UNIVERSITÉ
BORDEAUX
SEGALEN

EMBL-EBI 

Normes pour l'échange des modèles

en biologie des systèmes

Mémoire présenté le 12 Novembre 2013 par

Nicolas LE NOVÈRE

en vue de l'obtention d'une

Habilitation à Diriger des Recherches de
l'université BORDEAUX SEGALEN

Président du jury :	Pr Frédéric Yves Bois, professeur à l'université de Compiègne
Rapporteur :	Dr François Fages, directeur de recherche à l'INRIA
Rapporteur :	Dr Macha Nikolski, chargée de recherche à l'Université de Bordeaux 1
Rapporteur :	Dr Denis Thieffry, professeur à l'École Normale Supérieure de Paris
Membre du jury :	Dr Anne Poupon, directrice de recherche à l'INRA

Remerciements

Merci tout d'abord aux membres du jury de cette HDR, qui ont tous répondu présent à l'appel.

La recherche décrite dans ce manuscrit a bénéficié de l'apport d'un nombre très important de collègues. Il est malheureusement impossible de les remercier tous ici, et je m'en excuse. Il est également impossible de remercier toutes les organisations ayant participé au financement des réunions qui ont permis le développement des normes et de leur support informatique.

Cependant, j'aimerais mentionner le Laboratoire Européen de Biologie Moléculaire (EMBL), qui à travers le support financier de mon groupe durant 9 ans a permis l'émergence de toutes ces normes et le développement de plusieurs outils clés pour leur support. L'Organisation japonaise pour le Développement des Nouvelles Énergies et de la Technologie Industrielle (NEDO) a financé le développement de SBGN. Le Conseil pour la Recherche en Biotechnologie et en Sciences Biologiques britannique (BBSRC), l'Institut National des Sciences Médicales Générales américain (NIGMS) et la Commission Européenne ont financé à différents degrés plusieurs des normes présentées.

Mon activité dans le domaines des normes pour la biologie des systèmes n'aurait jamais existé sans le support de Dame Professeure Janet Thornton, qui m'a recruté au laboratoire Européen de Bioinformatique, et m'a toujours soutenu malgré l'aspect non traditionnel de cette activité.

Aucune de ces normes n'aurait vu le jour sans les impulsions visionnaires du Dr Hiroaki Kitano, une première fois pour SBML et une seconde pour SBGN. Ces visions se sont transformées en réalités largement grâce au Dr Michael Hucka, qui a fait de SBML une des normes les mieux supportées en biologie. Mike est également le premier à avoir entrevu l'importance de la sémantique et la nécessité de centraliser l'échange des modèles.

Le Dr Andrew Finney a joué un rôle important dans la genèse de MIRIAM et son support par SBML. Camille Laibe a développé l'infrastructure supportant l'utilisation des identifiants partagés, et Dr Nick Juty a curé le contenu du registre MIRIAM. Mélanie Courtot et Camille Laibe ont développé l'infrastructure supportant le développement et l'utilisation de SBO. Nick Juty a maintenu l'ontologie. Le Dr Dagmar Waltemath a participé à la genèse de MIASE, SED-ML et KiSAO, et est la principale source d'énergie derrière SED-ML. Dr Christian Knüpfer est le développeur initial et principal de TEDDY. Anna Zhukova a entièrement reconstruit KiSAO, est est responsable de la plus grande partie de son contenu actuel. En plus de son activité d'éditeur pour SED-ML, le Dr Frank Bergmann a supporté inlassablement nos efforts, développant outils de démonstration et bibliothèques logicielles.

Les formats normalisés décrits dans ce manuscrit sont maintenus par des éditeurs élus par la

communauté. Ces éditeurs ne sont pas rémunérés, et ont tous une activité professionnelle à coté. Leur travail d'éditeur est rarement reconnu, que ce soit par les utilisateurs ou d'un point de vue académique. Je les remercie, en mon nom, et au nom des coordinateurs du projet COMBINE.

Je voudrais aussi remercier la communauté toute entière des modélisateurs en biologie des systèmes pour leur réactivité, leur imagination et leur esprit d'initiative. Avoir un retour technique par des personnes qui maîtrisent les subtilités informatiques tout en comprenant les problèmes biologiques est sans prix. Mais plus encore, je remercie cette communauté pour son excellent esprit de camaraderie, qui transforme l'atmosphère des réunions de travail et permets des progrès plus rapide.

Mon grand écart entre modélisation des voies de signalisation et développement de normes ne m'a souvent pas permis d'accorder à mon groupe de recherche toute l'attention qui lui était due. Je m'en excuse et remercie ses membres de leur attitude compréhensive (teintée parfois d'incompréhension quant à l'objet de mes obsessions).

Enfin je veux remercier ma famille, qui a due supporter mes voyages incessants, mes discours techniques totalement incompréhensibles et mes tee-shirts aux acronymes mystérieux et à l'esthétique douteuse.

