	Hill	Event	Logic	Other	Read	Write	Both	Open	Acad.	Comm.
Boole																
CellDesigner																
ComBio																
MathSBML																
SBML View Editor																
SBML Toolkit																
Toolbox																

Go to the SBML Software Matrix

Simulators: Discrete stochastic

Note that several of the ODE/SDE based simulators also include some form of stochastic simulation capability, and vice versa.

- StochKit** - Rule based language for generating models. Includes ODE and SDE simulations. Open source. Download requires small based regression. Requires Perl, Optional GUI uses Qt. SBML, export. (Linux/Mac/Win)
- Copasi** - ODE & Stochastic Simulator, model builder. SBML, Import (3.1.1.2), Export (3.2). Free Non-Commercial license. Commercial license available. (Linux/Mac/Win)
- Cytosim** - Stochastic simulator based on process calculus. Free download. Java Based.
- Stasy** - Chemical kinetics simulator. SBML, Import/Export (3.1 SubSet). Includes Gillespie, Gibson-Bruck, stochastic and ODE/SDE deterministic methods. Java, open source. Free download. (Linux/Mac/Windows)
- ESS** - Exact Stochastic Simulator. Part of the (ST)ORNG bio-SPICE tool-set which includes the BioSpice model editor. Requires **BioSpice Dashboard**. Free download. Includes source code.
- StochSim2** - from the SBML Project. Source code. Requires SBML.

Simulators: Continuous deterministic

- StochKit** - Programming environment with GUI, simulator, and rule-based modeling language. SBML, import.
- StochKit** - Rule based language for generating models. Includes ODE and SDE simulations. Open source. Requires small based regression. Requires Perl, Optional GUI uses Qt. SBML, export. (Linux/Mac/Win)
- StochSim2** - Graphical model development and simulation tool for genetic regulatory networks based on Bio-Tool-Box. Free download. SBML, export.
- StochSim2** - GUI/GUI-less, parameter optimization, sensitivity analysis. Terminal based. Free Download (includes source). Requires Python, SciPy, Cython, SBML 2.3.0+, ROOT, Shroud, Optional C/C++ (required for Open Modica (required for ODEs with events), Open MPI, Scientific Python, SBML, import. (Linux/Mac)
- CellDesigner** - Graphical model development and simulation tool for all types of SBML models. Includes simulation. SBML, import & export.
- Cell Simulator** - Graphical model editor. Petri-net based algorithm. SBML, Import. Commercial. (Linux/Mac/Win)
- Cellware** - Rule Diagram model editor & simulator. SBML, Import. Free download. (Mac/Linux/Win-Port) (Linux/Mac/Win)
- Copasi** - ODE & Stochastic Simulator, model builder. SBML, Import (3.1.1.2), Export (3.2). Free Non-Commercial license. Commercial license available. (Linux/Mac/Win)
- SBMLView** - SBMLView editor and viewer. Free. Requires installation of Java. License: GPL.

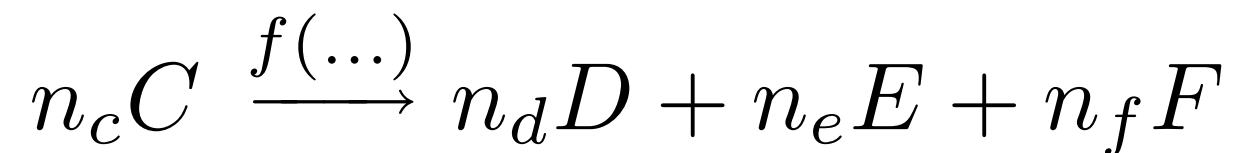
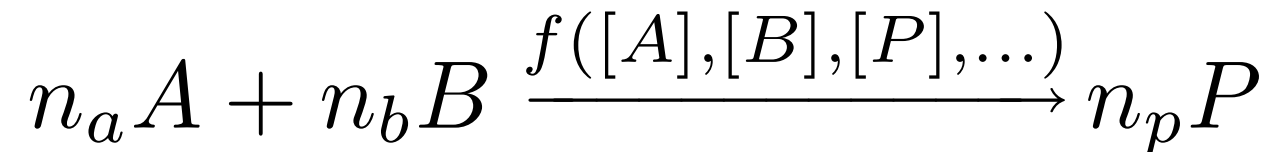
Go to the SBML Software Summary

