HAL open science

SBML Level 3: an extensible format for the exchange and reuse of biological models

Sarah Keating, Frank Bergmann, Tomáš Helikar, Rahuman Malik-sheriff, Ion Moraru, Martin Meier-schellersheim, Richard Adams, Nicholas Allen, Bastian Angermann, Marco Antoniotti, et al.

To cite this version:

Sarah Keating, Frank Bergmann, Tomáš Helikar, Rahuman Malik-sheriff, Ion Moraru, et al.. SBML Level 3: an extensible format for the exchange and reuse of biological models. Molecular Systems Biology, 2020, 16 (8), pp.1-21. 10.15252/msb.20199110 . hal-02924909

HAL Id: hal-02924909

https://hal.inrae.fr/hal-02924909

Submitted on 28 Aug 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

Review

OPEN ACCESS

molecular systems biology

SBML Level 3: an extensible format for the exchange and reuse of biological models

Sarah M Keating[1,2,3,†], Dagmar Waltemath[4,†], Matthias König[5], Fengkai Zhang[6], Andreas Dräger[7,8,9], Claudine Chaouiya[10,11], Frank T Bergmann[3], Andrew Finney[12], Colin S Gillespie[13], Tomáš Helikar[14], Stefan Hoops[15], Rahuman S Malik-Sheriff[2], Stuart L Moodie[16], Ion I Moraru[17], Chris J Myers[18], Aurélien Naldi[19], Brett G Olivier[1,3,20], Sven Sahle[3], James C Schaff[21], Lucian P Smith[1,22], Maciej J Swat[23], Denis Thieffry[19], Leandro Watanabe[18], Darren J Wilkinson[13,24], Michael L Blinov[17], Kimberly Begley[25], James R Faeder[26], Harold F Gómez[27], Thomas M Hamm[7,8], Yuichiro Inagaki[28], Wolfram Liebermeister[29], Allyson L Lister[30], Daniel Lucio[31], Eric Mjolsness[32], Carole J Proctor[33], Karthik Raman[34,35,36], Nicolas Rodriguez[37], Clifford A Shaffer[38], Bruce E Shapiro[39], Joerg Stelling[40], Neil Swainston[41], Naoki Tanimura[42], John Wagner[43], Martin Meier-Schellersheim[6], Herbert M Sauro[22], Bernhard Palsson[44], Hamid Bolouri[45], Hiroaki Kitano[46,47], Akira Funahashi[48], Henning Hermjakob[2], John C Doyle[1], Michael Hucka[1,*] & SBML Level 3 Community members[‡]

Abstract

Systems biology has experienced dramatic growth in the number, size, and complexity of computational models. To reproduce simulation results and reuse models, researchers must exchange unambiguous model descriptions. We review the latest edition of the Systems Biology Markup Language (SBML), a format designed for this purpose. A community of modelers and software authors developed SBML Level 3 over the past decade. Its modular form consists of a core suited to representing reaction-based models and packages that extend the core with features suited to other model types including constraint-based models, reaction-diffusion models, logical network models, and rule-based models. The format leverages two decades of SBML and a rich software ecosystem that transformed how systems biologists build and interact with models. More recently, the rise of multiscale models of whole cells and organs, and new data sources such as single-cell measurements and live imaging, has precipitated new ways of integrating data with models. We provide our perspectives on the challenges presented by these developments and how SBML Level 3 provides the foundation needed to support this evolution.

Keywords computational modeling; file format; interoperability; reproducibility; systems biology

Subject Categories Computational Biology; Metabolism; Methods and Resources

DOI 10.15252/msb.20199110 | Received 11 July 2019 | Revised 24 June 2020 | Accepted 9 July 2020

Mol Syst Biol. (2020) 16: e9110

Introduction

Systems modeling and numerical simulations in biology can be traced to the mid-20th century. Though general theorizing about systems began earlier, the application of systems analysis to biology

1 Computing and Mathematical Sciences, California Institute of Technology, Pasadena, CA, USA
2 European Bioinformatics Institute, European Molecular Biology Laboratory (EMBL-EBI), Hinxton, UK
3 BioQuant/COS, Heidelberg University, Heidelberg, Germany
4 Medical Informatics, Institute for Community Health, University Medicine Greifswald, Greifswald, Germany
5 Institute for Theoretical Biology, Humboldt-University Berlin, Berlin, Germany
6 Laboratory of Immune System Biology, National Institute of Allergy and Infectious Diseases, National Institutes of Health, Bethesda, MD, USA
7 Computational Systems Biology of Infection and Antimicrobial-Resistant Pathogens, Institute for Biomedical Informatics (IBMI), University of Tübingen, Tübingen, Germany
8 Department of Computer Science, University of Tübingen, Tübingen, Germany
9 German Center for Infection Research (DZIF), Tübingen, Germany
10 Aix-Marseille Université, CNRS, Centrale Marseille, Marseille, France

© 2020 California Institute of Technology. Published under the terms of the CC BY 4.0 license

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

gained attention in the 1950s thanks to the work of biologists such as von Bertalanffy and Kacser (Von Bertalanffy, 1950; Kacser, 1957). The era of numerical simulation in biology truly began with the landmark works of Chance on enzyme kinetics (Chance *et al*, 1940), Hodgkin and Huxley on the molecular basis of neuronal transmission (Hodgkin & Huxley, 1952), and Turing on the chemical basis of morphogenesis (Turing, 1952). Since then, the number and variety of models have grown in all of the life sciences. As precise descriptions of phenomena that can be simulated, analyzed, and compared with experimental data, models provide unique insights that can confirm or refute hypotheses, suggest new experiments, and identify refinements to the models.

The availability of more data, more powerful modeling methods, and dramatically increased computing power led to the rise of systems biology as a compelling research theme around the turn of the millennium (Kitano, 2000; Ideker *et al*, 2001). Though computational models were at first published as printed equations in journal articles, the desire to reuse an ever-increasing number of models called for digital formats that were interoperable between software systems and could be easily exchanged between scientists (topics of interest as early as the 1960s; c.f. Garfinkel, 1969). This drove efforts to create tool-*independent* ways of representing models that could avoid the potential for human translation errors, be stored in databases, and provide a common starting point for simulations and analyses regardless of the software used (Goddard *et al*, 2001; Hucka *et al*, 2001; Lloyd *et al*, 2004). One such effort was SBML, the Systems Biology Markup Language. Its initial design was motivated by discussions to create a "metabolic model file format" following a 1999 workshop (recounted by Kell & Mendes, 2008). A distributed community thereafter discussed ideas that informed work at Caltech in late 1999/early 2000 and led (after a series of public drafts) to the specification of the official version of SBML Level 1 version 1 being released in March 2001 (Hucka *et al*, 2003).

While SBML was initially developed to exchange compartmental models of biochemical reaction networks primarily formulated in terms of chemical kinetics (Hucka *et al*, 2001), it was always understood that there existed more types of models than the initial version of SBML could represent explicitly. However, seeking community consensus on a limited set of simpler features, which could be readily implemented in software at the time, was deemed a more pragmatic strategy. A deliberate decision was taken to delay the addition of more advanced capabilities to a later time. As a result, SBML has evolved in stages in a community-driven fashion that has benefited from the efforts of many researchers worldwide over two decades. As time passed, the need to support a broader range of model types, modeling frameworks, and research areas became apparent. SBML's success in serving as an interchange format for basic types of models led communities of modelers to ask whether it could be adapted or expanded to support more types. In addition to reaction-diffusion models, alternative modeling frameworks have risen in popularity in the past decade (Machado *et al*, 2011), and researchers have faced interoperability problems

11 Instituto Gulbenkian de Ciência, Oeiras, Portugal
12 ANSYS UK Ltd, Milton Park, Oxfordshire,UK
13 School of Mathematics, Statistics and Physics, Newcastle University, Newcastle upon Tyne, UK
14 Department of Biochemistry, University of Nebraska–Lincoln,Lincoln, NE, USA
15 Biocomplexity Institute & Initiative, University of Virginia, Charlottesville, VA, USA
16 Eight Pillars Ltd, Edinburgh, UK
17 Center for Cell Analysis and Modeling, UConn Health, Farmington, CT, USA
18 Department of Electrical and Computer Engineering, University of Utah, Salt Lake City,UT, USA
19 Institut de Biologie de l'ENS (IBENS), Département de Biologie, École Normale Supérieure, CNRS, INSERM, Université PSL, Paris, France
20 SysBioLab, AIMMS, Vrije Universiteit Amsterdam, Amsterdam, the Netherlands
21 Applied BioMath, LLC, Concord, MA, USA
22 Department of Bioengineering, University of Washington, Seattle, WA, USA
23 Simcyp (a Certara company), Sheffield, South Yorkshire, UK
24 The Alan Turing Institute, British Library, London, UK
25 California Institute of Technology, Pasadena, CA, USA
26 Department of Computational and Systems Biology, University of Pittsburgh, Pittsburgh, PA, USA
27 Department of Biosystems Science and Engineering, ETH Zürich, Basel,Switzerland
28 Management & IT Consulting Division, Mizuho Information & Research Institute, Inc.,Tokyo, Japan
29 Université Paris-Saclay, INRAE, MaIAGE,Jouy-en-Josas, France
30 Oxford e-Research Centre (OeRC), Department of Engineering Science, University of Oxford, Oxford, UK
31 College of Sciences, NC State University, Raleigh,NC, USA
32 Department of Computer Science, University of California, Irvine, CA, USA
33 Institute of Cellular Medicine, Newcastle University, Newcastle upon Tyne, UK
34 Department of Biotechnology, Bhupat and Jyoti Mehta School of Biosciences, Indian Institute of Technology (IIT) Madras, Chennai, India
35 Initiative for Biological Systems Engineering (IBSE), IIT Madras, Chennai, India
36 Robert Bosch Centre for Data Science and Artificial Intelligence (RBC-DSAI), IIT Madras, Chennai, India
37 The Babraham Institute, Cambridge, UK
38 Department of Computer Science, Virginia Tech, Blacksburg, VA, USA
39 Department of Mathematics, California State University, Northridge, CA, USA
40 Department of Biosystems Science and Engineering, SIB Swiss Institute of Bioinformatics, ETH Zürich, Basel, Switzerland
41 Institute of Integrative Biology, University of Liverpool, Liverpool, UK
42 Science Solutions Division, Mizuho Information & Research Institute, Inc., Tokyo, Japan
43 IBM Research Australia, Melbourne, Vic., Australia
44 Department of Bioengineering, University of California San Diego, La Jolla, CA, USA
45 Systems Immunology, Benaroya Research Institute at Virginia Mason, Seattle, WA, USA
46 The Systems Biology Institute, Tokyo, Japan
47 Okinawa Institute of Science and Technology, Okinawa, Japan
48 Department of Biosciences and Informatics, Keio University, Yokohama, Kanagawa, Japan
*Corresponding author. Tel: +1 626 395 3418; E-mail: mhucka@caltech.edu
†These authors contributed equally to this work
‡A complete list of members and affiliations appears in the Appendix 1

 © 2020 California Institute of Technology Published under the terms of the CC BY 4.0 license

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

between software tools developed for their use. These needs drove a profound change in SBML's structure: A facility to permit layering the core of SBML with new features suited to more types of models, together with a way for individual models to identify which sets of extensions they need for proper interpretation. The release of SBML Level 3 (Hucka *et al*, 2010) has provided a new foundation to enable the exchange of a greater variety of models in various domains of biology (Fig 1).

In the rest of this article, we begin by summarizing SBML's general structure and then describe the modularity introduced in Level 3 and the wide range of modeling formalisms supported by Level 3 packages. We follow that by describing the community aspects of SBML development. We continue with a discussion of SBML's impact on both computational modeling and the modeling community, and finally, we close with a discussion of forthcoming challenges.

The structure of SBML

The core of SBML is focused on encoding models in which entities are located in containers and are acted upon by processes that modify, create, or destroy entities. The containers do not need to correspond to physical structures; they can be conceptual or abstract. Additional constructs allow parameters, initial conditions, other variables, and other mathematical relationships to be defined (Fig 2A). In the most common type of model, the "entities" are biochemical substances, the "containers" are well-mixed and spatially homogeneous, and the "processes" are biochemical reactions happening within or between the containers. This originally led to the SBML constructs being named *species*, *compartments*, and *reactions*, respectively (Fig 2B), but these names are historical artifacts and belie the generality of the underlying scheme. Software applications can map the names to other concepts to better suit their purposes. For instance, "species" could be mapped to populations of molecules, cells, or even organisms.

Modelers and software developers are encouraged to use SBML's reaction construct to define a model's behavior in preference to formulating the model explicitly as a system of equations. This gives users freedom to convert the model into the final format they prefer—a simpler operation than (for example) inferring a reaction network from a system of differential equations. More importantly, the approach also naturally handles models where reaction kinetics are unknown or unneeded, such as interaction maps, and supports the elaboration of the reaction construct using SBML packages (discussed below). That said, the use of reactions is optional, and SBML provides features sufficient for encoding a large diversity of purely mathematical models, too. Whether using reactions or not, values of model variables and their changes over time may be fixed or determined by mathematical expressions, either before or during simulation, continuously or in response to discrete events, with or without time delays. Units of measurement can be specified for all entities and values; in addition to adding a layer of essential physical knowledge (after all, how else could one interpret whether a time course is in milliseconds or years?), information about units can be used to verify the relationships expressed in a model. Units also facilitate reuse of models and components, interconnection of models, conversion of models between different frameworks, and integration of data with models.

SBML does not dictate which framework must be used to analyze or simulate a model; in fact, it purposefully lacks any explicit way to specify what is done with a model—whether to run simulations or other types of analyses, how to run them, or how to present the results—because externalizing this information enhances model reusability and permits independent innovation in separate but complementary formats. Two of the most popular methods for time-course simulation are commonly used: one is numerical integration of differential equations created from the reactions and other relationships affecting model variables, and the other is simulating the time evolution of the model as a stochastic system via algorithms such as the one developed by Gillespie (1977). Alternative approaches are also in use, particularly when a model is enhanced with SBML packages.

Any element of an SBML model can be elaborated using machine-readable metadata as well as human-readable notes. For metadata, two schemes are supported. The first is direct labeling of SBML elements with terms from the Systems Biology Ontology (SBO; Courtot *et al*, 2011), which allows the mathematical semantics of every element of a model to be precisely specified. The second scheme uses semantic web technologies and provides greater flexibility to capture additional metadata. For instance, a molecular species in a model can be linked to a UniProt entry (The UniProt Consortium, 2017) if it represents a protein, or to ChEBI entry (Hastings *et al*, 2013) if it represents a simple chemical. Gene Ontology terms (GO; Ashburner *et al*, 2000) can be attached to species, compartments, and mathematical elements representing biological processes and functions. Simple provenance data such as identities of creators can be added to facilitate attribution and versioning. To help standardize how annotations are stored, SBML encourages the use of guidelines and resources established for this purpose (Le Novère *et al*, 2005). Finally, software tools can also use annotations to encode tool-specific data in their own formats, thus providing a way to capture data that might otherwise be lost. Annotations thereby help enrich the meaning of model components, facilitate the understanding and reuse of models, and help software work with SBML more flexibly (Neal *et al*, 2019).

The core features described above have been a backbone of SBML ever since Level 2, even as SBML continued to evolve. The development of the modular Level 3, discussed in the next section, provided an opportunity to rethink and redesign a few other rarely used features. For example, the species *charge* attribute, designed to represent molecular charge, was removed in Level 3 in favor of letting an SBML package introduce more complete support for the relevant concepts.