Résumé

Avec le succès croissant de la biologie des systèmes, modèles mathématiques et simulations numériques se multiplient dans la littérature. De plus, la taille de ces modèles augmente ainsi que leur complexité. Enfin, les utilisations des modèles informatiques se diversifient, impliquant analyses, représentations, intégration avec d'autre types de données etc. Des normes sont donc nécessaires pour s'assurer que les informations partagées, incluant la description des modèles ainsi que des procédures à leur appliquer, soient comprises et utilisées correctement. Il existe trois types principaux de normes en biologie. Les directives minimales listent l'information qui doit absolument être fournie avec une donnée afin de la rendre intelligible. Les formats structurés permettent d'encoder les données et les informations prescrites. Enfin les terminologies ou vocabulaires contrôlés, dont les ontologies, permettent une sémantique définie et partagée.

Le *Systems Biology Markup Language*, développé depuis 2000, fournit un format structuré pour encoder les modèles en biologie des systèmes. Cependant, la description de la seule structure des modèles ne suffit pas à leur vérification et utilisation. Depuis 2005, j'ai mené la construction de briques complémentaires pour construire un mur couvrant tout le cycle de vie des modèles et de leurs utilisations. La *Minimal Information Required in the Annotation of Models* (MIRIAM) liste l'information qui doit être fournie avec un modèle pour permettre une réutilisation optimale. La *Minimal Information About a Simulation Experiment* (MIASE) liste l'information qui doit être fournie pour permettre de reproduire une expérience de simulation numérique. Afin d'encoder l'information requise par MIASE, nous avons créé le *Simulation Experiment Description Markup Language* (SED-ML). Afin d'ajouter une couche sémantique aux formats structurés, nous avons créé trois ontologies. La *Systems Biology Ontology* caractérise les éléments d'un modèle. La *Kinetic Simulation Algorithm Ontology* classifies les algorithmes utilisés pour simuler et analyser les modèles. La *Terminology for the Description of Dynamics* permet de décrire le comportement temporel des variables d'un système. Enfin afin de faciliter l'exploration, la compréhension et l'échange des modèles, nous avons développé la *Systems Biology Graphical Notation* (SBGN). SBGN est un ensemble de langage visuels fait de symboles et syntaxes normalisés pour représenter les réseaux d'interactions biochimiques.

L'ensemble de ces outils informatiques a permis d'améliorer l'échange et la réutilisation des modèles en biologie des systèmes. En sus de cet objectif attendu, un nouveau domaine d'activité s'est développé, dont le sujet est la construction, la comparaison et l'intégration des modèles, entre eux et avec d'autres types de données biologiques.

Table des matières

Abréviations utilisées

Nota bene : Les versions étendues des acronymes sont considérées comme des noms propres, et sont utilisées telles quelles dans le texte, sans traduction en français.

COMBINE **Co**mputational **M**odeling in **Bi**ology **Ne**twork

KiSAO **K**inetic **S**imulation **A**lgorithm **O**ntology

MIASE **M**inimum **I**nformation **R**equired **I**n the **A**nnotation of **M**odels

MIRIAM **M**inimum **I**nformation **A**bout a **S**imulation **E**xperiment

SED-ML **S**imulation **E**xperiment **D**escription **M**arkup **L**anguage

SBGN **S**ystems **B**iology **G**raphical **N**otation

SBGN-ML **S**ystems **B**iology **G**raphical **N**otation **M**arkup **L**anguage

SBML **S**ystems **B**iology **M**arkup **L**anguage

SBO **S**ystems **B**iology **O**ntology

TEDDY **Te**rminology for the **D**escription of **Dy**namics

URI **U**niform **Resource Identifier**

XHTML e**X**tensible **H**ypertext **M**arkup **L**anguage

XML **E**xtensible **M**arkup **L**anguage

1 Introduction et présentation du travail

Il y a une quinzaine d'années, l'essor de la biologie des systèmes a profondément changé le paysage de la recherche en biologie. Il est encore tôt pour évaluer l'impact de cet évènement (parfois caractérisé comme un « changement de paradigme », similaire à l'arrivée de la physiologie au XIX^{ème} siècle ou de la biologie moléculaire au milieu du XX^{ème}). Il n'est pas du ressort de cette thèse de discuter de manière approfondie ce qu'est la biologie des systèmes. Le lecteur peut se reporter à des références externes comme Kitano (2002) ou Ideker *et al.* (2001). Il suffit de rappeler qu'il s'agit de considérer le fonctionnement d'une brique du vivant (par exemple une protéine) au sein d'un système d'interactions, et l'effet de ces interactions sur les propriétés émergentes de ce système. Bien qu'une telle approche ait été proposée il y a bien longtemps (von Bertalanffy, 1928), seule la disponibilité d'une large quantité de données expérimentales couplée à un accroissement considérable de la puissance informatique ont permis son déploiement massif.

Un des aspects clés de la biologie des systèmes est l'utilisation de modèles informatiques pour simuler les processus biologiques. Dès le milieu du siècle dernier, dans la discussion d'un papier où lui-même utilisait une approche analytique pour décrire les gradients de morphogène, Alan Turing appelait de ses vœux l'utilisation de tels modèles, (Turing, 1952). Ses désirs devaient être réalisés la même année par Alan Lloyd Hodgkin et Andrew Huxley lorsqu'ils expliquèrent la propagation des potentiels d'action le long des axones. Décrivant de manière quantitative un comportement cellulaire émergeant de l'interaction entre plusieurs composants moléculaires différents, les canaux potassium et sodium, leur travail peut être vu comme le départ de la biologie computationelle des systèmes. L'utilisation de modèles informatiques pour l'analyse des systèmes biologiques s'est ensuite progressivement développée, que ce soit pour les réseaux métaboliques (Chance *et al.*, 1969), les réseaux de régulation génétiques (Kauffman, 1969) ou bien les voies de signalisation (Meyer & Stryer, 1988).