Thursday, April 21, 2011

Basic parts of SBML

The **reaction** is central: a process occurring at a given rate

- Participants are pools of entities (**species**)



⋮

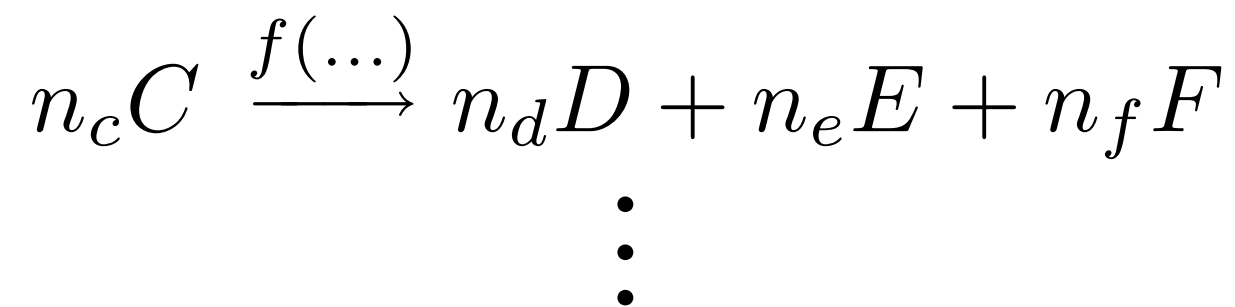
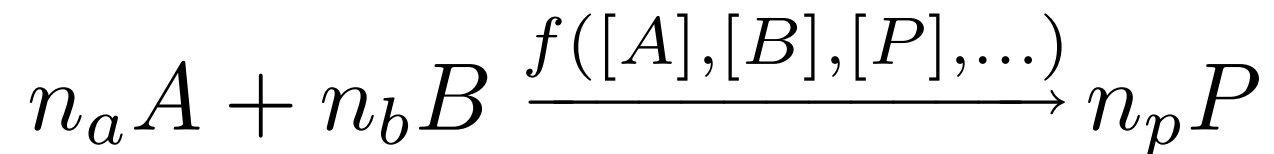
Models can further include:

- Other constants & variables
- Compartments
- Explicit math
- Discontinuous events
- Unit definitions
- Annotations

Basic parts of SBML

The **reaction** is central: a process occurring at a given rate

- Participants are pools of entities (**species**)



Can be anything
conceptually
compatible

Models can further include:

- Other constants & variables
- Compartments
- Explicit math
- Discontinuous events
- Unit definitions
- Annotations

Current SBML specifications

Current specifications:

- **Level 3 Version 1 Core**
 - 6 Oct. 2010
- **Level 2 Version 4**
 - 22 Dec. 2008
- **Level 1 Version 2**
 - 28 Aug. 2003

The screenshot shows the SBML.org website. The browser address bar displays 'http://sbml.org/Documents/Specifications'. The page header includes the SBML.org logo and the title 'The Systems Biology Markup Language'. A navigation menu contains links for News, Documents, Downloads, Forums, Facilities, Community, Events, and About. A search bar is also present. The main content area is titled 'Specifications' and contains text about SBML Level 3, including its modular nature and the fact that it does not render Level 2 obsolete. A 'Contents' sidebar lists the hierarchy of specifications: 1 SBML Level 3 (1.1 SBML Level 3 Version 1 Core, 1.2 Past releases of Level 3 Version 1 Core), 2 SBML Level 2 (2.1 SBML Level 2 Version 4, 2.2 Past Releases and Versions of SBML Level 2), and 3 SBML Level 1 (3.1 SBML Level 1 Version 2, 3.2 Past Versions of SBML Level 1). Below the text, there is a section for 'SBML Level 3' with a sub-section for 'SBML Level 3 Version 1 Core'. This section states that only Version 1 Core has been released and provides a link to the community wiki. It also mentions that the most recent release is 'Release 1'. At the bottom, there is a box titled 'The Systems Biology Markup Language (SBML): Language Specification for Level 3 Version 1 Core' which lists the authors (Michael Hucka, Frank Bergmann, Stefan Hoops, Sarah Keating, Sven Sahle, James Schaff, Lucian Smith, and Darren Wilkinson) and states that this is the final Release 1 specification of 6 Oct. 2010. It also provides a link to the document in Nature Precedings and a backup copy of the document. To the right of this box, there are links for 'Specification' (with a PDF icon), 'Errata' (with an HTML icon), 'Submit issue' (with an HTML icon), and 'Schemas' (with an XML icon).