SBML Level 3's modularity and breadth

Constant evolution in scientific methods presents challenges for the creation of software tools and standards. One challenge arises because the creation of new standards requires labor, testing, and time. This often causes standardization efforts to lag behind the latest technical developments in a constantly moving field. A second challenge is that users want support for new methods and standards in software tools, which pressures developers to implement support

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

© 2020 California Institute of Technology Published under the terms of the CC BY 4

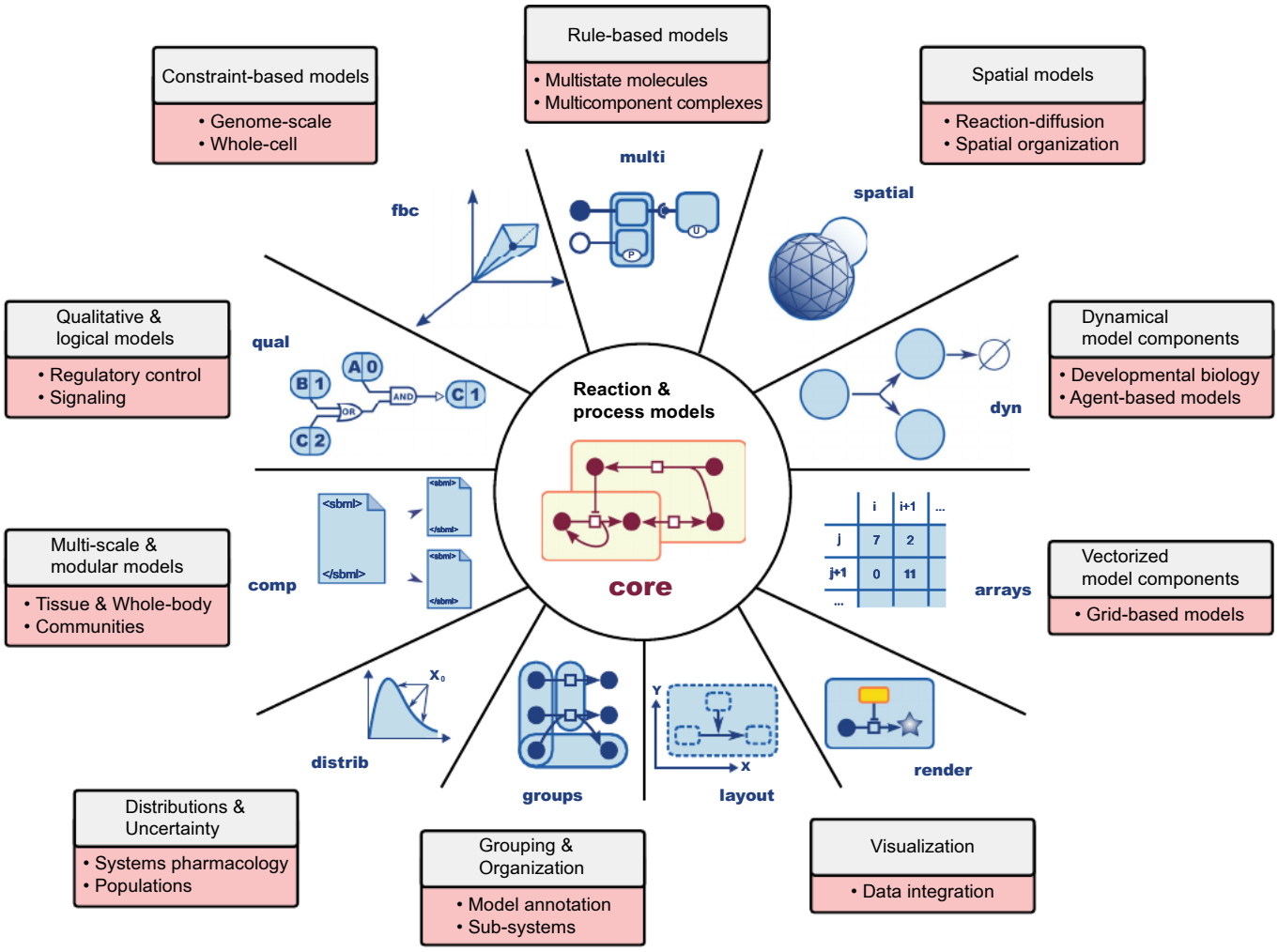

Figure 1. SBML Level 3 (Hucka *et al*, 2019) consists of a core (center) and specialized SBML Level 3 *packages* (in blue), which provide syntactical constructs to support additional modeling approaches.

The packages support new types of modeling (in the gray boxes) needed for large and complex models such as those used in various domains and fields of biology (in the light red boxes). The meanings of SBML package labels such as "fbc" are given in Table 1, with additional package information in Box 1.

quickly. Combined with the first challenge, it means that sometimes problems with a standard's definition are not discovered until more developers attempt to use it in different situations, which in turn often means that revisions to a standard are needed after it is published. Finally, another challenge is that software development often takes place under resource constraints (funding and time), limiting the scope of work that software developers can undertake—including, sometimes, limiting how many features of a standard they can support in their software.

The SBML community sought to address these challenges by putting in place certain structural features in SBML's development process. The first is the notion of *Levels*. A Level in SBML is an attempt to provide a given set of features for describing models, with higher Levels providing more powerful features. For example, the ability to express discrete events was added to SBML Level 2 but does not exist in Level 1. SBML Levels are mostly upwardly compatible, in the sense that the vast majority of models encoded in Level n can be translated to Level $n + 1$. *Versions* are used to introduce refinements to a given Level to account for realizations that come from real-life use of SBML. Finally, SBML Level 3 introduced an extensible modular architecture consisting of a central set of fixed features (named *SBML Level 3 Core*), and a scheme for adding *packages* that can augment the *Core* by extending existing elements, adding new elements, and adjusting the meaning or scope of elements. A model declares which packages it uses in order to guide its interpretation by software applications. If a software tool detects the presence of packages that it does not support, it may inform users if it cannot work with the model. Together, these three features (Levels, Versions, packages) help address the challenges discussed above: they ease coping with evolution in methods by collecting significant changes into discrete stages (SBML Levels), they help deal with the inevitable need for revisions (Versions within Levels), and they allow developers to limit the feature set

 © 2020 California Institute of Technology Published under the terms of the CC BY 4.0 license

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

A

```
<?xml version="1.0" encoding="UTF-8"?>
<sbml xmlns="http://www.sbml.org/sbml/level3/version2/core" level="3" version="2"
  xmlns:fbc="http://www.sbml.org/sbml/level3/version1/fbc/version2" fbc:required="false"
  xmlns:comp="http://www.sbml.org/sbml/level3/version1/comp/version1" comp:required="true"
  xmlns:layout="http://www.sbml.org/sbml/level3/version1/layout/version1" layout:required="false" ...>
                                                                          declaration of packages
  <model id="tiny_example" substanceUnits="mmole" timeUnits="second" volumeUnits="litre" ...>

    <listOfUnitDefinitions> ... </listOfUnitDefinitions>                  units
    <listOfFunctionDefinitions> ... </listOfFunctionDefinitions>          functions
    <listOfCompartments> ... </listOfCompartments>
    <listOfSpecies> ... </listOfSpecies>                                  variables
    <listOfParameters> ... </listOfParameters>
    <listOfInitialAssignments> ... </listOfInitialAssignments>
    <listOfRules> ... </listOfRules>
    <listOfConstraints> ... </listOfConstraints>
    <listOfReactions> ... </listOfReactions>                              relationships
    <listOfEvents> ... </listOfEvents>                                    core

    <layout:listOfLayouts xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <layout:layout layout:id="layout_1" ...>
        <layout:dimensions layout:width="700" layout:height="700" .../>
        <layout:listOfCompartmentGlyphs> ... </layout:listOfCompartmentGlyphs>
        <layout:listOfSpeciesGlyphs> ... </layout:listOfSpeciesGlyphs>
        <layout:listOfReactionGlyphs> ... </layout:listOfReactionGlyphs>
        <layout:listOfTextGlyphs> ... </layout:listOfTextGlyphs>
      </layout:layout>
    </layout:listOfLayouts>                                               package

  </model>
</sbml>
```

references

B

```
<unitDefinition id="mmole">
  <listOfUnits>
    <unit kind="mole" exponent="1" scale="-3" multiplier="1"/>
  </listOfUnits>
</unitDefinition>
...
<compartment id="c" name="cell compartment" size="1e-05" units="litre" constant="true" ... />
...
<species metaid="meta_glc" id="glc" name="glucose" initialConcentration="5" sboTerm="SBO:0000247"
  compartment="c"  substanceUnits="mmole" hasOnlySubstanceUnits="false" boundaryCondition="false"
  constant="false" fbc:charge="0" fbc:chemicalFormula="C6H12O6">
  <annotation>
    ...
    <bqbiol:is>
      <rdf:li rdf:resource="http://identifiers.org/chebi/CHEBI:4167"/>
    ...
<species/>
...
<parameter id="Vmax_GK" value="1e-06" sboTerm="SBO:0000186" constant="true" units="mmole_per_s" ...>
<parameter id="Km_glc" value="0.5" sboTerm="SBO:0000027" constant="true" units="mM" ...>
...
<reaction id="GK" name="Glucokinase" reversible="false" compartment="c" sboTerm="SBO:0000176" ...>
  ...
    <speciesReference species="glc" stoichiometry="1" constant="true"/>
  ...
  <kineticLaw>
    <math xmlns="http://www.w3.org/1998/Math/MathML">
      <apply>
        <times/>
        <ci> Vmax_GK </ci>
        <apply>
          <divide/>
          <ci> glc </ci>
          <apply>
            <plus/>
            <ci> Km_glc </ci>
            <ci> glc </ci>
          </apply>
  ...
```

Figure 2.

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

© 2020 California Institute of Technology Published under the terms of the CC BY 4

Figure 2. A closer look at SBML.

(A) Fragments of the global structure of an SBML file. In this example, the use of several SBML packages is declared in the file header. Model elements in the file include the descriptions of model variables, as well as their relationships. Elements of the same type are collected into "ListOf" elements; model parameters are in the ListOfParameters element. SBML package elements can refer to elements in the SBML Core as necessary. (B) Model elements are linked through unique identifiers used in the mathematical constructs and the elements describing the reactions, the molecular species, and their localization. The full model for this example is available in BioModels Database (Malik-Sheriff *et al*, 2020) as the model with identifier MODEL1904090001.

they implement (SBML Levels on the one hand, and SBML Level 3 packages on the other).

Packages allow SBML Level 3 (Hucka *et al*, 2019) to represent many model types and characteristics in a more natural way than if they had to be shoehorned into SBML Core constructs exclusively. Twelve packages have been proposed to date (Table 1); eight have been fully developed into consensus specifications and are each used by at least two software implementations (Box 1), and another two have draft specifications in use by software tools. New packages can be developed independently, within dedicated communities, at a pace that suits them. This was the case for logical modeling with the CoLoMoTo community (Naldi *et al*, 2015), constraint-based modeling within the COBRA community (Heirendt *et al*, 2019), and rule-based modeling with a community of like-minded software creators (Faeder *et al*, 2009; Zhang *et al*, 2013; Palmisano *et al*, 2014; Boutillier *et al*, 2018).

Several benefits accrue from leveraging SBML as a starting point rather than creating a new, independent format. One is it makes clear where common features overlap. Most computational modeling frameworks in the domain of biology share some common concepts —variables that represent characteristics of different kinds of entities and processes that represent interactions between entities, containers/locations, etc.—and reusing SBML Level 3 Core constructs makes the conceptual similarities explicit. This in turn makes interpretation of models easier (no need to learn new terminology) and reuse simpler (no need to translate between independent formats). Another benefit is that the creators of the format can leverage existing features developed for SBML, such as mechanisms for annotations, rather than spend time developing new approaches to achieving the same goals in a new format. This in turn leads to another benefit: the ability to reuse at least some parts of existing software libraries developed for SBML. It also means that a software application may be able to interpret at least *some* fundamental aspects of a model even if the application is not designed to work with a particular SBML Level 3 package, by virtue of understanding SBML Core (and perhaps other packages used by the model). This improves the potential for model reuse, and benefits model creators and software developers alike. Finally, a common foundation simplifies the creation of multiframework models in which some parts of the model use one formalism and other parts use others [e.g., coupling kinetic models with flux balance analysis; Watanabe *et al*, 2018).

Though this modular approach has benefits, it is not without potential pitfalls. The main risks are fragmentation of the community, and incompatibility of packages due to complex feature dependencies. The SBML community has addressed the former by maintaining communications between package developers; the community processes have such interactions built in. As for the latter, API libraries (see Box 2) can handle *some* combinations of packages and hide some of the complexity. Still, there remain some combinations of packages that are not fully understood, and it remains for future work to define how (if ever) they can be combined for use in a single model.

SBML as a community standard

SBML's success can be attributed largely to its community-based development and its consensus-oriented approach. SBML has always been developed through engagement with its user community to achieve goals expressed by that same community. To resolve occasionally conflicting technical demands, a guiding principle has been to seek consensus between different viewpoints and the needs of different groups, to find a middle ground that would be—while perhaps not a perfect solution—an *acceptable* and *usable* solution. This attracted the researchers and software developers who constitute SBML's foremost stakeholders. By using SBML in everything from software to textbooks, they helped drive further development to face the real needs expressed by the people who have those needs. This engagement allowed faster feedback from users to developers and has helped produce a rich toolkit of software and other resources that facilitate SBML's incorporation into software (Box 2).

Over the years, the community has designed rules to organize its governance, develop and maintain the specifications, and facilitate collaboration among users. The development of SBML and its Level 3 packages is shepherded by the SBML Editors, a group of community-elected volunteers serving terms of 3 years who follow a written and public process detailed on the web portal SBML.org.[1] SBML Editors write or review SBML specification documents, organize discussions and vote on specific technical issues, and enact the decisions of the community. Major proposed changes to the specifications and packages are discussed by the community via the SBML mailing lists[2] as well as during annual face-to-face meetings.

The community currently comes together twice a year within the context of meetings organized by COMBINE the Computational Modeling in Biology Network; Hucka *et al*, 2015). *HARMONY* (the Hackathon on Resources for Modeling in Biology) is a codefest that focuses on the development of software, in particular via the development of libraries, tools, and specifications; by contrast, the *COMBINE Forum* meetings focus on the presentation of novel tools and the discussion of proposed features. In addition to these general meetings, special SBML working groups are organized as needed to drive SBML package development. COMBINE's central activity is coordinating and harmonizing standardization in computational biology, and SBML is one of its core standards. FAIRsharing, a broader community network that covers life sciences more comprehensively (Sansone *et al*, 2019), maintains interconnected and organized collections of resources in many areas, including curated links between SBML and many associated funders, databases, and standards.[3]

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

 © 2020 California Institute of Technology Published under the terms of the CC BY 4.0 license

Table 1. Summary of SBML Level 3 Package statuses. Symbols: ● = released; ○ = not released; ✓ = complete; ◷= in progress; n/a = not applicable.