1.1 Encodage des modèles dans un format standard

Le décollage de la biologie des systèmes a entraîné une prolifération de ces modèles informatiques (Figure 1.1). Bien que BioModels Database ne contienne pas tous les modèles publiés en biologie des systèmes, son contenu reflète l'évolution du domaine. Non seulement le nombre de modèles déposés croît de manière régulière, mais ils deviennent également plus complexes.

Alors qu'en 2005 le nombre moyen d'expressions mathématiques par modèle dans BioModels Database était environ 20, il est maintenant proche de 200. Pour de ne pas ré-inventer la roue et de se concentrer sur des questions nouvelles, il est important de pouvoir réutiliser les modèles existants. De plus, la complexité des modèles rend toute ré-implémentation ardue et sujette à erreur. Afin de pouvoir reproduire les résultats publiés, et ré-utiliser les modèles, tels quels ou modifiés, il faut donc disposer du code-source de ces modèles.

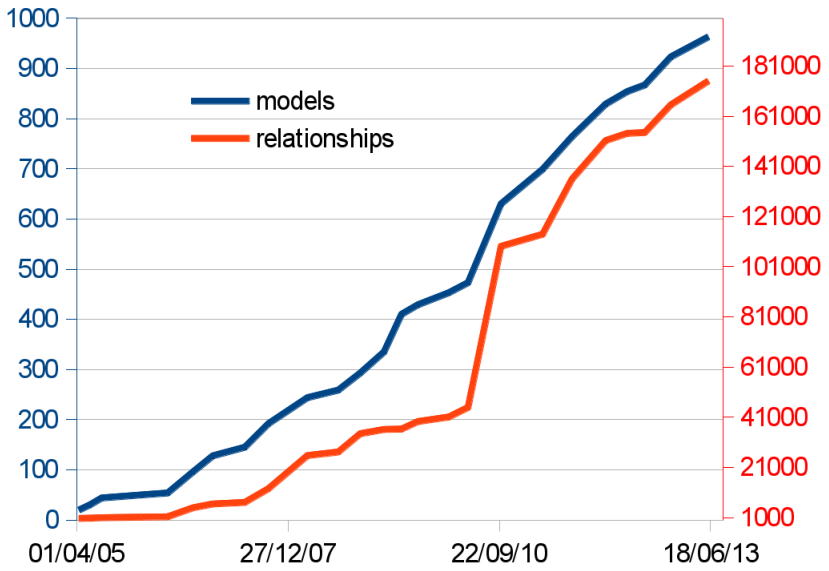


FIGURE 1.1 – Croissance du contenu de la branche « littérature » de BioModels Database depuis sa création.

Cependant, disposer du code-source ne suffit pas. Il faut pouvoir le compiler — si le modèle est directement écrit dans un language de programmation —, ou obtenir le logiciel de simulation utilisé — si le modèle est formé de fichiers de configuration. D'une part, cette compilation peut être ardue. Certains logiciels requiert l'utilisation de bibliothèques logicielles difficiles d'accès ou d'installation, commerciales, ou incompatible entre elles. Certains logiciels ne peuvent pas être installés du tout sur une plate-forme locale. D'autre part, l'utilisation du même logiciel permet de répéter une expérience, mais pas de la répliquer. Si le logiciel contient une erreur générant des résultats incorrects, ces erreurs vont être répétées. C'est pourquoi il est important, voire nécessaire, d'utiliser des formats ouverts pour échanger les modèles, lus et compris par plusieurs logiciels. Les modélisateurs des processus biochimiques et cellulaires, bientôt renommés biologistes des systèmes, lancèrent plusieurs projets à la fin des années 1990s en vue de développer de tels formats (pour un bref historique voir Kell & Mendes, 2008).

Le plus utilisé de ces formats est le *Systems Biology Markup Language* (SBML) (Hucka *et al.*, 2003), créé au sein du ERATO Kitano project en collaboration avec un groupe de développeurs représentant sept logiciels de simulation. Alors en stage post-doctoral chez Dennis

Bray à l'université de Cambridge (GB), je participais au développement d'un simulateur stochastique de réactions biochimiques (StochSim, Le Novère & Shimizu, 2001), ce qui me permis de rejoindre le projet dès l'origine. SBML est un format déclaratif et non procédural. Il permet de décrire la structure d'un modèle mathématique, mais non sa construction (par exemple à partir de règles) ou bien les opérations à mener pour obtenir un résultat numérique. SBML est un langage structuré basé sur XML (Bray *et al.*, 1997). Un fichier SBML décrit (Figure 1.2) :

- les variables d'un modèle, à la fois les variables d'état (par exemple taille des compartiments, concentrations des espèces moléculaires etc.) et les paramètres (qui peuvent être eux-mêmes constants ou variables) ;
- les relations mathématiques liant les variables entre elles, que ce soit des expressions algébriques explicites, des expressions différentielles explicites ou implicites (par exemple reconstructible à partir des cinétiques chimiques), ou bien des évènements conditionels discrets ;
- des métadonnées, soit en format libre (n'importe quelle code XHTML valide est autorisée) lisible par un être humain, soit structurées et destinées à être lues par un programme informatique.