On Nature Precedings & sbml.org/Documents

Evolution of features took time & practical experience

Level 1	Level 2	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

Cannot keep growing the feature set indefinitely

SBML Level 3 is modular:

- “Core” defines common aspects
- “Packages” add optional features
 - Models declare which packages they use
 - Tools can tell their users which packages they support
- Analogy to XML — XML is core, then there’s XPath, MathML, etc.

Rationale: even if a tool isn’t equipped to handle a given package, probably can still interpret *some* aspects of the model

Additional expected benefit: decouple development of individual L3 pkgs

Level 3 package	Active?	libSBML 5 implementation?
Graph layout	✓	★
Groups	✓	★
Spatial	✓	★
Flux balance constraints	✓	★
Hierarchical composition	✓	★ (in progress)
Multicomponent species	✓	
Annotations	✓	
Graph rendering	✓	
Distribution & ranges	✓	
Qualitative models	✓	
Dynamic structures		
Arrays & sets		

Level 3 package	Active?	libSBML 5 implementation?
Graph layout	<div>Storing model diagrams inside SBML files</div>	
Groups	✓	★
Spatial	✓	★
Flux balance constraints	✓	★
Hierarchical composition	✓	★ (in progress)
Multicomponent species	✓	
Annotations	✓	
Graph rendering	✓	
Distribution & ranges	✓	
Qualitative models	✓	
Dynamic structures		
Arrays & sets		

Level 3 package	Active?	libSBML 5 implementation?
Graph layout	✓	★
Groups	Grouping model entities together, for conceptual and annotation purposes	
Spatial		
Flux balance constraints	✓	★
Hierarchical composition	✓	★ (in progress)
Multicomponent species	✓	
Annotations	✓	
Graph rendering	✓	
Distribution & ranges	✓	
Qualitative models	✓	
Dynamic structures		
Arrays & sets		

Level 3 package	Active?	libSBML 5 implementation?
Graph layout	✓	★
Groups	✓	★
Spatial	2-D and 3-D spatial geometries and spatial processes	
Flux balance constraints	✓	★
Hierarchical composition	✓	★ (in progress)
Multicomponent species	✓	
Annotations	✓	
Graph rendering	✓	
Distribution & ranges	✓	
Qualitative models	✓	
Dynamic structures		
Arrays & sets		

Level 3 package	Active?	libSBML 5 implementation?
Graph layout	✓	★
Groups	✓	★
Spatial	✓	★
Flux balance constraints	✓	★
Hierarchical composition	Models composed of submodels	
Multicomponent species	✓	
Annotations	✓	
Graph rendering	✓	
Distribution & ranges	✓	
Qualitative models	✓	
Dynamic structures		
Arrays & sets		

Level 3 package	Active?	libSBML 5 implementation?
Graph layout	✓	★
Groups	✓	★
Spatial	✓	★
Flux balance constraints	A.k.a. steady-state models, flux balance analysis models	
Hierarchical composition	✓	★ (in progress)
Multicomponent species	✓	
Annotations	✓	
Graph rendering	✓	
Distribution & ranges	✓	
Qualitative models	✓	
Dynamic structures		
Arrays & sets		

Level 3 package	Active?	libSBML 5 implementation?
Graph layout	✓	★
Groups	✓	★
Spatial	✓	★
Flux balance constraints	✓	★
Hierarchical composition	✓	★ (in progress)
Multicomponent species	✓	
Annotations	✓	
Graph rendering	✓	
Distribution & ranges	✓	
Qualitative models	✓	
Dynamic structures		
Arrays & sets		

The excitement is palpable

Overheard at this meeting:

- “There’s been a massive appearance of packages overnight!”
- “This package stuff is really easy. It’s almost as if you planned it.”