Package name	Label	Purpose	Specification	libSBML Support	JSBML Support	Test Suite	Reference
• Distributions	distrib	Define statistical distributions for quantitative values.	◷	✓	✓	◷	Smith *et al* (2020)
• Flux Balance Constraints	fbc	Define constraint-based models (a.k.a. steady-state models)	✓	✓	✓	✓	Olivier and Bergmann (2018)
• Groups	groups	Collect elements together for annotation purposes. Groups have no mathematical meaning and do not affect simulations	✓	✓	✓	n/a	Hucka and Smith (2016)
• Hierarchical Model Composition	comp	Define models composed of other models. The "submodels" can be stored in the same file or as separate files.	✓	✓	✓	✓	Smith *et al* (2015)
• Layout	layout	Store positions and sizes of model components in network diagrams of SBML models. (Cf. the Rendering package.)	✓	✓	✓	n/a	Gauges *et al* (2015)
• Multistate, Multicomponent, & Multicompartment Species	multi	Define features such as states or binding sites on molecular species, optionally in combination with rule-based processes	✓	✓	✓	◷	Zhang and Meier-Schellersheim (2018)
• Qualitative Models	qual	Allow model where SBML species' values represent qualitative activity levels rather than amounts or concentrations	✓	✓	✓	◷	Chaouiya *et al* (2015)
• Rendering	render	Extend the Layout package to enable storing graphical symbols and styles, curves, colors, and gradients in network diagrams	✓	✓	✓	n/a	Bergmann *et al* (2018)
○ Arrays	arrays	Define arrays of elements, such as arrays of compartments. (Core SBML Level 3 supports only scalar values.)	◷	✓	✓	◷	
○ Dynamical Processes	dyn	Describe the creation, destruction, and movement of model elements during simulation	◷	✓	✓	◷	
○ Extended math	math	Additional constructs not included in the subset of MathML used by SBML Level 3 Core for mathematical expressions	◷	◷	◷	◷	
○ Spatial Processes	spatial	Define spatially inhomogeneous compartment geometries and processes such as diffusion	◷	✓	✓	◷	

Impact of SBML

As contributors to developments in methods, software, and standards over the past two decades (Hucka *et al*, 2015), we can attest to SBML's profound impact on the field, both from our own first-hand experiences and from surveys (Klipp *et al*, 2007) that indicate SBML has become a *de facto* standard. The impact is a result of SBML's community-oriented development approach and its design.

The SBML development process has helped shape the field partly by directly involving software developers and modelers. Frequent workshops have provided essential feedback for developers to help them better serve modelers' needs (e.g., Waltemath *et al*, 2014). Workshops as well as resources such as the SBML Software Guide (see Box 2) helped raise awareness of existing tools, which in turn increased their use and the use of SBML. This helped create a culture of sharing models and building on existing work in systems biology (Stanford *et al*, 2015). It also led to new activities centered on the models themselves, including automatic model generation, analysis of model structures, model retrieval, and integration of models with experimental data (Dräger & Palsson, 2014). SBML's successful approach to community organization has led other standardization efforts (BioPAX, NeuroML, SBGN, SED-ML) to adopt some of the same approaches; SBML was also a founding member of COMBINE (Hucka *et al*, 2015), discussed above. Some of the primary standardization efforts in COMBINE, such as BioPAX (Demir *et al*, 2010) and NeuroML (Gleeson *et al*, 2010), are more domain-specific than SBML; others, such as CellML (Lloyd *et al*, 2004), overlap SBML's primary domains but offer alternative abstractions; and finally, still others, such as SBGN (van Iersel *et al*, 2012), SBOL (Roehner *et al*, 2016), and SED-ML (Waltemath *et al*, 2011), are complementary formats.

Before the advent of SBML, it was challenging to exchange models because software tools used incompatible definition schemes. As models increased in size and complexity, manually rewriting them became more difficult, error-prone, and eventually

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

© 2020 California Institute of Technology Published under the terms of the CC BY 4

untenable. The development of SBML has enabled the use of a single model description throughout a project's life cycle even when projects involve heterogeneous software tools (Box 3). SBML-compatible software tools today allow researchers to use SBML in all aspects of a modeling project, including creation (manual or automated), annotation, comparison, merging, parametrization, simulation/analysis, results comparison, network motif discovery, system identification, omics data integration, visualization, and more. Such use of a standardized format, along with standard annotation schemes (Neal *et al*, 2019) and training in reproducible methods, improves research workflows and is generally recognized as promoting research reproducibility (Waltemath & Wolkenhauer, 2016).

The availability of a well-defined format has also facilitated the comparison of software tools to each other. Using SBML-encoded models has become the norm to assess the accuracy of modeling software: initially it is done manually using models from BioModels Database (Bergmann & Sauro, 2008), and now, it is more commonly done using the SBML Test Suite (Box 2). SBML's semantics are defined precisely enough that many simulation systems can produce equivalent results for over 1200 test cases, lending confidence that SBML-based simulations can be reproducible in different software environments.

While chemical kinetics models have been a staple of systems biology, other modeling frameworks exist. These have benefited from efforts to extend Level 3 to better suit their specific characteristics. Even when models could in principle be encoded using core SBML constructs, the use of features explicitly adapted to the needs of a domain can make model interpretation less error-prone and more natural. The former issue was demonstrated vividly when *ad hoc* methods of encoding genome-scale models led to incorrect interpretations, and a subsequent proposal to use SBML Level 3 "fbc" addressed representational inconsistencies that had hindered reproducibility (Ebrahim *et al*, 2015). The use of more domain-specific forms of encoding has been preferred by several communities, such as the qualitative and rule-based modeling communities. For example, the quickly adopted package SBML Level 3 "qual" (Chaouiya *et al*, 2015) supports software interoperability for qualitative modeling, illustrated by the use of CellNOpt (Terfve *et al*, 2012), which provides a set of optimal Boolean models that best explains the causal relationships between elements of a signal transduction network and associated data, and the subsequent use of GINsim (Chaouiya *et al*, 2012) or Cell Collective (Helikar *et al*, 2012) to assess the dynamical properties of these models. Rule-based modeling can represent models that are impossible to express as reaction networks, such as polymerization (Faeder *et al*, 2009), or simply impractical to represent due to the combinatorial number of reactions implied by the rules (Hlavacek *et al*, 2003). Storing rule definitions in SBML is now feasible with the "multi" package, allowing rule-based modeling tools such as Simmune (Zhang *et al*, 2013) and BioNetGen (Faeder *et al*, 2009) to read and write the same model definitions.

SBML has also eased the automated processing of models to the point where they have become just another type of data in the life sciences. SBML is used today as an import/export format by many databases of mathematical models (Misirli *et al*, 2014; Norsigian *et al*, 2019; Malik-Sheriff *et al*, 2020), as well as by pathway databases (Caspi *et al*, 2015; Mi *et al*, 2016; Fabregat *et al*, 2017) and reaction databases (Ganter *et al*, 2013; Wittig *et al*, 2017). SBML is the preferred format for model curation in BioModels Database (Malik-Sheriff *et al*, 2020), not only because of its popularity but also because of its provisions to precisely encode and annotate models to support reproducible modeling. SBML is also used to share models by more generic data management platforms such as SEEK (Wolstencroft *et al*, 2016) and comprehensive online simulation environments (e.g., Moraru *et al*, 2008; Weidemann *et al*, 2008; Lee *et al*, 2009; Peters *et al*, 2017). Moreover, having an agreed-upon format has facilitated the introduction of better model management strategies. This includes support for tasks such as model storage and retrieval (Henkel *et al*, 2015), version control (Scharm *et al*, 2016b), and checking quality and validity (Liebermeister, 2008; Lieven *et al*, 2020). The proliferation of derived models has led to the development of methods to compare model structure and semantic annotations (Lambusch *et al*, 2018), culminating in the development of several methods to quantify model similarities (Henkel *et al*, 2016), that can then be used to improve the relevance of model searches. Once model elements can be compared, one can align, combine, and merge different models (Krause *et al*, 2010).

A broader impact of SBML as a *de facto* standard has been the support of publishers and funding agencies. Many journals, aware of the challenges surrounding the reproducibility of scientific results, encourage authors not only to describe their models but also to make their models available in electronic form. *Molecular Systems Biology* was the first supporter of submissions in SBML format (beginning in 2005[4,5]). Today, most journals still avoid *requiring* a specific format, though some such as the BMC[6] and FEBS[7] journals do explicitly encourage authors to submit SBML files as supporting material for research where it is relevant. Others, such as Biophysical Journal (Nickerson & Hunter, 2017), recommend authors deposit models in repositories such as BioModels Database, which encourages the use of common standard formats such as SBML. Many funding agencies also now have policies related to data sharing, and some program announcements suggested the use of SBML where appropriate.[8]

Finally, the continued development of SBML has stimulated collaborative work and the creation of consortia. This has led to better awareness and communication within groups interested in specific modeling frameworks. A good example is the CoLoMoTo effort mentioned above; it was launched by researchers who needed a format to exchange qualitative models between their software tools and developed the Qualitative Modeling package for SBML (Naldi *et al*, 2015) as the solution. Nevertheless, challenges remain, as discussed in the next section. These will need to be confronted to ensure the longevity of SBML as well as continued developments.

Forthcoming challenges

For nearly two decades, SBML has supported mathematical modeling in systems biology by helping to focus the efforts of the community and foster a culture of openness and sharing. The field is evolving rapidly, which presents challenges that the community and SBML must face.

The first challenge is to remain usable in the face of relentless growth in model sizes. One of the drivers of larger size is the rising

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

 © 2020 California Institute of Technology Published under the terms of the CC BY 4.0 license

SBML Level 3 packages officially part of the standard

Distributions

The "distrib" package (Smith *et al*, 2020) provides the means to encode information about the distribution and uncertainty of numerical values assigned to a model element. Biological models often contain elements that have inexact numerical values, since they are based on values that are stochastic in nature or data that contains uncertainty; however, core SBML has no direct support for encoding values sampled from distributions. The recently-finalized "distrib" package adds constructs for sampling of random values from probability distributions and describing uncertainty statistics about element values.

Hierarchical model composition

The "comp" package (Smith *et al*, 2015) allows users to build models from other complete models or from model fragments, as a way to manage complexity and construct composite models. "Submodels" can be described within the same SBML file or linked from external files. A submodel can act as a template, and the same definition can be reused multiple times in other models to avoid duplication and enable reuse of parts. The "comp" package also enables submodels to have explicit interfaces (known as *ports*) for optional black-box encapsulation. Finally, "comp" was designed so that a hierarchical model can be converted into a single SBML model that does not use any "comp" features, making it readable by software that does not directly support the package. The library libSBML (Bornstein *et al*, 2008) provides a facility to do this.

Flux balance constraints

The "fbc" package (Olivier & Bergmann, 2018) provides a means of encoding constraint-based models and optimizations, such as is done in Flux Balance Analysis (Bordbar *et al*, 2014). Constructs in the "fbc" package allow for the definition of a list of objectives for minimization or maximization, as well as flux bounds on reactions and gene-reaction mappings. Additional information such as chemical formula and charge enable further model analyses, including calculation of reaction mass balances, electron leaks, or implausible sources of matter.

Groups

The "groups" package (Hucka & Smith, 2016) provides constructs to describe conceptual relationships between model elements. Groupings can indicate classification, partonomy, or merely a collection of things; a group's meaning can be specified using semantic annotations. Groups have no semantic meaning and cannot influence the mathematical interpretation of an SBML model.

Multistate, multicomponent, and multicompartment species

The "multi" package (Zhang and Meier-Schellersheim, 2018) manages the combinatorics produced by entities either composed of multiple components, such as molecular complexes, or that can exist in multiple states, such as proteins with post-translational modifications. With the "multi" package, rules can be defined for how reactions depend on the states of the entities and their locations. The package adds syntactic constructs for molecular species types, compartment types, features, binding sites, and bonds. Entire families of molecular complexes sharing certain properties can be defined using patterns created using these constructs.

Qualitative models

The "qual" package (Chaouiya *et al*, 2015) provides constructs to encode models whose dynamics can be represented by discrete, reachable states connected by state transitions denoting qualitative updates of model elements. Examples include logical regulatory networks (Boolean or multivalued) and Petri nets. The "qual" package introduces SBML elements to allow the definition of qualitative species, which are used to associate discrete levels of activities with entity pools, as well as transitions, which define the possible changes between states in the transition graph.

Layout and rendering

The "layout" (Gauges *et al*, 2015) and "render" (Bergmann *et al*, 2018) packages extend SBML to allow graphical representations of networks or pathways to be stored within SBML files. The "layout" package enables the encoding of positions and sizes of graphical elements such as nodes and lines, while the information about colors, fonts, etc., is defined by the "render" package. This separation presents several advantages. For example, applications can offer multiple styles for visualizing the same layout of a network map. Most of the essential aspects of a network diagram can be expressed using just the "layout" package, and thus tools do not necessarily have to implement a full graphics environment if they do not need to support customizing a diagram's look-and-feel.

popularity of genome-scale metabolic models (Bordbar *et al*, 2014), which can be produced semi-automatically (Henry *et al*, 2010). Modeling approaches have also been developed to combine the use of several such models (e.g., Bordbar *et al*, 2011). It is reasonable to expect models of ecosystems to be produced soon (microbiomes and their host). Model sizes will also increase as more models of tissues and organs are exchanged and reused, encouraged by the use of software packages that facilitate this approach, such as the open-source tools CHASTE (Mirams *et al*, 2013) and CompuCell3D (Swat *et al*, 2012). The challenge this presents is how to define, organize, and manage large models. Meeting the challenge will require a combination of novel approaches to model storage (e.g., Henkel *et al*, 2015) and comparison (e.g., Scharm *et al*, 2016a,b), as well as more effective use of SBML Level 3 features. For example, the SBML Hierarchical Model Composition ("comp") package (Smith *et al*, 2015) provides a way to encode models in SBML out of separate building blocks or from preexisting models; this can make larger models easier to structure and maintain, and it is a natural way to construct multiscale models. Similarly, the Arrays package may help to define and structure larger models by allowing models to be defined in a more compact form.

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

© 2020 California Institute of Technology Published under the terms of the CC BY 4

Software infrastructure for SBML

Application Programming Interface (API)
Open-source (LGPL) libraries and code generators help read, write, manipulate, validate, and transform SBML. They support all Levels and Versions of SBML, and all Level 3 packages

1 LibSBML (Bornstein *et al*, 2008) (http://sbml.org/Software/libSBML), written in C++, offers interfaces for C, C++, C#, Java, JavaScript, MATLAB, Octave, Perl, PHP, Python, R, and Ruby
2 JSBML (Rodriguez *et al*, 2015) (http://sbml.org/Software/JSBML) offers a pure Java API
3 Deviser (http://sbml.org/Software/Deviser) generates libSBML code for rapid package prototyping

Test Suite
The SBML Test Suite (http://sbml.org/Software/SBML_Test_Suite) helps developers implement SBML compatibility and helps users check SBML features supported in software

1 Thousands of test cases for
   - Semantic interpretation of models (for both deterministic and stochastic simulation)
   - Syntactic correctness
2 A graphical front end enables cases to be filtered by Level/Version and type of test
3 An online database allows results to be uploaded and compared with results from other simulators

Validation Facilities
Validation software can check files for compliance to the definition of SBML, good modeling practices, and consistency of units

1 API libraries include built-in validation
2 Online validator has simple user interface (http://sbml.org/Facilities/Validator)
3 Web services support software access
Validation ensures compliance with:
   - SBML syntax
   - SBML validation rules published as part of each accepted SBML specification

Conversion Facilities
Converters (http://sbml.org/Software/Converters) can translate some other formats to/from SBML

1 Conversion tools support format conversions from MATLAB, BioPAX, CellML, XPP, SBtab, and others
2 Online services such as SBFC (Rodriguez *et al*, 2016) convert uploaded files to a variety of formats
3 API libraries provide converters between different SBML Levels/Versions and different SBML constructs

Software Guide
A catalog (http://sbml.org/SBML_Software_Guide) of software applications, libraries, and online services known to support SBML—over 290 entries to date

1 A tabular interface highlights supported SBML features of each software system.
2 A list interface displays human-readable summaries of software systems.
3 Software can be added to the list upon request.

A related challenge concerns human usability of SBML and similar XML-based formats. Though SBML is intended for software, not humans, to use directly, desire for a text-based or spreadsheet-based equivalent is often voiced (e.g., Kirouac *et al*, 2019). Various answers have been developed in the form of text-based notations (e.g., Gillespie *et al*, 2006; Smith *et al*, 2009) and spreadsheet conventions (e.g., Lubitz *et al*, 2016), with bidirectional translators for SBML. These formats have undeniable appeal for many users and use cases, despite that they do not capture the entirety of SBML (often having limited or missing facilities to express units, annotations, or SBML packages). Their chief drawback is that they become error-prone to use as model size increases. Graphical user interfaces (GUIs; e.g., Funahashi *et al*, 2003; Hoops *et al*, 2006; Moraru *et al*, 2008) can overcome this; software with GUIs can help with the cognitive burden of tracking large numbers of model elements. On the other hand, GUIs can be tedious to use when entering large models, performance of some software does not scale well with increasing model sizes, and some cannot be controlled programmatically for automation purposes. A middle ground may be domain-specific modeling languages layered on top of programming languages such as Python (e.g., Lopez *et al*, 2013; Olivier *et al*, 2005. However, these tend to appeal only to users who are comfortable with (or willing to take time to learn) the programming language used as a substrate. Overall, further innovation in this area would be welcome, both to help support SBML Level 3 packages and to help users cope with ever-increasing model sizes.