Le développement de SBML a eu un effet profond sur la communauté des modélisateurs en biologie des systèmes. L'apparition cette *lingua franca* s'est accompagnée d'outils facilitant son utilisation, comme les bibliothèques standards libSBML (Bornstein *et al.*, 2008) et JSBML (Dräger *et al.*, 2011), ainsi que les connecteurs à des outils de modélisation usuels comme MatLab (SBMLtoolbox, Keating *et al.*, 2006) et Mathematica (MathSBML Shapiro *et al.*, 2004). Le résultat a été un support relativement rapide par un grand nombre d'outils[1], une acceptation par la communauté des modélisateurs[2], et le développement d'un « écosystème » d'outils informatiques pour la création, l'analyse et l'échange des modèles en biologie des systèmes. Mais plus encore, de par son développement ouvert, impliquant tout volontaire, SBML a mis en contact des scientifiques d'origines différentes, physiciens, ingénieurs, biologistes, mathématiciens, et s'intéressant à des champs variés du vivant. Outre l'ouverture d'esprit engendrée par ces débats, de nombreuses collaborations scientifiques ont également vues le jour. SBML est développé sur un mode consensuel, et sont développement est coordonné par un groupe d'éditeurs élus par la communauté. On peut donc parler d'une norme plutôt que d'un standard (les deux mots se traduisant par *standard* en anglais). La communauté a fait un pas supplémentaire vers une normalisation « classique » avec la creation de COMBINE[3] (Computational Modeling in Biology Network), qui coordonne le développement de tous les efforts décrits dans ce manuscrit.

Le focus initial de SBML était la modélisation quantitative des réseaux biochimiques par des approches de cinétique chimique. Les raisons en étaient historiques (SBML s'est basé sur

[1] 254 étaient recensés en septembre 2013 dans le guide des logiciels SBML `http://sbml.org/SBML_Software_Guide`)

[2] Le papier princeps (Hucka *et al.*, 2003) était cité plus de 1800 fois d'après Google Scholar en Septembre 2013

[3] `http://co.mbine.org`

```xml
<?xml version="1.0" encoding="UTF-8"?>
<sbml xmlns="http://www.sbml.org/sbml/level2/version4" level="2" version="4">
  <model name="Simple Model">
    <listOfCompartments>
      <compartment id="cell" size="1" />
    </listOfCompartments>
    <listOfSpecies>
      <species id="A" compartment="cell" initialConcentration="1"/>
      <species id="B" compartment="cell" initialConcentration="1"/>
    </listOfSpecies>
    <listOfParameters>
      <parameter id="k1" value="0.1"/>
    </listOfParameters>
    <listOfReactions>
      <reaction id="r1" reversible="false">
        <listOfReactants>
          <speciesReference species="A"/>
        </listOfReactants>
        <listOfProducts>
          <speciesReference species="B"/>
        </listOfProducts>
        <kineticLaw>
          <math xmlns="http://www.w3.org/1998/Math/MathML">
            <apply>
              <times/>
              <ci> cell </ci>
              <ci> k1 </ci>
              <ci> A </ci>
            </apply>
          </math>
        </kineticLaw>
      </reaction>
    </listOfReactions>
  </model>
</sbml>
```

FIGURE 1.2 – Fichier SBML simple, décrivant la conversion d'une espèce moléculaire A en une espèce B via une réaction. Il est important de noter qu'une *species* SBML n'est pas obligatoirement une espèce moléculaire, mais toute chose qui peut être transformée en d'autres par des processus.

un format développé pour échanger des modèles métaboliques[4]), pratiques (les équipes créant SBML en 2000 développaient toutes des logiciels utilisant ces représentations) et politiques (la compréhension des réseaux biochimiques était une cible relativement plus aisée à atteindre pour démontrer l'utilité de la modélisation en biologie). Cependant nous étions conscients des limitations de cette approche, et des extensions ont été proposées dès la fin 2000. Ces extensions ont finalement vu le jour sous la formes des paquets de SBML niveau 3.

[4]Les premières tentatives peuvent être trouvés à `http://lenoverelab.org/documents/MML-10May2000.pdf` et `http://lenoverelab.org/documents/1st-mention-SBML-8August2000.pdf`.