SBML Development Process Progress

The screenshot shows a web browser window with the address bar displaying http://sbml.org/Documents/SBML_Development_Process. The browser tabs include "Process/SBML Development Proc..." and "Development Process - SBML.org". The website header features the SBML.org logo and the title "The Systems Biology Markup Language". A navigation menu includes links for News, Documents, Downloads, Forums, Facilities, Community, Events, and About, along with a Google Site Search bar. The main content area is titled "SBML Development Process" and includes a parent page link to "SBML.org / Documents". The text describes the evolution of SBML's development process, noting its participative, community-oriented nature and the transition to a more formal organization starting in 2003. A "Contents [hide]" section lists the following topics:

- 1 Goals and Motivations for SBML
- 2 Goals of the SBML Development Process
- 3 SBML Community Organization
 - 3.1 The SBML Forum
 - 3.1.1 Requirements for membership
 - 3.1.2 Conduct of meetings
 - 3.2 The SBML Editors
 - 3.2.1 Responsibilities of SBML Editors
 - 3.2.2 Terms for SBML Editors
 - 3.2.3 Election process for SBML Editors
 - 3.2.4 Selection of the Chair of the SBML Editors
 - 3.3 The SBML Team
- 4 SBML Development Procedures and Guidelines
 - 4.1 Development Roadmap
 - 4.2 General Procedures and Guidelines
 - 4.2.1 Public participation
 - 4.2.2 Communication and transparency
 - 4.2.3 Achieving consensus
 - 4.2.4 Organizing the community
 - 4.3 SBML Levels, Versions, and Releases
 - 4.3.1 SBML Levels, Versions, and Releases
 - 4.3.2 Process for SBML Level 2

http://sbml.org/Documents/SBML_Development_Process

Elaboration of process for Level 3 packages

Documents/SBML Development Process/SBML Development Process for SBML Level 3 - SBML.org

http://sbml.org/Documents/SBML_Development_Process/

Google

Process/SBML Development Proc... Development Process - SBML.org

SBML.org The Systems Biology Markup Language

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

Parent pages: [SBML.org](#) / [Documents](#) / [SBML Development Process](#)

SBML Development Process for SBML Level 3

The overall SBML Development Process is detailed on a [separate page](#). The present page describes specific aspects of the process that concern the development of SBML Level 3.

SBML Level 3 is modular, in the sense of having a defined *core* set of features and optional *packages* adding features on top of the core. This modular approach means that models can declare which feature-sets they use, and likewise, software tools can declare which packages they support. It also means that the development of SBML Level 3 can proceed in a modular fashion. The development process for Level 3 is designed around this concept.

Packages take significant time and effort to develop. It would be unreasonable to require the production of a complete specification for a package before the SBML Forum is asked to vote on whether the package is even considered worthwhile and appropriate for SBML Level 3. Therefore, the development of packages is divided into two main stages:

- The **proposal development stage**
- The **specification development stage**

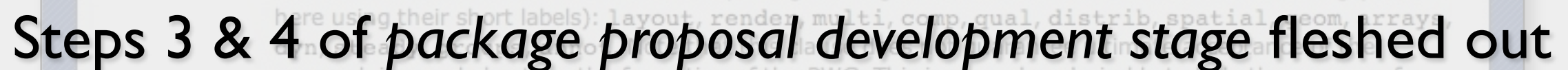
This separation means that *proposals* for packages may be produced at relatively low cost in terms of effort, time and other resources. As explained below, the *specification* stage requires more effort, including software implementations. Only after the purpose and general outline of a proposed package are accepted does the full specification need to be produced.