Because of the diversity of biological phenomena amenable to mathematical modeling, as well as their scales and properties, it is likely that a broad variety of modeling approaches will be added to every researcher's essential toolbox (Cvijovic *et al*, 2014). Methods such as multiagent and lattice approaches are coming into wider use to represent evolving cell populations, cell migration, and deformation. Some researchers are experimenting with solutions using existing SBML packages (Watanabe & Myers, 2016; Varela *et al*, 2019). Modeling the development of tissues and organ function may also require combining these approaches with reaction-diffusion models, or multiphysics approaches (Nickerson *et al*, 2016). Population modeling will need to complement traditional instance-based systems if we want to take into account patient variability or information coming from single-cell measurements (Levin *et al*, 1997). The coupling of different approaches within the same simulation experiment is also becoming more frequent. Biomolecular reactions modeled using ODEs, Poisson processes, and Flux Balance Analyses have been coupled in the first whole-cell model (Karr *et al*, 2015). At the organ level, liver lobules have been modeled using a combination of metabolism and multiagent models (Schliess *et al*, 2014). Several approaches mixing modeling of cell mechanical properties and gene regulatory networks or signaling networks have been used

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

 © 2020 California Institute of Technology Published under the terms of the CC BY 4.0 license

Examples of SBML use cases

SBML's impact on computational systems biology includes its facilitation of collaborative work. In multiple instances, it has precipitated entirely new projects, as illustrated by the examples below.

SBML throughout the model life cycle

Encoding a model in a standard format such as SBML makes it easier to use different software tools for different purposes without format conversion, and thus makes it easier to leverage the most suitable tools at different points in a workflow. The following is an example. A signaling pathway can be designed graphically using CellDesigner (Funahashi *et al*, 2003). The resulting model can then be semi-automatically annotated using the online tool semanticSBML (Krause *et al*, 2010). Experimental kinetic information can be retrieved in SBML format from the SABIO-Reaction Kinetics database (Wittig *et al*,2017). Tools such as COPASI (Hoops *et al*, 2006) and PyBioNetFit (Mitra *et al*, 2019) provide facilities to estimate parameters and to simulate the model with various algorithms. Other SBML-enabled tools such as Tellurium (Medley *et al*, 2018) and PySCeS (Olivier *et al*, 2005) provide capabilities such as identifiability and bifurcation analysis. Each step of the process applied to a model from creation to publication of results—modeling, simulation, and analysis—can be documented using notes attached to every model element. The model can even be turned into a publishable document using SBML2LaTeX (Dräger *et al*, 2009). Finally, the model can be exported from selected modeling tools, together with data and other information all bundled together in COMBINE Archive format (Bergmann *et al*, 2014) and published in model repositories such as BioModels Database (Malik-Sheriff *et al*, 2020).

Pipeline for automated model building

Being able to describe model elements precisely using semantic annotations facilitates the creation of automated pipelines (Dräger *et al*, 2010). Such pipelines can combine existing models with databases of molecular phenotypes or reaction kinetics (Li *et al*, 2010). They can also generate models *de novo* from data resources, as has been demonstrated by the Path2Models project (Büchel *et al*, 2013). Path2Models has produced 143,000 SBML models —all fully annotated—for over 2,600 organisms, by using pathway data. Metabolic pathways were encoded in SBML Level 3 Core while signaling pathways were encoded with the SBML "qual" package (Chaouiya *et al*, 2013). Moreover, constraint-based models of genome-scale reconstruction were provided for each organism. Other pipelines have now been built, including ones that can systematically generate alternative models for different tissue types (Wang *et al*, 2012) and patient data (Uhlen *et al*, 2017), an important step toward personalized medicine.

Development, sharing, and reuse of genome-scale models of human metabolism

Constraint-based modeling approaches such as Flux Balance Analysis and its variants permit the use of whole-genome reconstructions together with experimental molecular phenotypes, in order to predict how mutations or different environments affect metabolism as well as predict drug targets and biomarkers (O'Brien *et al*, 2015). With the availability of genome-scale metabolic reconstructions, the use of metabolic flux models at the same scale has been increasing (Bordbar *et al*, 2014). A recent development in the field has been the curation of consensus metabolic models, in particular for human metabolism (Brunk *et al*, 2018). Those community efforts rely on SBML for encoding and sharing the models, including annotations, which are crucial to being able to reuse the reconstructions later, and also for visual representation using the Layout (Gauges *et al*, 2015) and Rendering (Bergmann *et al*, 2018) packages. The Flux Balance Constraint package (Olivier & Bergmann, 2018) enables encoding of the information required for model optimization and flux calculation. Unambiguous encoding in SBML has been shown to be crucial for interpreting models and precisely computing fluxes (Ebrahim *et al*, 2015; Ravikrishnan & Raman, 2015), and new validation tools for genome-scale metabolic models have been made available by the larger community (e.g., MEMOTE; Lieven *et al*, 2020).

to study morphogenesis (e.g., Tanaka *et al*, 2015). The coupling of different approaches can be done within a single hybrid model, or each model can be simulated using different software and with dynamic synchronization at run time (Mattioni & Le Novère, 2013). Once again, the SBML "comp" package can play a role in supporting these approaches, but other methods and software will be needed in the future, as well as better support for coupling models at run time using, for example, SED-ML (Waltemath *et al*, 2011).

These developments are arising in a landscape where structural models are sometimes not the central object of study, and instead function as collection of integrated information. An example of this is RECON3D, a comprehensive human metabolic network with metabolite and protein structure information (Brunk *et al*, 2018). SBML will continue to have a pivotal role here too. When SBML was introduced, the state of modeling workflows and software tools was more primitive and it was natural that a model was self-contained. SBML-encoded models often had predefined parameter values (as initial values for state variables or parameters for mathematical expressions), but today, modelers increasingly want to use the same model with different parameterizations, sometimes with parameter values expressed as distributions, lists, or ranges rather than unique values. A project may also use an ensemble of related models that differ in parameters or in turning some model elements on or off (Kuepfer *et al*, 2007). The semantic annotation of SBML elements also has become increasingly important, forming a bedrock for many of the analyses using SBML-encoded models. The growth in size and scope of annotations has recently led the modeling community to propose a standard way of storing annotations in separate linked files (Neal *et al*, 2019), relying on the COMBINE Archive format (Bergmann *et al*, 2014) to bundle everything together. Other formats that can complement SBML have been developed, and further coordination and evolution will undoubtedly happen in the future. As mentioned above, SED-ML is a format that provides a way to encode what to do with a model, which complements SBML and compensates for its lack of features to define procedures. Finally, experimentation in integrating SBML more directly with other formats and data also continues. For instance, preliminary work has shown that SBML can be enriched with SBOL (Voigt *et al*, 2018) to provide models of DNA components' behavior (Roehner & Myers, 2014), and conversely, ongoing work in supporting genome-scale models of metabolism and gene expression (known as *ME-models*, Thiele *et al*, 2012) augments SBML with SBOL to more fully capture models for use with ME-modeling software. Future developments in modeling paradigms may require

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

© 2020 California Institute of Technology Published under the terms of the CC BY 4

similar flexibility in how models are represented: some may be best served by implementing new SBML packages, others by extending existing packages, still others by combining SBML with other formats.

Besides the technical challenges, social and cultural challenges also exist for formats such as SBML. One is to continue raising awareness among researchers, software developers, and funders of the existence of SBML and related COMBINE standards. Some may not yet be using SBML simply because they are not aware of it, or its recent addition of support for many modeling formalisms (Fig 1). Raising awareness will require continual education and outreach, especially to students and early-career scientists. Awareness would be aided by greater promotion on the part of journals and reviewers of the use of SBML and related formats in paper submission guidelines. Despite some progress in this area (discussed in the previous section), the lack of stronger demands by journals and reviewers is surely one reason authors are either not aware or not motivated to publish their models in software-independent formats.

In addition, usability of standard formats depends crucially on their implementation in software tools, and motivating this work is another challenge for SBML. A pivotal factor for the success of SBML has been the extensive software ecosystem, which provides relatively easy import and export of SBML from popular software systems. However, implementing full SBML compatibility in software is not a simple matter, and problems with compatibility in the software ecosystem can be a significant source of frustration. Improving the software requires continuous investment in tool development.

That, in turn, is related to a final challenge: obtaining and maintaining funding. By virtue of not being a native format of any particular software tool, a format such as SBML may require extra work to define by consensus, and then again for developers to implement in software—and still, it will lag behind the leading edge of research because exchange formats only become important after more than one software system has something to exchange. Funders may wonder whether the resources, time and effort spent on standards development would not be better applied to other goals. However, these costs must be weighed against the costs to a whole research field of *not* having standards—and there are many such costs. To take one example, models in nonstandard formats are more difficult to review, verify, and reuse. Journal reviewers may not have access to the necessary software, or the software may not be well tested, all of which increase the chances that the published model contains errors. Researchers can spend substantial time attempting to reproduce the results, only to fail. Worse, this is a repeating cost: failures to reproduce models are rarely published or publicized, which means an untold number of researchers may spend time (and research funding) on a futile effort. Funders recognize that too many research results are irreproducible, and have urged community action (e.g., Collins & Tabak, 2014). The continued development of exchange formats, such as SBML, is a crucial and cost-effective means to enable reproducible research.

Conclusion

SBML and associated software libraries and tools have been instrumental in the growth of systems biology. As modeling and simulation grew in popularity, SBML allowed researchers to exchange and (re)use new models in an open, well-supported, interoperable format. SBML has made possible much of the research pursued by the authors of this article and also helped us to structure our thoughts about our models and the biology they represent. Today, scientists can build, manipulate, annotate, store, reuse, publish, and connect models to each other and to basic data sources. In effect, SBML has turned models into a kind of data and transformed modeling in biology from an art to an exercise in engineering.

As the field of systems biology continues to grow and address emerging challenges, SBML will grow along with it. This evolution will (as it always has) depend on close cooperation between biologists and software developers. We hope that SBML will continue to be a source of inspiration for many researchers, especially those new to the field. In return, may they help develop the next generation of SBML to support more comprehensive, richer, and more diverse models, and expand the reach of systems modeling toward entire cells, organs, and organisms.

Acknowledgements

We sincerely thank all current and past SBML users, developers, contributors, supporters, advisors, administrators, and community members. We give special thanks to the following people for contributions and support: Jim Anderson, Nadia Anwar, Gordon Ball, Duncan Bérenguier, Upinder Bhalla, Frédéric Y. Bois, Benjamin Bornstein, Richard Boys, Ann Chasson, Thomas Cokelaer, Marco Donizelli, Alexander Dörr, Marine Dumousseau (Sivade), Lisa Falk, David Fange, Ed Frank, Ralph Gauges, Martin Ginkel, Nail Gizzatkulov, Victoria Gor, Igor Goryanin, Ryan N. Gutenkunst, Arnaud Henry, Stefanie Hoffmann, Duncan Hull, Dagmar Iber, Gael Jalowicki, Henrik Johansson, Akiya Jouraku, Devesh Khandelwal, Thomas B. L. Kirkwood, Victor Kofia, Benjamin L. Kovitz, Bryan Kowal, Andreas Kremling, Ursula Kummer, Hiroyuki Kuwahara, Anuradha Lakshminarayana, Nicolas Le Novère, Thomas S. Ligon, Adrian Lopez, Timo Lubitz, Peter Lyster, Natalia Maltsev, Jakob Matthes, Joanne Matthews, Tommaso Mazza, Eric Minch, Sebastian Nagel, Maki Nakayama, Poul M. F. Nielsen, German Nudelman, Anika Oellrich, Nobuyuki Ohta, Michel Page, Victoria Petri, Ranjit Randhawa, Veerasamy Ravichandran, Elisabeth Remy, Isabel Rojas, Ursula Rost, Jan D. Rudolph, Takayuki Saito, Takeshi Sakurada, Howard Salis, Maria J. Schilstra, Marvin Schulz, Shalin Shah, Daryl Shanley, Tom Shimizu, Jacky Snoep, Hugh D. Spence, Yves Sucaet, Linda Taddeo, Jose Juan Tapia, Alex Thomas, Jannis Uhlendorf, Martijn P. van Iersel, Marc Vass, Jonathan Webb, Katja Wengler, Benjamin Wicks, Sarala Wimalaratne, Haoran Yu, Thomas J. Zajac, W. Jim Zheng, and Jason Zwolak.

The principal authors thank many funding agencies for their support of this work. F.B., A.D., M.H., T.M.H., S.M.K., B.O., and L.S., as well as SBML.org and its online resources, were supported by the National Institute of General Medical Sciences (NIGMS, US), grant R01-GM070923 (PI: Hucka). In addition, F.B. has been supported by the Bundesministerium für Bildung und Forschung (BMBF, DE), grant de.NBI ModSim1, 031L0104A (PI: Ursula Kummer). M.L.B. has been supported by NIH (US) grant P41-GM103313 and R01-GM095485. A.D. has been supported by infrastructural funding from the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation), Cluster of Excellence EXC 2124 Controlling Microbes to Fight Infections, and by the German Center for Infection Research (DZIF). A.F. was supported by the Grant-in-Aid for Young Scientists (B), grant 21700328 from JSPS KAKENHI (JP) to Keio University. J.F. was supported by National Institutes of Health (NIH, US) grant P41-GM103712 to the National Center for Multiscale Modeling of Biological Systems (MMBioS). H.H. was supported by the Biotechnology and Biological Sciences Research Council (BBSRC, UK) "MultiMod" project (grant BB/N019482/1). T.H. was supported by NIH (US) grant 5R35-GM119770-03 to the University of

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

 © 2020 California Institute of Technology Published under the terms of the CC BY 4.0 license

Nebraska–Lincoln. S.H. was supported by NIGMS (US) grant R01-GM080219. M.K. was supported by the Federal Ministry of Education and Research (BMBF, DE), research network Systems Medicine of the Liver (LiSyM), grant 031L0054, Humboldt-University Berlin (PI: König). A.L. was supported by the BBSRC (UK) while working at the Centre for Integrated Systems Biology of Ageing and Nutrition (CISBAN), Newcastle University. C.M. was supported by the National Science Foundation (NSF, USA) under grant CCF-1748200 and CCF-1856740. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF. I.M. was supported by NIH grant P41-EB023912 and P41-GM103313. K.R. was supported by the Department of Biotechnology, Government of India (grant BT/PR4949/BRB/10/1048/2012). M.M.-S. was supported by the Intramural Research Program of NIAID, NIH (US). R.M.-S. was supported by the BBSRC (UK) "MultiMod" project (grant BB/N019482/1). B.P.'s was supported by NIH (US) grant GM57089 to the University of California, San Diego, and by the Novo Nordisk Foundation Grant NNF10CC1016517. H.M.S. was supported by NIGMS (US) grant R01-GM123032 (PI: Sauro) and by the National Institute of Biomedical Imaging and Bioengineering (NIBIB, US) grant P41-EB023912 (PI: Sauro). J.C.S. was supported by NIGMS (US) grant P41-GM103313. M.S. was supported by the DDMoRe program (EU), Innovative Medicines Initiative Joint Undertaking under grant agreement 115156. N.S. was supported by BBSRC (UK) grant "Centre for Synthetic Biology of Fine and Speciality Chemicals (SYNBIOCHEM) ", grant BB/M017702/1 (PI: Nigel S. Scrutton). F.Z. was supported by the Intramural Research Program of NIAID, NIH (US).