1.2 Curation des modèles

Avec l'accroissement du nombre de modèles encodés en SBML (mais aussi en d'autres formats comme CellML, Lloyd *et al.*, 2004) — en particulier des modèles appliqués à des questions biologiques précises — ainsi que de la prolifération d'outils informatiques utilisant SBML, il devint clair que l'absence de critères de qualité ainsi que de sources sures où obtenir des modèles posait problème. Après le « hackathon » SBML 2004 que j'organisais à l'EBI, Mike Hucka, directeur du groupe gérant les spécifications et les outils clés supportant SBML, esquissa un projet qu'il appela « BioModels.net »(Hucka, 2004). Le projet ne vit jamais le jour sous la forme envisagée. Cependant plusieurs des idées décrites dans le document initial ont guidé la communauté pendant presque une décennie. La première proposition était l'établissement d'un dépôt de modèles centralisé, indépendamment du format, proposant des outils de simulation et d'analyse. Cette idée sera rejetée lors d'une réunion de toutes les parties prenantes à Heidelberg en automne 2004. Cela nous entraînera à centrer les versions initiales de BioModels Database (Le Novère *et al.*, 2006; Li *et al.*, 2010b)[5] uniquement sur les modèles encodés en SBML. En revanche il y eu unanimité sur la nécessité d'améliorer la qualité des modèles, et de définir des critères sans lesquels toute échange est inutile. Une fois discutés et formalisés, ces critères formeront la *Minimum Information Required In the Annotation of Models*[6] (MIRIAM)[7], sujet du chapitre 2.

MIRIAM est formée de 2 parties distinctes. D'une part, une série de règles fournissent des contraintes auxquelles le modèle doit souscrire (codage dans un format ouvert, correspondance topologique avec les phénomènes représentés, modèle fournit avec toutes les informations nécessaires à la poursuite de simulations numériques etc.). D'autre part, le modèle doit être accompagné de métadonnées globalement pertinentes (attribution, licence etc.) ou bien reliées à un composant particulier du modèle, comme des références croisées vers d'autres sources de données. Tout au long de l'année 2005, Andrew Finney et moi avons essayé de formaliser des « triples » pour encoder cette seconde partie dans des modèles SBMLLe Novère & Finney (2005). Cette formalisation est souvent appelée « MIRIAM annotation ». Une des structures cruciales de ces annotations consiste en des URIs (Berners-Lee *et al.*, 2005) partagés[8]. Ces URIs permettent en effet de relier un composant d'un modèle avec ce qu'il est censé représenter, mais aussi d'identifier des composants différents, parfois dans des modèles différents. Dans le papier initial (chapitre 2), nous mentionnions :

> *To enable interoperability, the community will have to agree on a set of standard, valid URIs. An online resource will be established to catalog the URIs and the*

[5] `http://www.ebi.ac.uk/biomodels`

[6] pour des raisons politiques qui ont bien vite disparues, le titre de l'article utilisait le verbe « requested » au lieu de l'injonction plus ferme « required » et limitait la portée du travail aux modèles biochimiques. Le nom officiel, désormait présent partout, utilise bien « required », et s'applique à tous les modèles quantitatifs décrivant des processus.

[7] `http://co.mbine.org/standards/miriam`

[8] `http://co.mbine.org/standards/miriam_uris`

> *corresponding physical URLs of the agreed-upon "data-types," whether these are controlled vocabularies or databases. This catalog will simply list the URIs and for each one, provide a corresponding summary of the syntax for the "identifier." An application programming interface (API) can be created so that software tools can retrieve valid URL(s) corresponding to a given URI.*

Pour les besoins de BioModels Database, j'ai développé un outil correspondant à la description ci-dessus, appelée *MIRIAM Resources* (Laibe & Le Novère, 2007), puis *MIRIAM Registry* (Juty *et al.*, 2012). En sus de la base de donnée contenant les informations sur les ressources à utiliser dans les URIs, nous fournissons un support logiciel incluant des Web Services pour utiliser la base à distance (Li *et al.*, 2010a). L'utilisation de ces URIs s'est progressivement étendue non seulement à d'autres formats encodant des modèles, mais également à d'autres types de données biologiques dans le cadre du web sémantique, en particulier avec le déploiement des URIs « Identifiers.org » (Juty *et al.*, 2012).

1.3 Description des simulations

BioModels Database comprend une branche distribuant des modèles en conformité avec MIRIAM. En particulier les curateurs vérifient que les simulations décrites dans le papier de référence reproduisent les résultats attendus. En ce qui concerne les simulations, MIRIAM précise :

> *6. The model, when instantiated within a suitable simulation environment, must be able to reproduce all relevant results given in the reference description that can readily be simulated. Not only does the simulation have to provide results qualitatively similar to the reference description, such as oscillation, bistability, chaos, but the quantitative values of variables, and their relationships (e.g., the shape of the phase portrait) must be reproduced within some epsilon, the difference being attributable to the algorithms used to run the simulation, and the roundup errors. Some software exists that can help to compare qualitatively the results of a simulation with a benchmark ; see for instance BIOCHAM.*

Cependant, MIRIAM ne précise pas quelles informations sont nécessaires pour implémenter cette règle. Si en général cette vérification est aisée à faire[9] — à condition que les autres points de MIRIAM soient validés, en particulier concernant les conditions initiales—, la façon dont les résultats sont présentés dans les articles cause parfois des problèmes. La figure 1.3 présente quatres exemples où le résultat présenté n'est pas directement le résultat numérique brut fourni par le logiciel de simulation.

[9] Nous n'entrerons pas ici dans une discussion comparant approches déterministes et stochastiques. Les auteurs de MIRIAM étaient au courant du problème.