[\[edit\]](#) **The package proposal development stage**

At this time (December 2008), a specification for the core for SBML Level 3 is in development, and no official packages exist yet. The following process is thus only partially specified. The process is summarized in the following flowchart and explained

Contents [hide]

- 1 The package proposal development stage
- 2 The package specification development stage



Notable parts elaborated

Voting/acceptance of package **proposals** (note: not the specifications)

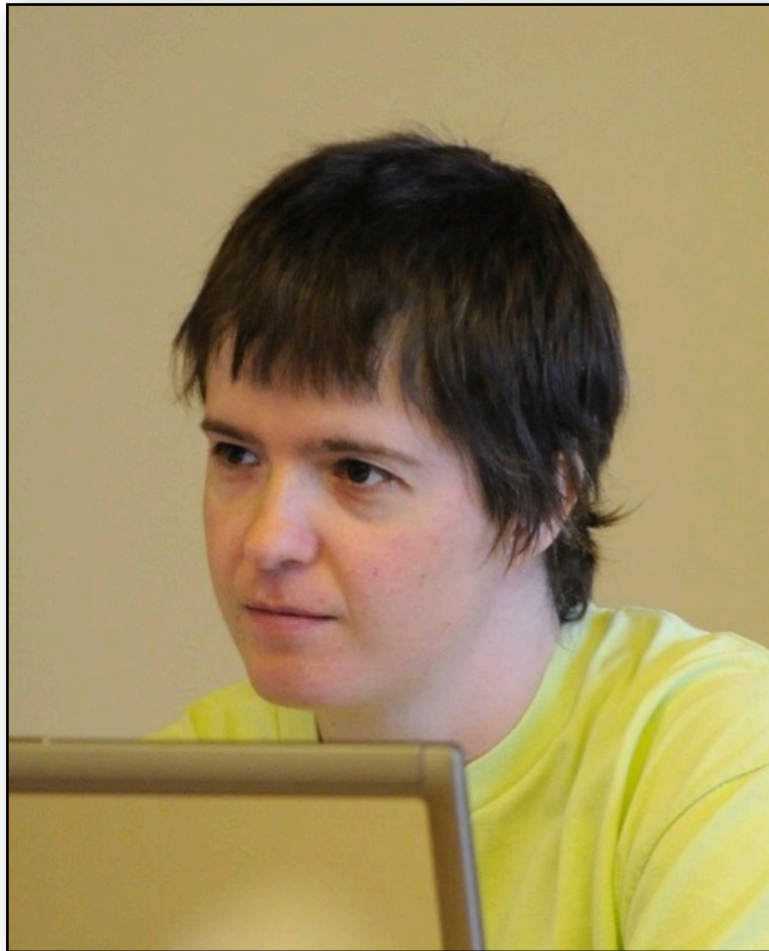
- Call for votes for existing package proposals will be issued within a matter of days

Introduction of *Package Working Groups* (PWGs)

- Group members will be
 - At least one author of the package proposal
 - At least one SBML Editor
 - Optionally, other interested persons
- Call for joining PWG will be issued for existing proposals with upcoming call for votes

New SBML Editors for 2011

Sarah Keating



Chris Myers



Status of the SBML Test Suite

Since COMBINE 2010:

- 65 new test cases
 - Includes tests for new Event features such as “persistence”
- Where possible, case results determined using analytical solutions

So far this year: informal alpha release of cases provided to some groups

- 2 groups reported tests were critical to correcting defects in their sw

Next steps:

- New test case archive + updated online system: within days of today
 - Will include SED-ML files
- Standalone test system to be rewritten after this meeting
 - Previous system to be replaced by Frank’s more feature-rich system

Status of SBML libraries

libSBML version 5 (full featured, many language interfaces, validation, etc.)

JSBML version 0.8 (lean, mean, pure Java machine)

— Watch the next two presentations —