We also thank the Google Summer of Code program[9] for support of SBML software development.

Author contributions

SMK, DW, MK, FZ, AD, CC, and MH wrote the bulk of the manuscript. Together with FTB, AF, CSG, TH, SH, RM-S, SLM, IIM, CJM, AN, BGO, SS, JCS, LPS, MJS, DT, LW, and DJW, they also wrote and/or edited specifications for SBML Level 3 Core and the Level 3 packages. MLB, KB, JRF, HFG, TMH, YI, WL, ALL, DL, EM, CJP, KR, NR, CAS, BES, JS, JCS, NS, NT, and JW contributed proposals for SBML Level 3 and/or are past or current members of the SBML Team. MM-S, HMS, BP, HB, HK, AF, HH, JCD, and MH were principal investigators (or the equivalent, depending on the institution) for grants supporting SBML development.

The SBML community members listed in the Appendix supported the development of SBML Level 3 through participation in discussions, commentary on specification documents, and/or implementation of SBML-using software.

Conflict of interest

TH has served as a shareholder and/or has consulted for Discovery Collective, Inc.

Note

1. http://sbml.org/Documents/SBML_Development_Process
2. http://sbml.org/Forums/
3. https://fairsharing.org/FAIRsharing.9qv71f
4. https://www.embo.org/news/press-releases/2005/now-live-molecular-systems-biology-a-first-in-systems-biology-publishing
5. https://www.embopress.org/page/journal/17444292/authorguide#datadeposition
6. https://www.biomedcentral.com/getpublished/writing-resources/additional-files
7. https://onlinelibrary.wiley.com/page/journal/17424658/homepage/ForAuthors.html
8. See, for example, https://grants.nih.gov/grants/guide/pa-files/par-08-023.html
9. https://summerofcode.withgoogle.com

References

Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, Davis AP, Dolinski K, Dwight SS, Eppig JT *et al* (2000) Gene Ontology: tool for the unification of biology. *Nat Genet* 25: 25 – 29

Bergmann FT, Sauro HM (2008) Comparing simulation results of SBML capable simulators. *Bioinformatics* 24: 1963 – 1965

Bergmann FT, Adams R, Moodie S, Cooper J, Glont M, Golebiewski M, Hucka M, Laibe C, Miller AK, Nickerson DP *et al* (2014) COMBINE archive and OMEX format: one file to share all information to reproduce a modeling project. *BMC Bioinformatics* 15: 369

Bergmann FT, Keating SM, Gauges R, Sahle S, Wengler K (2018) SBML level 3 package: render, version 1, release 1. *J Integr Bioinform* 15: 20170078

Bordbar A, Feist AM, Usaite-Black R, Woodcock J, Palsson BØ, Famili I (2011) A multi-tissue type genome-scale metabolic network for analysis of whole-body systems physiology. *BMC Syst Biol* 5: 180

Bordbar A, Monk JM, King ZA, Palsson BØ (2014) Constraint-based models predict metabolic and associated cellular functions. *Nat Rev Genet* 15: 107 – 120

Bornstein BJ, Keating SM, Jouraku A, Hucka M (2008) LibSBML: an API library for SBML. *Bioinformatics* 24: 880 – 881

Boutillier P, Maasha M, Li X, Medina-Abarca HF, Krivine J, Feret J, Cristescu I, Forbes AG, Fontana W (2018) The Kappa platformfor rule-based modeling. *Bioinformatics* 34: i583 – i592

Brunk E, Sahoo S, Zielinski DC, Altunkaya A, Dräger A, Mih N, Gatto F, Nilsson A, Gonzalez GAP, Aurich MK *et al* (2018) Recon3D enables a three-dimensional view of gene variation in human metabolism. *Nat Biotechnol* 36: 272â˘A ¸S281

Büchel F, Rodriguez N, Swainston N, Wrzodek C, Czauderna T, Keller R, Mittag F, Schubert M, Glont M, Golebiewski M *et al* (2013) Path2Models: large-scale generation of computational models from biochemical pathway maps. *BMC Syst Biol* 7: 116

Caspi R, Billington R, Ferrer L, Foerster H, Fulcher CA, Keseler IM, Kothari A, Krummenacker M, Laten-dresse M, Mueller LA *et al* (2015) TheMetaCyc database of metabolic pathways and enzymes and the BioCyc collection of pathway/genome databases. *Nucleic Acids Res* 44: D471 – D480

Chance B, Brainerd J, Cajori F, Millikan G (1940) The kinetics of the enzyme-substrate compound of peroxidase and their relation to the Michaelis theory. *Science* 92: 455

Chaouiya C, Naldi A, Thieffry D (2012) Logical Modelling of Gene Regulatory Networks with GINsim. In *Bacterial Molecular Networks*, van Helden J, Toussaint A, Thieffry D (eds), Vol. 804, Methods in Molecular Biology (Methods and Protocols), New York, NY: Springer.

Chaouiya C, Bérenguier D, Keating SM, Naldi A, van Iersel MP, Rodriguez N, Dräger A, Büchel F, Cokelaer T, Kowal B *et al* (2013) SBML qualitative models: a model representation format and infrastructure to foster interactions between qualitative modelling formalisms and tools. *BMC Syst Biol* 7: 135

Chaouiya C, Keating SM, Berenguier D, Naldi A, Thieffry D, Le Novère N, Iersel MPV, Helikar T (2015) SBML level 3 package: qualitative models, version 1, release 1. *J Integr Bioinform* 12: 691 – 730

Collins FS, Tabak LA (2014) Policy: NIH plans to enhance reproducibility. *Nature* 505: 612 – 613

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

© 2020 California Institute of Technology Published under the terms of the CC BY 4

Courtot M, Juty NJ, Knüpfer C, Waltemath D, Zhukova A, Dräger A, Dumontier M, Finney A, Golebiewski M, Hastings J *et al* (2011) Controlled vocabularies and semantics in systems biology. *Mol Syst Biol* 7: 1

Cvijovic M, Almquist J, Hagmar J, Hohmann S, Kaltenbach HM, Klipp E, Krantz M, Mendes P, Nelander S, Nielsen J *et al* (2014) Bridging the gaps in systems biology. *Mol Genet Genomics* 289: 727 – 734

Demir E, Cary MP, Paley S, Fukuda K, Lemer C, Vastrik I, Wu G, D'Eustachio P, Schaefer C, Luciano J *et al* (2010) The BioPAX community standard for pathway data sharing. *Nat Biotechnol* 28: 935

Dräger A, Planatscher H, Wouamba DM, Schröder A, Hucka M, Endler L, Golebiewski M, Müller W, Zell A (2009) SBML2LATEX: conversion of SBML files into human-readable reports. *Bioinformatics* 25: 1455 – 1456

Dräger A, Schröder A, Zell A (2010) Automating mathematical modeling of biochemical reaction networks. In *Systems Biology for Signaling Networks*, Choi S (ed), pp 159 – 205. New York, NY: Springer

Dräger A, Palsson BØ (2014) Improving collaboration by standardization efforts in systems biology. *Front Bioeng Biotechnol* 2: 61

Ebrahim A, Almaas E, Bauer E, Bordbar A, Burgard AP, Chang RL, Dräger A, Famili I, Feist AM, Fleming RMT *et al* (2015) Do genome-scale models need exact solvers or clearer standards? *Mol Syst Biol* 11: 831

Fabregat A, Jupe S, Matthews L, Sidiropoulos K, Gillespie M, Garapati P, Haw R, Jassal B, Korninger F, May B *et al* (2017) The Reactome pathway knowledgebase. *Nucleic Acids Res* 46: D649 – D655

Faeder JR, Blinov ML, Hlavacek WS (2009) Rule-based modeling of biochemical systems with BioNetGen. In *Methods in Molecular Biology: Systems Biology*, Maly IV (ed), pp 113 – 167. Totowa, NJ: Humana Press

Funahashi A, Tanimura N, Morohashi M, Kitano H (2003) Cell Designer: a process diagram editor for gene-regulatory and biochemical networks. *BIOSILICO* 1: 159 – 162

Ganter M, Bernard T, Moretti S, Stelling J, Pagni M (2013) MetaNetX.org: a website and repository for accessing, analysing and manipulating metabolic networks. *Bioinformatics* 29: 815 – 816

Garfinkel D (1969) Construction of biochemical computer models. *FEBS Lett* 2: S9 – S13

Gauges R, Rost U, Sahle S, Wengler K, Bergmann FT (2015) The systems biology markup language (SBML) level 3 package: layout, version 1 core. *J Integr Bioinform* 12: 267

Gillespie DT (1977) Exact stochastic simulation of coupled chemical reactions. *J Phy Chem* 81: 2340 – 2361

Gillespie CS, Wilkinson DJ, Proctor CJ, Shanley DP, Boys RJ, Kirkwood TBL (2006) Tools for the SBML community. *Bioinformatics* 22: 628 – 629

Gleeson P, Crook S, Cannon RC, Hines ML, Billings GO, Farinella M, Morse TM, Davison AP, Ray S, Bhalla US *et al* (2010) NeuroML: a language for describing data driven models of neurons and networks with a high degree of biological detail. *PLoS Comput Biol* 6: 1 – 19

Goddard NH, Hucka M, Howell F, Cornelis H, Shankar K, Beeman D (2001) Towards NeuroML: model description methods for collaborative modelling in neuroscience. *Philos Trans R Soc Lond B Biol Sci* 356: 12091228

Hastings J, deMatos P, Dekker A, Ennis M, Harsha B, Kale N, Muthukrishnan V, Owen G, Turner S, Williams M *et al* (2013) The ChEBI reference database and ontology for biologically relevant chemistry: enhancements for 2013. *Nucleic Acids Res* 41: D456 – D463

Heirendt L, Arreckx S, Pfau T, Mendoza SN, Richelle A, Heinken A, Haraldsdóttir HS, Wachowiak J, Keating SM, Vlasov V *et al* (2019) Creation and analysis of biochemical constraint-basedmodels using the COBRA toolbox vol 3.0. *Nat Protoc* 14: 639 – 702

Helikar T, Kowal B, McClenathan S, Bruckner M, Rowley T, Madrahimov A, Wicks B, Shrestha M, Limbu K, Rogers JA (2012) The cell collective: toward an open and collaborative approach to systems biology. *BMC Syst Biol* 6: 96

Henkel R, Wolkenhauer O, Waltemath D (2015) Combining computational models, semantic annotations and simulation experiments in a graph database. *Database* 2015: bau130

Henkel R, Hoehndorf R, Kacprowski T, Knüpfer C, Liebermeister W, Waltemath D (2016) Notions of similarity for systems biology models. *Brief Bioinform* 19: 77 – 88

Henry CS, DeJongh M, Best AA, Frybarger PM, Linsay B, Stevens RL (2010) High-throughput generation, optimization and analysis of genome-scale metabolic models. *Nat Biotechnol* 28: 977

Hlavacek WS, Faeder JR, Blinov ML, Perelson AS, Goldstein B (2003) The complexity of complexes in signal transduction. *Biotechnol Bioeng* 84: 783 – 794

Hodgkin AL, Huxley AF (1952) A quantitative description of membrane current and its application to conduction and excitation in nerve. *J Physiol* 117: 500 – 544

Hoops S, Sahle S, Gauges R, Lee C, Pahle J, Simus N, Singhal M, Xu L, Mendes P, Kummer U (2006) COPASI – a complex pathway simulator. *Bioinformatics* 22: 3067 – 3074

Hucka M, Sauro HM, Finney A, Bolouri H, Doyle J, Kitano H (2001) The ERATO systems biology work bench: an integrated environment for multiscale and multitheoretic simulations in systems biology. In *Foundations of Systems Biology*, Kitano H (ed), pp 125 – 143. Cambridge, MA: MIT Press

Hucka M, Finney A, Sauro HM, Bolouri H, Doyle JC, Kitano H, Arkin AP, Bornstein BJ, Bray D, Cornish-Bowden A *et al* (2003) The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics* 19: 524 – 531

Hucka M, Bergmann FT, Hoops S, Keating SM, Sahle S, Schaff JC, Smith L, Wilkinson DJ (2010) The systems biology markup language (SBML): language specification for level 3 version 1 core. *Nature Preced* https://www.nature.com/articles/npre.2010.4959.1

Hucka M, Nickerson DP, Bader GD, Bergmann FT, Cooper J, Demir E, Garny A, Golebiewski M, Myers CJ, Schreiber F *et al* (2015) Promoting coordinated development of community-based information standards for modeling in biology: the COMBINE Initiative. *Front Bioeng Biotechnol* 3: 19.

Hucka M, Smith LP (2016) SBML level 3 package: groups, version 1 release 1. *J Integr Bioinform* 13: 8 – 29

Hucka M, Bergmann FT, Chaouiya C, Dräger A, Hoops S, Keating SM, König M, Novère NL, Myers CJ, Olivier BG *et al* (2019) The systems biology markup language (SBML): language specification for level 3 version 2 core release 2. *J Integr Bioinform* 16: 20190021

Ideker T, Galitski T, Hood L (2001) A new approach to decoding life: systems biology. *Annu Rev Genomics Hum Genet* 2: 343 – 372

van Iersel MP, Villéger AC, Czauderna T, Boyd SE, Bergmann FT, Luna A, Demir E, Sorokin A, Dogrusoz U, Matsuoka Y *et al* (2012) Software support for SBGN maps: SBGN-ML and LibSBGN. *Bioinformatics* 28: 2016 – 2021

Kacser H (1957) Some physico-chemical aspects of biological organisation. In *The Strategy of the Genes: A Discussion of Some Aspects of Theoretical Biology*, Waddington CH (ed), pp 191 – 249. London: George Allen & Unwin Ltd

Karr JR, Takahashi K, Funahashi A (2015) The principles of whole-cell modeling. *Curr Opin Microbiol* 27: 18 – 24

Kell DB, Mendes P (2008) The markup is the model: reasoning about systems biology models in the Semantic Web era. *J Theor Biol* 252: 538 – 543

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

 © 2020 California Institute of Technology Published under the terms of the CC BY 4.0 license

Kirouac DC, Cicali B, Schmidt S (2019) Reproducibility of quantitative systems pharmacology models: current challenges and future opportunities. *CPT Pharmacometrics Syst Pharmacol* 8: 205 – 210

Kitano H (2000) Perspectives on systems biology. *New Generat Comput* 18: 199 – 216

Klipp E, Liebermeister W, Helbig A, Kowald A, Schaber J (2007) Systems biology standards—the community speaks. *Nat Biotechnol* 25: 390 – 391

Krause F, Uhlendorf J, Lubitz T, Schulz M, Klipp E, Liebermeister W (2010) Annotation and merging of SBML models with semantic SBML. *Bioinformatics* 26: 421 – 422

Kuepfer L, Peter M, Sauer U, Stelling J (2007) Ensemble modeling for analysis of cell signaling dynamics. *Nat Biotechnol* 25: 1001

Lambusch F, Waltemath D, Wolkenhauer O, Sandkuhl K, Rosenke C, Henkel R (2018) Identifying frequent patterns in biochemical reaction networks: a workflow. *Database* 2018

Le Novère N, Finney A, Hucka M, Bhalla US, Campagne F, Collado-Vides J, Crampin EJ, Halstead M, Klipp E, Mendes P *et al* (2005) Minimum information requested in the annotation of biochemicalmodels (MIRIAM). *Nat Biotechnol* 23: 1509 – 1515

Lee DY, Saha R, KhanYusufi FN, Park W, Karimi IA (2009) Web-based applications for building, managing and analysing kineticmodels of biological systems. *Brief Bioinform* 10: 65 – 74

Levin SA, Grenfell B, Hastings A, Perelson AS (1997) Mathematical and computational challenges in population biology and ecosystems science. *Science* 275: 334 – 343