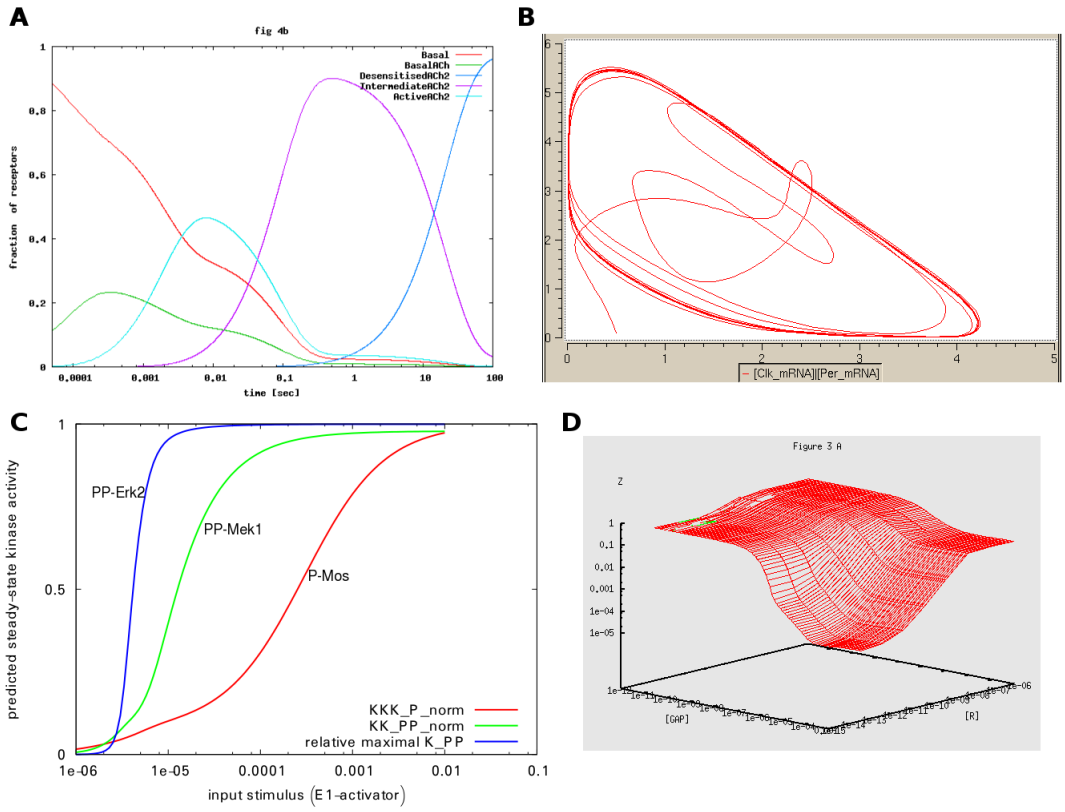


FIGURE 1.3 – Graphes de simulation obtenus avec des modèles de BioModels Database, reproduisant les résultats présentés dans les papiers originaux. Les étapes de simulation et présentation sont de complexité croissante. A) Simulation du modèle BIOMD0000001 reproduisant la Figure 4b de Edelstein *et al.* (1996). L'axe des abscisses est le logarithme du temps. B) Simulation du modèle BIOMD0000022 reproduisant la Figure 3d de Ueda *et al.* (2001). La quantité d'ARNm du gène *dClk* est reportée en fonction de celui du gène *Per*. C) Simulation du modèle BIOMD0000000009 reproduisant la Figure 2B de Huang & Ferrell (1996). Le graphe représente les formes actives des MAP kinase, MAP kinase kinase et MAP kinase kinase kinase à l'état stationnaire après stimulation par Ras. Les valeurs sont normalisées. Les résultats ont été obtenus par 100 simulations avec des valeurs logarithmiquement croissantes de Ras. D) Simulation du modèle BIOMD0000000086 reproduisant la Figure 3A de Bornheimer *et al.* (2004). Le graphe représente la fraction active de protéine G (somme de toutes les formes moléculaires du modèle contenant une protéine G active) à l'état stationnaire après stimulation des récepteurs les activant. Les résultats ont été obtenus par un balayage bi-dimensionnel de paramètres avec des valeurs logarithmiquement croissantes de récepteurs et de protéines GAP.

J'ai donc décidé de développer un triptyque {directives, format, terminologie} pour les expériences de simulations, miroir de {MIRIAM, SBML, SBO[10]} pour la structure des modèles.

[10] voir la section 1.4 **Sémantique et ontologies** pour plus d'information concernant SBO.

L'occasion m'en a été donnée par la visite de Dagmar Waltemath, alors étudiante en thèse à l'université de Rostock.

Le premier volet de ce triptyque est l'*Minimal Information About a Simulation Experiment* (MIASE) (Waltemath *et al.*, 2011a)[11], sujet du chapitre 3. Les règles de MIASE sont divisées en trois volets : 1) L'identification des modèles et des modifications à leur appliquer, 2) la description des étapes de simulations et autres procédures utilisées, 3) la description de la nature et du format des données fournies par l'expérience.