Li P, Dada JO, Jameson D, Spasic I, Swainston N, Carroll K, Dunn W, Khan F, Malys N, Messiha HL *et al* (2010) Systematic integration of experimental data and models in systems biology. *BMC Bioinformatics* 11: 582

Liebermeister W (2008) Validity and combination of biochemical models. In *Proceedings of 3rd International ESCECWorkshop on Experimental Standard Conditions on Enzyme Characterizations*, Kettner C, Hicks MG (eds), pp 24. Frankfurt: Beilstein-Institut

Lieven C, Beber ME, Olivier BG, Bergmann FT, Ataman M, Babaei P, Bartell JA, Blank LM, Chauhan S, Correia K *et al* (2020) MEMOTE for standardized genome-scale metabolic model testing. *Nat Biotechnol* 38: 272 – 276

Lloyd CM, Halstead MDB, Nielsen PF (2004) CellML: its future, present and past. *Prog Biophys Mol Biol* 85: 433 – 450

Lopez CF, Muhlich JL, Bachman JA, Sorger PK (2013) Programming biological models in python using PySB. *Mol Syst Biol* 9: 646

Lubitz T, Hahn J, Bergmann FT, Noor E, Klipp E, Liebermeister W (2016) SBtab: a flexible table format for data exchange in systems biology. *Bioinformatics* 32: 2559 – 2561

Machado D, Costa RS, Rocha M, Ferreira EC, Tidor B, Rocha I (2011) Modeling formalisms in systems biology. *AMB Express* 1: 45

Malik-Sheriff RS, Glont M, Nguyen TVN, Tiwari K, Roberts MG, Xavier A, Vu MT, Men J, Maire M, Kananathan S *et al* (2020) BioModels—15 years of sharing computational models in life science. *Nucleic Acids Res* 48(D1): D407 – D415

Mattioni M, Le Novère N (2013) Integration of biochemical and electrical signaling-multiscale model of the medium spiny neuron of the striatum. *PLoS ONE* 8: e66811

Medley JK, Choi K, König M, Smith L, Gu S, Hellerstein J, Sealfon SC, Sauro HM (2018) Tellurium notebooks–An environment for reproducible dynamical modeling in systems biology. *PLoS Comput Biol* 14: e1006220

Mi H, Huang X, Muruganujan A, Tang H, Mills C, Kang D, Thomas PD (2016) PANTHER version 11: expanded annotation data from Gene Ontology and Reactome pathways, and data analysis tool enhancements. *Nucleic Acids Res* 45: D183 – D189

Mirams GR, Arthurs CJ, Bernabeu MO, Bordas R, Cooper J, Corrias A, Davit Y, Dunn SJ, Fletcher AG, Harvey DG *et al* (2013) Chaste: an open source C++ library for computational physiology and biology. *PLoS Comput Biol* 9: e1002970

Misirli G, Hallinan J, Wipat A (2014) Composable modular models for synthetic biology. *ACM J Emerg Technol Comput Syst* 11: 22

Mitra ED, Suderman R, Colvin J, Ionkov A, Hu A, Sauro HM, Posner RG, Hlavacek WS (2019) PyBioNetFit and the biological property specification language. *iScience* 19: 1012 – 1036

Moraru II, Morgan F, Li Y, Loew LM, Schaff JC, Lakshminarayana A, Slepchenko BM, Gao F, Blinov ML (2008) Virtual Cell modelling and simulation software environment. *IET Syst Biol* 2: 352 – 362

Naldi A, Monteiro PT, Müssel C, Consortium for Logical Models and Tools, Kestler HA, Thieffry D, Xenarios I, Saez-Rodriguez J, Helikar T, Chaouiya C (2015) Cooperative development of logical modelling standards and tools with CoLoMoTo. *Bioinformatics* 31: 1154 – 1159

Neal ML, König M, Nickerson D, Mısırlı G, Kalbasi R, Dräger A, Atalag K, Chelliah V, Cooling MT, Cook DL *et al* (2019) Harmonizing semantic annotations for computational models in biology. *Brief Bioinform* 20: 540 – 550

Nickerson D, Atalag K, de Bono B, Geiger J, Goble C, Hollmann S, Lonien J, Müller W, Regierer B, Stanford NJ *et al* (2016) The Human Physiome: how standards, software and innovative service infrastructures are providing the building blocks to make it achievable. *Interface Focus* 6: 20150103

Nickerson DP, Hunter PJ (2017) Introducing the physiome journal: improving reproducibility, reuse, and discovery of computational models. In *2017 IEEE 13th International Conference on e-Science (e-Science)*, Auckland. 2017: 448 – 449 https://doi.org/10.1109/eScience.2017.65

Norsigian CJ, Pusarla N, McConn JL, Yurkovich JT, Dräger A, Palsson BO, King Z (2019) BiGGModels 2020: multi-strain genome-scale models and expansion across the phylogenetic tree. *Nucleic Acids Res* 48(D1): D402 – D406

O'Brien EJ, Monk JM, Palsson BØ (2015) Using genome-scale models to predict biological capabilities. *Cell* 161: 971 – 987

Olivier BG, Rohwer JM, Hofmeyr JHS (2005) Modelling cellular systems with PySCeS. *Bioinformatics* 21: 560 – 561

Olivier BG, Bergmann FT (2018) SBML level 3 package: flux balance constraints version 2. *J Integr Bioinform* 15: 20170082

Palmisano A, Hoops S, Watson LT, Jones TC Jr, Tyson JJ, Shaffer CA (2014) Multistate Model Builder (MSMB): a flexible editor for compact biochemical models. *BMC Syst Biol* 8: 42

Peters M, Eicher JJ, van Niekerk DD, Waltemath D, Snoep JL (2017) The JWS online simulation database. *Bioinformatics* 33: 1589 – 1590

Ravikrishnan A, Raman K (2015) Critical assessment of genome-scale metabolic networks: the need for a unified standard. *Brief Bioinform* 16: 1057 – 1068

Roehner N, Myers CJ (2014) A methodology to annotate systems biology markup language models with the synthetic biology open language. *ACS Synth Biol* 3: 57 – 66

Rodriguez N, Thomas A, Watanabe L, Vazirabad IY, Kofia V, Gómez HF, Mittag F, Matthes J, Rudolph JD, Wrzodek F *et al* (2015) JSBML 1.0: providing a smorgasbord of options to encode systems biology models. *Bioinformatics* 31: 3383 – 3386

Roehner N, Beal J, Clancy K, Bartley B, Misirli G, Grünberg R, Oberortner E, Pocock M, Bissell M, Madsen C *et al* (2016) Sharing structure and

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

© 2020 California Institute of Technology Published under the terms of the CC BY 4

/

function in biological design with SBOL 2.0. *ACS Synth Biol* 5: 498 – 506

Rodriguez N, Pettit JB, Dalle Pezze P, Li L, Henry A, van Iersel MP, Jalowicki G, Kutmon M, Natarajan KN, Tolnay D *et al* (2016) The systems biology format converter. *BMC Bioinformatics* 17: 154

Sansone SA, McQuilton P, Rocca-Serra P, Gonzalez-Beltran A, Izzo M, Lister A, Thurston M, Batista D, Granell R, Adekale M *et al* (2019) FAIRsharing as a community approach to standards, repositories and policies. *Nat Biotechnol* 37: 358 – 367

Scharm M, Waltemath D, Mendes P, Wolkenhauer O (2016a) COMODI: an ontology to characterise differences in versions of computational models in biology. *J Biomed Semantics* 7: 46

Scharm M, Wolkenhauer O, Waltemath D (2016b) An algorithm to detect and communicate the differences in computational models describing biological systems. *Bioinformatics* 32: 563 – 570

Schliess F, Hoehme S, Henkel SG, Ghallab A, Driesch D, Böttger J, Guthke R, Pfaff M, Hengstler JG, Gebhardt R *et al* (2014) Integrated metabolic spatial-temporal model for the prediction of ammonia detoxification during liver damage and regeneration. *Hepatology* 60: 2040 – 2051

Smith LP, Moodie SL, Bergmann FT, Gillespie C, Keating SM, König M, Myers CJ, Swat MJ, Wilkinson DJ, Hucka M (2020) Systems Biology Markup Language (SBML) Level 3 Package: Distributions, Version 1, Release 1, *J Integr Bioinform* (published online ahead of print), 20200018. https://doi.org/10.1515/jib-2020-0018

Smith LP, Bergmann FT, Chandran D, Sauro HM (2009) Antimony: a modular model definition language. *Bioinformatics* 25: 2452 – 2454

Smith LP, Hucka M, Hoops S, Finney A, Ginkel M, Myers CJ, Moraru I, Liebermeister W (2015) SBML level 3 package: hierarchical model composition, version 1 release 3. *J Integr Bioinform* 12: 268

Stanford NJ, Wolstencroft K, Golebiewski M, Kania R, Juty N, Tomlinson C, Owen S, Butcher S, Hermjakob H, Le Novère N *et al* (2015) The evolution of standards and data management practices in systems biology. *Mol Syst Biol* 11: 851

Swat MH, Thomas GL, Belmonte JM, Shirinifard A, Hmeljak D, Glazier JA (2012) Multi-scale modeling of tissues using CompuCell3D. In *Methods in Cell Biology*, Volume 110, pp. 325 – 366. Amsterdam: Elsevier

Tanaka S, Sichau D, Iber D (2015) LBIBCell: a cell-based simulation environment for morphogenetic problems. *Bioinformatics* 31: 2340 – 2347

Terfve C, Cokelaer T, Henriques D, MacNamara A, Goncalves E, Morris MK, van Iersel M, Lauffenburger DA, Saez-Rodriguez J (2012) CellNOptR: a flexible toolkit to train protein signaling networks to data using multiple logic formalisms. *BMC Syst Biol* 6: 133

The UniProt Consortium (2017) UniProt: the universal protein knowledgebase. *Nucleic Acids Res* 45: D158 – D169

Thiele I, Fleming RMT, Que R, Bordbar A, Diep D, Palsson BO (2012) Multiscale modeling of metabolism and macromolecular synthesis in *E. coli* and its application to the evolution of codon usage. *PLoS ONE* 7: e45635

Turing AM (1952) The chemical basis of morphogenesis. *Philos Trans R Soc Lond B Biol Sci* 237: 5 – 72

Uhlen M, Zhang C, Lee S, Sjöstedt E, Fagerberg L, Bidkhori G, Benfeitas R, Arif M, Liu Z, Edfors F *et al* (2017) A pathology atlas of the human cancer transcriptome. *Science* 357 : eaan2507

Varela PL, Ramos CV, Monteiro PT, Chaouiya C (2019) EpiLog: a software for the logical modelling of epithelial dynamics [version 2; peer review3 approved]. *F1000Research* 7: 1145

Voigt MA, Dräger A, Lloyd C, King ZA, Yang L (2018) draeger-lab/SBMLme: SBMLme converter (Version 0.0.6). Available from Zenodo at https://doi.org/10.5281/zenodo.1238905

Von Bertalanffy L (1950) An outline of general system theory. *Br J Philos Sci* 1: 134 – 165

Waltemath D, Adams R, Bergmann FT, Hucka M, Kolpakov F, Miller AK, Moraru II, Nickerson D, Sahle S, Snoep JL *et al* (2011) Reproducible computational biology experiments with SED-ML–the simulation experiment description markup language. *BMC Syst Biol* 5: 198

Waltemath D, Bergmann FT, Chaouiya C, Czauderna T, Gleeson P, Goble C, Golebiewski M, Hucka M, Juty N, Krebs O *et al* (2014) Meeting report from the fourth meeting of the Computational Modeling in Biology Network (COMBINE). *Stand Genomic Sci* 9: 1285 – 1301

Waltemath D, Wolkenhauer O (2016) How modeling standards, software, and initiatives support reproducibility in systems biology and systems medicine. *IEEE Trans Biomed Eng* 63: 1999 – 2006

Wang Y, Eddy JA, Price ND (2012) Reconstruction of genome-scale metabolic models for 126 human tissues using mCADRE. *BMC Syst Biol* 6: 153

Watanabe LH, Myers CJ (2016) Efficient analysis of systems biologymarkup language models of cellular populations using arrays. *ACS Synth Biol* 5: 835 – 841

Watanabe LH, König M, Myers CJ (2018) Dynamic Flux Balance AnalysisModels in SBML. *bioRxiv* 245076. https://doi.org/10.1101/245076

Weidemann A, Richter S, Stein M, Sahle S, Gauges R, Gabdoulline R, Surovtsova I, Semmelrock N, Besson B, Rojas I *et al* (2008) SYCAMORE—a systems biology computational analysis and modeling research environment. *Bioinformatics* 24: 1463 – 1464

Wittig U, Rey M, Weidemann A, Kania R, Müller W (2017) SABIO-RK: an updated resource for manually curated biochemical reaction kinetics. *Nucleic Acids Res* 46: D656 – D660

Wolstencroft K, Krebs O, Snoep JL, Stanford NJ, Bacall F, Golebiewski M, Kuzyakiv R, Nguyen Q, Owen S, Soiland-Reyes S *et al* (2016) FAIRDOMHub: a repository and collaboration environment for sharing systems biology research. *Nucleic Acids Res* 45: D404 – D407

Zhang F, Angermann BR, Meier-Schellersheim M (2013) The Simmune Modeler visual interface for creating signaling networks based on bi-molecular interactions. *Bioinformatics* 29: 1229 – 1230

Zhang F, Meier-Schellersheim M (2018) SBML Level 3 package: multistate, multicomponent and multicompartment species, version 1, release 1. *J Integr Bioinform* 15: 20170077

Appendix 1

Principal authors

Sarah M Keating [S.M.K.] (https://orcid.org/0000-0002-3356-3542) (1) Computing and Mathematical Sciences, California Institute of Technology, Pasadena, California 91125, US; (2) European Bioinformatics Institute, European Molecular Biology Laboratory (EMBL-EBI), Wellcome Genome Campus, Hinxton, Cambridgeshire, UK; (3) BioQuant/COS, Heidelberg University, Heidelberg 69120, Germany.

Dagmar Waltemath [D.W.] (https://orcid.org/0000-0002-5886-5563) Medical Informatics, Institute for Community Health, University Medicine Greifswald, Greifswald, Germany.

Matthias König [M.K.] (https://orcid.org/0000-0003-1725-179X) Institute for Theoretical Biology, Humboldt-University Berlin, Berlin, 10115, Germany.

Fengkai Zhang [F.Z.] (https://orcid.org/0000-0001-7112-9328) Laboratory of Immune System Biology, National Institute of Allergy and Infectious Diseases, National Institutes of Health, Bethesda, Maryland 20892, United States.

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

 © 2020 California Institute of Technology Published under the terms of the CC BY 4.0 license

Andreas Dräger [A.D.] (https://orcid.org/0000-0002-1240-5553) (1) Computational Systems Biology of Infection and Antimicrobial-Resistant Pathogens, Institute for Biomedical Informatics (IBMI), University of Tübingen, 72076 Tübingen, Germany; (2) Department of Computer Science, University of Tübingen, 72076 Tübingen, Germany; (3) German Center for Infection Research (DZIF), partner site Tübingen, Germany.

Claudine Chaouiya [C.C.] (https://orcid.org/0000-0003-2350-0756) (1) Aix Marseille Univ, CNRS, Centrale Marseille, I2M, Marseille, 13288, France; (2) Instituto Gulbenkian de Ciência, Oeiras, P-2780-156, Portugal.

Frank T Bergmann [F.T.B.] (https://orcid.org/0000-0001-5553-4702) BioQUANT/COS, Heidelberg University, Heidelberg, Germany.

Andrew Finney [A.M.F.] ANSYS UK Ltd, United Kingdom.

Colin S Gillespie [C.G.] (https://orcid.org/0000-0003-1787-0275) School of Mathematics, Statistics and Physics, Newcastle University, Newcastle upon Tyne, United Kingdom.

Tomáš Helikar [T.H.] (https://orcid.org/0000-0003-3653-1906) Department of Biochemistry, University of Nebraska – Lincoln, Lincoln, Nebraska 68588, United States.

Stefan Hoops [S.H.] (https://orcid.org/0000-0001-8503-8371) Biocomplexity Institute & Initiative, University of Virginia, 995 Research Park Boulevard, Charlottesville, Virginia 22911, United States.

Rahuman S Malik-Sheriff [R.M.-S.] (https://orcid.org/0000-0003-0705-9809) European Bioinformatics Institute, European Molecular Biology Laboratory (EMBL-EBI), Wellcome Genome Campus, Hinxton, Cambridge CB10 1SD, United Kingdom.

Stuart L Moodie [S.M.] (https://orcid.org/0000-0001-6191-5595) Eight Pillars Ltd, 19 Redford Walk, Edinburgh EH13 0AG, United Kingdom.

Ion I Moraru [I.M.] (https://orcid.org/0000-0002-3746-9676) Center for Cell Analysis and Modeling, UConn Health, Farmington, Connecticut 06030, United States.

Chris J Myers [C.M.] (https://orcid.org/0000-0002-8762-8444) Department of Electrical and Computer Engineering, University of Utah, Salt Lake City, Utah 84112, United States.

Aurélien Naldi [A.N.] (https://orcid.org/0000-0002-6495-2655) Computational Systems Biology Team, Institut de Biologie de l'ENS (IBENS), Département de biologie, École normale supérieure, CNRS, INSERM, Université PSL, 75005 Paris, France.

Brett G Olivier [B.O.] (https://orcid.org/0000-0002-5293-5321) (1) SysBioLab, AIMMS, Vrije Universiteit Amsterdam, De Boelelaan 1085, NL-1081HV Amsterdam, The Netherlands; (2) BioQuant/COS, Heidelberg University, Heidelberg, 69120, Germany; (3) Computing and Mathematical Sciences, California Institute of Technology, Pasadena, California 91125, United States.

Sven Sahle [S.S.1] COS Heidelberg/BIOQUANT, Heidelberg University, 69120 Heidelberg, Germany.

James C Schaff [J.C.S.] (https://orcid.org/0000-0003-3286-7736) Applied BioMath, LLC, Concord, Massachusetts 01742, United States.

Lucian P Smith [L.S.] (https://orcid.org/0000-0001-7002-6386) (1) Computing and Mathematical Sciences, California Institute of Technology, Pasadena, California, United States; (2) Department of Bioengineering, University of Washington, Seattle, Washington, United States.

Maciej J Swat [M.S.] Simcyp (a Certara company).

Denis Thieffry [D.T.1] (https://orcid.org/0000-0003-0271-1757) Institut de Biologie de l'ENS (IBENS), Département de biologie, École normale supérieure, CNRS, INSERM, Université PSL, 75005 Paris, France.

Leandro Watanabe [L.W.] (https://orcid.org/0000-0001-7030-8690) Department of Electrical and Computer Engineering, University of Utah, Salt Lake City, Utah, 84112, United States.

Darren J Wilkinson [D.J.W.] (https://orcid.org/0000-0003-0736-802X) (1) School of Mathematics, Statistics and Physics, Newcastle University, Newcastle upon Tyne, NE1 7RU, United Kingdom; (2) The Alan Turing Institute, British Library, London, NW1 2DB, United Kingdom.

Michael L Blinov [M.L.B.] (https://orcid.org/0000-0002-9363-9705) Center for Cell Analysis and Modeling, UConn Health, Farmington, Connecticut 06030, United States.

Kimberly Begley [K.B.] (https://orcid.org/0000-0002-1642-7493) Consultant, California Institute of Technology, Pasadena, California 91125, United States.

James R Faeder [J.F.] (https://orcid.org/0000-0001-8127-609X) Department of Computational and Systems Biology, University of Pittsburgh, Pittsburgh, Pennsylvania 15260, United States.

Harold F Gómez [H.G.] Department of Biosystems Science and Engineering, ETH Zürich, Mattenstrasse 26, 4058, Basel, Switzerland.

Thomas M Hamm [T.M.H.] (https://orcid.org/0000-0001-9579-7267) (1) Computational Systems Biology of Infection and Antimicrobial-Resistant Pathogens, Institute for Biomedical Informatics (IBMI), University of Tübingen, 72076 Tübingen, Germany; (2) Department of Computer Science, University of Tübingen, 72076 Tübingen, Germany.

Yuichiro Inagaki [Y.I.] (https://orcid.org/0000-0003-4011-8487) Management & IT Consulting Division, Mizuho Information & Research Institute, Inc., 2-3, Kanda-Nishikicho, Chiyoda-ku, Tokyo, 101-8443, Japan.

Wolfram Liebermeister [W.L.] (https://orcid.org/0000-0002-2568-2381) Université Paris-Saclay, INRAE, MaIAGE, 78350, Jouy-en-Josas, France.

Allyson L Lister [A.L.] (https://orcid.org/0000-0002-7702-4495) Oxford e-Research Centre (OeRC), Department of Engineering Science, University of Oxford, United Kingdom.

Daniel Lucio [D.L.] (https://orcid.org/0000-0002-8912-7213) College of Sciences, North Carolina State University, Raleigh, North Carolina 27695, United States.

Eric Mjolsness [E.M.] (https://orcid.org/0000-0002-9085-9171) Department of Computer Science, University of California, Irvine CA 92697-3435, United States.

Carole J Proctor [C.P.] (https://orcid.org/0000-0002-1366-1399) Institute of Cellular Medicine, Newcastle University, Newcastle upon Tyne, NE4 5PL, United Kingdom.

Karthik Raman [K.R.] (https://orcid.org/0000-0002-9311-7093) (1) Department of Biotechnology, Bhupat and Jyoti Mehta School of Biosciences, Indian Institute of Technology (IIT) Madras, Chennai - 600 036, India; (2) Initiative for Biological Systems Engineering (IBSE), IIT Madras; (2) Robert Bosch Centre for Data Science and Artificial Intelligence (RBC-DSAI), IIT Madras.

Nicolas Rodriguez [N.R.] (https://orcid.org/0000-0002-9290-7894) The Babraham Institute, Cambridge, CB22 3AT, United Kingdom.

Clifford A Shaffer [C.S.] (https://orcid.org/0000-0003-0001-0295) Department of Computer Science, Virginia Tech, Blacksburg, Virginia 24061, United States.

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

© 2020 California Institute of Technology Published under the terms of the CC BY 4

/

Bruce E Shapiro [B.S.] Department of Mathematics, California State University, Northridge, CA 91325, United States.

Joerg Stelling [J.S.] (https://orcid.org/0000-0002-1145-891X) Department of Biosystems Science and Engineering and SIB Swiss Institute of Bioinformatics, ETH Zurich, 4058 Basel, Switzerland.

Neil Swainston [N.S.] (https://orcid.org/0000-0001-7020-1236) Institute of Integrative Biology, University of Liverpool, Liverpool L69 7ZB, United Kingdom.

Naoki Tanimura [N.T.] Science Solutions Division, Mizuho Information & Research Institute, Inc., 2-3, Kanda-Nishikicho, Chiyoda-ku, Tokyo, 101-8443 Japan.

John Wagner [J.W.] IBM Research Australia, Melbourne, Australia.

Martin Meier-Schellersheim [M.M.-S.] (https://orcid.org/0000-0002-8754-6377) Laboratory of Immune System Biology, NIAID, Bethesda, Maryland 20892, United States.

Herbert M Sauro [H.M.S.] (https://orcid.org/0000-0002-3659-6817) University of Washington, Department of Bioengineering, Box 355061, Seattle, WA 98195-5061.

Bernhard Palsson [B.P.] (https://orcid.org/0000-0003-2357-6785) Department of Bioengineering, University of California, San Diego, La Jolla, California 92093, United States.

Hamid Bolouri [H.B.] Systems Immunology, Benaroya Research Institute at Virginia Mason, 1201 Ninth Avenue, Seattle, Washington 98101, United States.

Hiroaki Kitano [H.K.] (https://orcid.org/0000-0002-3589-1953) (1) The Systems Biology Institute, Tokyo, Japan; (2) Okinawa Institute of Science and Technology, Okinawa, Japan.

Akira Funahashi [A.F.] (https://orcid.org/0000-0003-0605-239X) Department of Biosciences and Informatics, Keio University, Yokohama, Kanagawa 223-8522, Japan.

Henning Hermjakob [H.H.] (https://orcid.org/0000-0001-8479-0262) European Bioinformatics Institute, European Molecular Biology Laboratory (EMBL-EBI), Wellcome Genome Campus, Hinxton, Cambridgeshire CB10 1SD, United Kingdom.

John C Doyle [J.C.D.] (https://orcid.org/0000-0002-1828-2486) Computing and Mathematical Sciences, California Institute of Technology, Pasadena, California 91125, United States.

Michael Hucka [M.H.] (https://orcid.org/0000-0001-9105-5960) Computing and Mathematical Sciences, California Institute of Technology, Pasadena, California 91125, United States.

SBML community members

Richard R Adams [R.A.] (https://orcid.org/0000-0003-1501-0941) ResearchSpace, 24 Fountainhall Road, Edinburgh EH9 2LW, United Kingdom.

Nicholas A Allen [N.A.] Amazon Web Services, Seattle, Washington, 98121, United States.

Bastian R Angermann [B.A.] (https://orcid.org/0000-0002-1328-8790) Computational Biology Unit, Laboratory of Systems Biology, National Institute of Allergy and Infectious Diseases, National Institutes of Health, Bethesda, Maryland 20892, United States.

Marco Antoniotti [M.A.] (https://orcid.org/0000-0002-2823-6838) Dipartimento di Informatica, Sistemistica e Comunicazione, Università degli Studi di Milano-Bicocca, Milan, Italy.

Gary D Bader [G.B.] (https://orcid.org/0000-0003-0185-8861) The Donnelly Centre, University of Toronto, Toronto M5S 3E1, Canada.

Jan Červený [J.ĂŒ.] (https://orcid.org/0000-0002-5046-3105) (1) Department of Adaptive Biotechnologies, Global Change Research Institute CAS, Brno, 60300, Czech Republic; (2) Systems Biology Laboratory, Faculty of Informatics, Masaryk University, Brno, 60300, Czech Republic; (3) Climate Change Cluster (C3), University of Technology Sydney, Ultimo, NSW 2007, Australia.

Mélanie Courtot [M.C.] (https://orcid.org/0000-0002-9551-6370) European Bioinformatics Institute, European Molecular Biology Laboratory (EMBL-EBI), Wellcome Genome Campus, Hinxton, Cambridgeshire CB10 1SD, United Kingdom.

Chris D Cox [C.D.C.] (https://orcid.org/0000-0001-9818-5477) Civil and Environmental Engineering, University of Tennessee, Knoxville, Tennessee 37996, United States.

Piero Dalle Pezze [P.D.P.] (https://orcid.org/0000-0003-1695-6763) The Babraham Institute, Babraham Campus, Cambridge, CB22 3AT, United Kingdom.

Emek Demir [E.D.] (https://orcid.org/0000-0002-3663-7113) (1) Department of Molecular and Medical Genetics, Oregon Health and Science University, Portland, Oregon 97239, United States; (2) Computational Biology Program, Oregon Health and Science University, Portland, Oregon 97239, United States.

William S Denney [W.D.] (https://orcid.org/0000-0002-5759-428X) Human Predictions, LLC, Cambridge, Massachusetts 02139, United States.

Harish Dharuri [H.D.] Illumina Inc., Foster City, California 94404, United States.

Julien Dorier [J.D.] (https://orcid.org/0000-0002-6717-2451) Vital-IT, SIB Swiss Institute of Bioinformatics, 1015 Lausanne, Switzerland.

Dirk Drasdo [D.D.] (https://orcid.org/0000-0002-9314-6281) (1) INRIA de Paris and Sorbonne Universites, UPMC Univ paris 6, LJLL, France; (2) Leibniz Research Centre for Working Environment and Human Factors, Dortmund, Germany; (3) IZBI, University of Leipzig, Germany.

Ali Ebrahim [A.E.] Department of Bioengineering, University of California, San Diego, La Jolla, California 92093, United States.

Johannes Eichner [J.E.] Computational Systems Biology of Infection and Antimicrobial-Resistant Pathogens, Center for Bioinformatics Tübingen (ZBIT), 72076 Tübingen, Germany.

Johan Elf [J.S.E.] (https://orcid.org/0000-0001-5522-1810) Dept Cell and Molecular Biology, SciLifeLab, Uppsala University, Uppsala, Sweden.

Lukas Endler [L.E.] (https://orcid.org/0000-0002-9115-8756) Institut f. Populationsgenetik, Vetmeduni Wien, 1210 Wien, Austria.

Chris T Evelo [C.E.] (https://orcid.org/0000-0002-5301-3142) (1) Department of Bioinformatics - BiGCaT, NUTRIM, Maastricht University, 6229ER Maastricht, The Netherlands; (2) Maastricht Centre for Systems Biology (MaCSBio), Maastricht University, 6229ER Maastricht, The Netherlands.

Christoph Flamm [C.F.] (https://orcid.org/0000-0001-5500-2415) Department of Theoretical Chemistry, University of Vienna, Vienna 1090, Austria.

Ronan MT Fleming [R.F.] (https://orcid.org/0000-0001-5346-9812) (1) Division of Systems Biomedicine and Pharmacology, Leiden Academic Centre for Drug Research, Faculty of Science, Leiden University, The Netherlands; and (2) School of Medicine, National University of Ireland, Galway, Ireland.

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

 © 2020 California Institute of Technology Published under the terms of the CC BY 4.0 license

Martina Fröhlich [M.F.] (https://orcid.org/0000-0001-6760-7200) Division of Applied Bioinformatics, DKFZ and NCT Heidelberg, 69120, Heidelberg, Germany.

Mihai Glont [M.G.G.] (https://orcid.org/0000-0002-8211-7581) European Bioinformatics Institute, European Molecular Biology Laboratory (EMBL-EBI), Wellcome Genome Campus, Hinxton, Cambridge, CB10 1SD, United Kingdom.

Emanuel Gonçalves [E.G.] (https://orcid.org/0000-0002-9967-5205) Wellcome Sanger Institute, Wellcome Genome Campus, Hinxton, United Kingdom.

Martin Golebiewski [M.G.] (https://orcid.org/0000-0002-8683-7084) HITS gGmbH, Schloss-Wolfsbrunnenweg 35, 69118 Heidelberg, Germany.

Hovakim Grabski [H.G.] (https://orcid.org/0000-0001-6115-9339) Department of Medical Biochemistry and Biotechnology, Institute of Biomedicine and Pharmacy, Russian-Armenian University, Yerevan 0051, Armenia.

Alex Gutteridge [A.G.] (https://orcid.org/0000-0001-7515-634X) GSK.

Damon Hachmeister [D.G.] CMT, LLC.

Leonard A Harris [L.H.] (https://orcid.org/0000-0003-2112-6940) Biochemistry, Vanderbilt University School of Medicine, Nashville, Tennessee 37232, United States.

Benjamin D Heavner [B.D.H.] (https://orcid.org/0000-0003-2898-9044) Department of Biostatistics, University of Washington, Seattle, Washington 98195, United States.

Ron Henkel [R.H.] (https://orcid.org/0000-0001-6211-2719) Business Information Systems, University of Rostock, 18059 Rostock, Germany.

William S Hlavacek [W.S.H.] (https://orcid.org/0000-0003-4383-8711) Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, United States.

Bin Hu [B.H.] (https://orcid.org/0000-0002-0278-8466) Clinical Cloud Bioinformatics, LLC, Atlanta, Georgia, United States.