Le second volet du triptyque est un format permettant de coder de manière formelle les informations requises par MIASE. Il s'agit du *Simulation Experiment Description Markup Language* (SED-ML) (Köhn & Le Novère, 2008; Waltemath *et al.*, 2011b)[12], sujet du chapitre 4. La structure de SED-ML suit dans les grandes lignes les directives MIASE. Seuls quelques types de simulations étaient supportés dans la version initiale[13]. Une version plus récente inclue un support des procédures répétées et emboîtées[14].

Le troisième volet du triptyque est formé par une terminologie décrite dans la section suivante.

1.4 Sémantique et ontologies

Les directives MIRIAM et MIASE décrivent les informations qu'il faut échanger afin de réutiliser des modèles et reproduire des simulations. Les langages SBML et SED-ML fournissent un moyen d'encoder ces informations de manière formelle. Cependant, ces formats sont pauvres en sémantique, que ce soit la sémantique biologique (ce que représentent les objets encodés) ou la sémantique de la modélisation elle-même (par exemple ce que signifie les mathématiques employées, quelles sont les hypothèses etc.)[15]. Une manière d'ajouter une couche sémantique est l'utilisation d'ontologies. Une discussion approfondie de ce qu'est une ontologie, en philosophie ou en informatique, dépasse le cadre de cette introduction. Le lecteur peut se référer à des ouvrages comme *Applied Ontology* (Munn & Smith, 2008). Disons simplement qu'une ontologie est une terminologie dont les entrées sont reliées par des relations explicites. Une **automobile** *est un* **véhicule**, un **bras** *est une partie de* **corps**, **liquide** *est un état de* **l'eau**, etc.

La première de ces ontologies, la *Systems Biology Ontology* (SBO)[16], est née en 2004, lors du forum SBML à Heidelberg. Mon idée initiale était de stocker, sous une forme digitale structurée et pérenne, les connaissances sur la modélisation en biologie des systèmes (et plus pré-

[11] `http://co.mbine.org/standards/miase`

[12] `http://sed-ml.org`

[13] `http://identifiers.org/combine.specifications/sed-ml.level-1.version-1`

[14] `http://identifiers.org/combine.specifications/sed-ml.level-1.version-1.RC`

[15] Certains formats utilisés en biologie des systèmes sont bien sûr riche en sémantique. C'est le cas de BioPAX Demir *et al.* (2010), qui est un format d'échange mais aussi une ontologie, basé sur XML, RDF et OWL. Comme BioPAX ne concerne pas pour l'instant les modèles mathématiques, je n'en parlerai pas plus dans ce manuscrit.

[16] `http://www.ebi.ac.uk/sbo/`

cisément pour l'anecdote « le contenu du cerveau de Pedro Mendes » !). Très vite nous nous sommes rendu compte que l'utilisation de SBO pouvait aider à encoder les hypothèses des modélisateurs, facilitant par exemple l'utilisation des modèles dans différents paradigmes de simulation, comme discrets ou continus, ou bien encore la conversion de modèles utilisant des lois approximatives, comme Michaelis-Menten, en modèles utilisant des lois plus générales, comme la loi d'action de masse. Lors du développement de SBGN (voir la section 1.5 **Représentation graphique des modèles**), il est devenu clair que SBO pourrait être une des « glues » entre les différents formats. Ces utilisations ont déterminé la structure actuelle de l'ontologie. SBO comporte différentes branches correspondant aux différents types d'entités présentes dans un modèle. Par exemple la branche **physical entity representation** contient un terme **macromolecule**, qui peut être utilisé pour annoter une *species* de SBML, la branche **occurring entity representation** contient un terme **phosphorylation** qui peut être utilisé pour annoter une *reaction* de SBML, la branche **mathematical expression** contient un terme **Henri-Michaelis-Menten rate law** qui peut être utilisé pour annoter une *kineticLaw* SBML. L'ensemble de l'ontologie est une hyponymie (c'est-à-dire où un terme est lié à son parent par une relation *est un*). Le développement et l'utilisation de SBO sont supportés par une infrastructure créée par Mélanie Courtot et Camille Laibe (Li *et al.*, 2010a; Courtot *et al.*, 2011). La curation de l'ontologie est principalement effectuée par Nick Juty.

La *Kinetic Simulation Algorithm Ontology* (KiSAO)[17] est une ontologie d'algorithmes utilisés pour la simulation et l'analyse des modèles. Son développement a été lancé de concert avec MIASE et SED-ML en collaboration avec Dagmar Waltemath. La majeure partie de l'ontologie actuelle est due à Anna Zhukova maintenant en thèse à l'université de Bordeaux. KiSAO comprends trois branches, chacune étant principalement une hyponymie :

- **modeling and simulation algorithm** comprend les algorithmes eux-mêmes, avec des termes comme **Gillespie direct algorithm** ;
- **modeling and simulation algorithm characteristic** comprend les propriétés qui caractérisent ces algorithmes, avec des termes comme **progression with adaptive time step** ;
- **modeling and simulation algorithm parameter** liste les paramètre nécessaires à l'implémentation d'une simulation utilisant les algorithmes, avec des termes comme **maximum step size**.

Ces branches sont liées entre elles par de relations *a comme caractéristique* et *a comme paramètre*. Les algorithmes peuvent également être liés entre eux par des relations plus complexes. Par exemple, certain algorithmes hybrides font référence à plusieurs autres.