Daniel R Hyduke [D.R.H.] Tegmine Therapeutics, San Francisco, California 94114, United States.

Kaito Ii [K.I.] Department of Bio-sciences and Informatics, Keio University, Kanagawa, 223-8522, Japan.

Hidde de Jong [H.d.J.] (https://orcid.org/0000-0002-2226-650X) Univ. Grenoble Alpes, Inria, 38000 Grenoble, France.

Nick Juty [N.J.] (https://orcid.org/0000-0002-2036-8350) School of Computer Science, University of Manchester, Manchester M13 9PL, United Kingdom.

Peter D Karp [P.K.] (https://orcid.org/0000-0002-5876-6418) Bioinformatics Research Group, SRI International, Menlo Park, California 94025, United States.

Jonathan R Karr [J.K.] (https://orcid.org/0000-0002-2605-5080) Icahn Institute and Department of Genetics and Genomic Sciences, Icahn School of Medicine at Mount Sinai, New York, New York, 10029, United States.

Douglas B Kell [D.K.] (https://orcid.org/0000-0001-5838-7963) Department of Biochemistry, Institute of Integrative Biology, Faculty of Health and Life Sciences, University of Liverpool, Crown St, Liverpool L69 7ZB, United Kingdom; (2) The Novo Nordisk Foundation Center for Biosustainability, Building 220, Chemitorvet, Technical University of Denmark, 2800 Kongens Lyngby, Denmark.

Roland Keller [R.K.] University of Tuebingen, Tübingen, Germany.

Ilya Kiselev [I.K.] (https://orcid.org/0000-0003-1057-5986) Institute of Computational Technologies SB RAS, Novosibirsk 630090, Russia.

Steffen Klamt [S.K.] (https://orcid.org/0000-0003-2563-7561) Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, Germany.

Edda Klipp [E.K.] (https://orcid.org/0000-0002-0567-7075) Theoretical Biophysics, Institute of Biology, Humboldt-Universität zu Berlin, 10099 Berlin, Germany.

Christian Knüpfer [C.K.] (https://orcid.org/0000-0002-1323-5445) Institute of Computer Science, Faculty of Mathematics and Computer Science, Friedrich Schiller University Jena, 07743 Jena, Germany.

Fedor Kolpakov [F.A.K.] (https://orcid.org/0000-0002-0396-0256) (1) Laboratory of Bioinformatics, Institute of Computational Technologies of SB RAS, Novosibirsk, 630090, Russian Federation; (2) Biosoft.Ru LLC, Novosibirsk, 630090, Russian Federation.

Falko Krause [F.K.] Theoretische Biophysik, Humboldt-Universität zu Berlin, 10099 Berlin, Germany.

Martina Kutmon [M.S.-K.] (https://orcid.org/0000-0002-7699-8191) (1) Department of Bioinformatics - BiGCaT, NUTRIM School of Nutrition and Translational Research in Metabolism, Maastricht University, 6229ER Maastricht, Netherlands; (2) Maastricht Centre for Systems Biology (MaCSBio), Maastricht University, 6229ER Maastricht, Netherlands.

Camille Laibe [C.L.1] (https://orcid.org/0000-0002-4625-743X) European Bioinformatics Institute, European Molecular Biology Laboratory (EMBL-EBI), Wellcome Genome Campus, Hinxton, Cambridgeshire, CB10 1SD, United Kingdom.

Conor Lawless [C.L.2] (https://orcid.org/0000-0002-4186-8506) Wellcome Centre for Mitochondrial Research, Newcastle University, Framlington Place, Newcastle upon Tyne, NE2 4HH, United Kingdom.

Lu Li [L.L.] (https://orcid.org/0000-0002-6199-522X) Signalling, The Babraham Institute, Cambridge, CB22 3AT, United Kingdom.

Leslie M Loew [L.M.L.] (https://orcid.org/0000-0002-1851-4646) University of Connecticut School of Medicine, Farmington, Connecticut 06030, United States.

Rainer Machne [R.M.] (https://orcid.org/0000-0002-1274-5099) Institute for Synthetic Microbiology, Heinrich Heine University Düsseldorf, Germany.

Yukiko Matsuoka [Y.M.] (https://orcid.org/0000-0002-5171-9096) The Systems Biology Institute, Tokyo, Japan.

Pedro Mendes [P.M.] (https://orcid.org/0000-0001-6507-9168) Center for Quantitative Medicine, University of Connecticut School of Medicine, Farmington, Connecticut 06030, United States.

Huaiyu Mi [H.M.] (https://orcid.org/0000-0001-8721-202X) Division of Bioinformatics, Department of Preventive Medicine, Keck School of Medicine of USC, Los Angenes, CA 90033, United States.

Florian Mittag [F.M.] (https://orcid.org/0000-0002-6472-0221) Tübingen, Germany.

Pedro T Monteiro [P.T.M.] (https://orcid.org/0000-0002-7934-5495) INESC-ID/Instituto Superior Técnico, Universidade de Lisboa, Rua Alves Redol 9, 1000-029 Lisbon, Portugal.

Kedar Nath Natarajan [K.N.] (https://orcid.org/0000-0002-9264-1280) University of Southern Denmark, Denmark.

Poul MF Nielsen [P.N.] (https://orcid.org/0000-0002-4704-0179) (1) Auckland Bioengineering Institute, The University of Auckland, Auckland, New Zealand; (2) Department of Engineering Science, The University of Auckland, Auckland, New Zealand.

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

© 2020 California Institute of Technology Published under the terms of the CC BY 4

Tramy Nguyen [T.N.] (https://orcid.org/0000-0002-8035-0259) Electrical and Computer Engineering, University of Utah, Salt Lake City, Utah 84129, United States.

Alida Palmisano [A.P.] (https://orcid.org/0000-0002-1859-3719) (1) Department of Computer Science, Virginia Tech, Blacksburg, Virginia, 24060, United States; (2) Department of Biological Sciences, Virginia Tech, Blacksburg, Virginia, 24060, United States.

Jean-Baptiste Pettit European Bioinformatics Institute, European Molecular Biology Laboratory (EMBL-EBI), Wellcome Genome Campus, Hinxton, Cambridgeshire CB10 1SD, United Kingdom.

Thomas Pfau [T.P.] (https://orcid.org/0000-0001-5048-2923) Systems Biology Group, Life Sciences Research Unit, University of Luxembourg, Luxembourg.

Robert D Phair [R.P.] (https://orcid.org/0000-0001-7979-2386) Integrative Bioinformatics, Inc.

Tomas Radivoyevitch [T.R.] (https://orcid.org/0000-0002-9701-1851) Quantitative Health Sciences, Cleveland Clinic, Cleveland, Ohio 44195, United States.

Johann M Rohwer [J.R.] (https://orcid.org/0000-0001-6288-8904) Department of Biochemistry, Stellenbosch University, Stellenbosch 7600, South Africa.

Oliver A Ruebenacker [O.R.] Broad Institute, Massachusetts, United States.

Julio Saez-Rodriguez [J.S-R.] (https://orcid.org/0000-0002-8552-8976) (1) Institute for Computational Biomedicine, Heidelberg University, Faculty of Medicine, & Heidelberg University Hospital, Bioquant, Heidelberg, Germany; (2) Joint Research Center for Computational Biomedicine, RWTH Aachen University, Faculty of Medicine, Aachen, Germany.

Martin Scharm [M.S.1] (https://orcid.org/0000-0003-4519-7030) Department of Systems Biology and Bioinformatics, University of Rostock, Rostock, 18051, Germany.

Henning Schmidt [H.S.] IntiQuan GmbH, CH-4055, Basel, Switzerland.

Falk Schreiber [F.S.] (1) Department of Computer and Information Science, University of Konstanz, Konstanz, 78464, Germany; (2) Faculty of IT, Monash University, Melbourne, Australia.

Michael Schubert [M.S.2] (https://orcid.org/0000-0002-6862-5221) European Molecular Biology Laboratory, European Bioinformatics Institute, Wellcome Genome Campus, Cambridge, CB10 1SD, United Kingdom.

Roman Schulte [R.S] Department of Computer Sience, Eberhard Karls University, Tübingen, 72074 Germany.

Stuart C Sealfon [S.C.S.] (https://orcid.org/0000-0001-5791-1217) Department of Neurology and Friedman Brain Institute, Icahn School of Medicine at Mount Sinai, New York, New York, 10029, United States.

Kieran Smallbone [K.S.] (https://orcid.org/0000-0003-2815-7045) Buxworth, High Peak SK23 7NS, United Kingdom.

Sylvain Soliman [S.S.2] (https://orcid.org/0000-0001-5525-7418) Inria Saclay-Île de France, Lifeware group, 91120 Palaiseau, France.

Melanie I Stefan [M.I.S.] (https://orcid.org/0000-0002-6086-7357) (1) Edinburgh Medical School: Biomedical Sciences, University of Edinburgh, Edinburgh, United Kingdom; (2) ZJU-UoE Institute, Zhejiang University School of Medicine, Haining, China.

Devin P Sullivan [D.P.S.] (https://orcid.org/0000-0001-6176-108X) Encodia Inc., San Diego, California, United States.

Koichi Takahashi [K.T.] (https://orcid.org/0000-0002-4235-9914) RIKEN Center for Biosystems Dynamics Research, Suita, Osaka 565-0874, Japan.

Bas Teusink [B.T.] (https://orcid.org/0000-0003-3929-0423) Amsterdam Institute for Molecules, Medicines and Systems, Vrije Universiteit Amsterdam, 1081HV Amsterdam, The Netherlands.

David Tolnay [D.T.2] (https://orcid.org/0000-0002-5281-6410) Computing and Mathematical Sciences, California Institute of Technology, Pasadena, California 91125, United States.

Ibrahim Vazirabad [I.V.] Diagnostic Laboratories and Blood Research Institute, Blood Center of Wisconsin, Milwaukee, Wisconsin, United States.

Axel von Kamp [A.K.] (https://orcid.org/0000-0002-2956-7815) Analysis and Redesign of Biological Networks, Max Planck Institute for Dynamics of Complex Technical Systems, 39106 Magdeburg, Germany.

Ulrike Wittig [U.W.] (https://orcid.org/0000-0002-9077-5664) Scientific Databases and Visualization, Heidelberg Institute for Theoretical Studies, 69118 Heidelberg, Germany.

Clemens Wrzodek [C.W.] (https://orcid.org/0000-0003-4995-780X) (1) Center for Bioinformatics Tuebingen (ZBIT), University of Tuebingen, Tübingen, Germany; (2) Roche Pharmaceutical Research and Early Development, pRED Informatics, Roche Innovation Center Munich, Nonnenwald 2, 82377 Penzberg, Germany.

Finja Wrzodek [F.W.] (https://orcid.org/0000-0001-5031-1016) (1) Center for Bioinformatics Tuebingen (ZBIT), University of Tuebingen, Tübingen, Germany; (2) Group Informatics, Roche Innovation Center Munich, Nonnenwald 2, 82377 Penzberg, Germany.

Ioannis Xenarios [I.X.] (https://orcid.org/0000-0002-3413-6841) Center for Integrative Genomics, Lausanne, University of Lausanne, Switzerland.

Takahiro G. Yamada [T.Y.] (https://orcid.org/0000-0002-1665-1778) Department of Biosciences and Informatics, Keio University, Yokohama, Kanagawa 223-8522, Japan.

Anna Zhukova [A.Z.] (https://orcid.org/0000-0003-2200-793) Institut Pasteur – Evolutionary Bioinformatics – C3BI, USR 3756 IP CNRS – Paris, France.

Jeremy Zucker [J.Z.] (https://orcid.org/0000-0002-7276-9009) Computational Biology, Pacific Northwest National Laboratory, Richland, Washington, 99352, United States.

Acknowledgments by SBML community members

B.A.'s work on SBML Level 3 was supported by the National Institutes of Health (NIH, US) and he is currently employed by AstraZeneca.

G.B. was supported by National Resource for Network Biology (NRNB, US), award 5P41-GM103504.

M.L.B. was supported by grant P41-GM103313 and R01-GM095485 from the NIH (US) to the University of Connecticut School of Medicine.

J.Č. was supported by the Czech Research Infrastructure Infrastructure for Systems Biology C4SYS, project LM2015055; GA ČR, grant 18-24397S; and the MEYS of the Czech Republic under OP RDE grant CZ.02.1.01/0.0/0.0/16_026/0008413 "Strategic Partnership for Environmental Technologies and Energy Production".

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].

 © 2020 California Institute of Technology Published under the terms of the CC BY 4.0 license

D.D. was supported by grant iLite (Innovations for LIver and Tissue Engineering) from Agence nationale de la recherche (ANR, France) to INRIA, Paris, France; by grant LiSyM (Liver Systems Medicine) from Bundesministerium für Bildung und Forschung (BMBF, Germany) to IfADo, Dortmund, Germany; by grant VLN (Virtual Liver Network) from Bundesministerium für Bildung und Forschung (BMBF, Germany) to IfADo, Dortmund, Germany; by grant Physicancer from Institut national de la santé et de la recherche médicale (INSERM, France) to INRIA, Paris, France; and by grant NOTOX (Predicting long-term toxic effects using computer models based on systems characterisations of organotypic cultures) from EU 7th Framework Programme to INRIA, Paris, France.

C.E. was supported by CHARME, COST, Action CA15110 MoU (PI: Chris Evelo) to Maastricht University.

R.F. was supported by the U.S. Department of Energy, Offices of Advanced Scientific Computing Research and the Biological and Environmental Research as part of the Scientific Discovery Through Advanced Computing program, grant DE-SC0010429.

A.G. was supported by the Japanese Society for the Promotion of Science.

M.G. was supported by funding to HITS from the German Federal Ministry of Education and Research (BMBF) as part of the Liver Systems Medicine Network LiSyM (FKZ 031L0056) and through the MulticellML project (FKZ 01ZX1707B), as well as from the European Union Horizon2020 framework programme of the European Commission under Grant Agreement 825843 and from the Klaus Tschira Foundation (KTS).

M.G.G. was supported by European Molecular Biology Laboratory (EMBL) core funding.

W.S.H. was supported by grant R01-GM111510 from the National Institute of General Medical Sciences (NIGMS, US) of the NIH.

D.K. was supported by Novo Nordisk Foundation grant NNF10CC1016517.

E.K. was supported by Deutsche Forschungsgemeinschaft (DFG, GRK1772, GRK 2290 and TRR175) and by People Programme (Marie Skłodowska-Curie Actions H2020-MSCA-ITN-2015-675585).

F.K. was supported by the Russian Foundation for Basic Research, 17-00-00296, Institute of Computational Technologies of SB RAS, Novosibirsk, 630090, Russian Federation (PI: Fedor Kolpakov).

J.K. was supported by grants R35-GM119771 and P41-EB023912 from the NIH (US) to the Icahn School of Medicine.

P.K. was supported by NIH grant GM075742 from the NIH (US) to SRI International.

S.K. was supported by grant 031L104B (de.NBI partner project "ModSim") by the German Federal Ministry of Education and Research to MPI Magdeburg.

L.M.L. was supported by grant P41-GM103313 from the NIH (US) to the University of Connecticut School of Medicine.

P.M. was supported by grant GM080219 from the NIH (US).

P.T.M was supported by grant PTDC/EEI-CTP/2914/2014 from Fundação para a Ciência e a Tecnologia (FCT) to INESC-ID.

A.P.'s contribution to this manuscript is a follow-up of the work done while employed at Virginia Tech. (VA, US). The opinions expressed in this manuscript are the authors' own and do not reflect the view of the National Institutes of Health, the Department of Health and Human Services, or the United States Government.

S.C.S. was supported by grant U19-AI117873 from the NIH (US).

License: This is an open access article under the terms of the Creative Commons Attribution 4.0 License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

Printed by [EMBO Press - 079.251.145.178 - /doi/epdf/10.15252/msb.20199110] at [28/08/2020].