KiSAO est principalement utilisée en conjonction avec SED-ML. SED-ML comporte une classe *Simulation* contenant un attribut *kisaoID*. Cela permet d'identifier l'algorithme à utiliser dans une étape de la description. Le logiciel interprétant la description peut alors identifier au sein des algorithmes qui lui sont disponibles celui dont les caractéristiques sont les plus proches,

[17] `http://biomodels.net/kisao/`

puis récupérer la liste des paramètres qui doivent être fournis par l'utilisateur. Cette procédure est facilitée par une bibliothèque écrite en Java (Zhukova *et al.*, 2012).

MIRIAM, SED-ML et KiSAO ne concernent que la description des expériences à conduire en utilisant les modèles. Les résultats obtenus ne sont pas couverts. Une troisième étape de standardisation était donc nécessaire. Pour démarrer le développement de la *Terminology for the Description of Dynamics* (TEDDY)[18] j'ai profité de la visite de Christian Knüpfer, alors en thèse de doctorat à l'université de Jena, qui s'intéressait à la sémantique des modèles en biologie(Knüpfer *et al.*, 2013). TEDDY comprend plusieurs branches, chacune étant principalement une hyponymie :

- **temporal behaviour** décrit les dynamiques elles-mêmes, comme **periodic orbit** ;
- **behaviour characteristic** décrit les propriétés des dynamiques, comme **period** ;
- **behaviour diversification** comprend les phénomènes qui impliquent plusieurs comportements dynamiques distincts, par exemple une **bifurcation** ;
- **functional motif** collecte les structures de modèles engendrant les comportements dynamiques, comme **negative feedback loop**.

Les différentes branches de TEDDY sont reliées par tout un ensemble de relations permettant de complètement décrire une dynamique. Au sein des branches, les relations hyponimiques sont complétées par exemple par *converge vers* etc.

Les trois ontologies présentées, SBO, KiSAO et TEDDY sont décrites plus en détail dans le chapitre 5.

1.5 Représentation graphique des modèles

Les outils de normalisation présentés jusqu'ici sont destinés à être utilisés par des logiciels et non pas directement par les chercheurs. Ces derniers interagissent avec les modèles via les interfaces des logiciels. Ces interfaces sont de trois types : Des interpréteurs de texte (comme MatLab®), des interfaces à widgets (COPASI, Hoops *et al.*, 2006) et des interfaces où le modèle est manipulé de manière graphique (CellDesigner, Funahashi *et al.*, 2008). Ces dernières interfaces rejoignent la façon dont la plupart des biologistes « pensent » le modèle, et le représentent sur un tableau ou lors d'une présentation. La plupart des bases de données distribuant des voies biochimiques présentent également une interface utilisateur basée sur des graphes. Il est donc important de pouvoir interpréter rapidement ces graphes, d'une manière non-ambigüe (ou avec une ambiguïté explicite et partagée). Mais contrairement à d'autres domaines scientifiques (comme l'électronique ou la mécanique quantique), il n'y avait pas de norme bien établie pour représenter les différents types de systèmes biochimiques. Avec la montée en puissance

[18] http ://biomodels.net/teddy

de la biologie des système et de son pendant appliqué la biologie synthétique, le besoin de représentations graphiques se faisait plus pressant.

Reconnaissant ce besoin Hiroaki Kitano, maitre d'œuvre du logiciel CellDesigner (et initiateur du projet qui avait donné naissance à SBML), approcha l'agence de financement japonaise NEDO, qui lui accorda un support ainsi qu'à Mike Hucka et moi. Le projet *Systems Biology Graphical Notation* (SBGN) fut lancé officiellement en 2005, et la première réunion impliquant les différentes communautés bénéficiaires l'année suivante. Le résultat est un ensemble de langages graphiques permettant de représenter différentes "vues" d'un système d'interactions biochimique (Le Novère *et al.*, 2009) :

- Les **Process Descriptions**[19] permettent une représentation mécaniste des processus moléculaires et cellulaires, au niveau de la réaction. Ce langage est adapté à la représentation des cinétiques chimiques, et il existe une bijection entre ses symboles et les éléments de SBML Core[20] ;

- Les **Entity Relationships**[21] permettent une description de relations indépendantes les unes des autres. Ce langage est adapté à la représentation des modèles basés sur des règles (du type *kappa*, Danos & Laneve, 2004), et il peut être utilisé pour représenter les modèles encodés à l'aide du paquet SBML "Multi" [22] ;

- Les **Activity Flows**[23] sont des descriptions non-mécanistes des influences dans les réseaux de régulation biologiques. Le langage est adapté à la représentation des modèles logiques (du type utilisé dans GinSIM, Gonzalez *et al.*, 2006), et il peut être utilisé pour représenter les modèles encodés à l'aide du paquet SBML "Qual" (Chaouiya *et al.*, 2014)[24] ;

SBGN est décrite plus en détail dans le chapitre 6.

SBGN est une norme graphique. Un fichier SBGN est effectivement un dessin qui respecte la symbolique de la notation. Cependant, échanger des fichiers de pixels n'est pas d'une grande aide dans un projet de modélisation. Au mieux ce peut être utile pour interagir avec un utilisateur final, scientifique lisant un article ou explorant une banque de réseaux biochimiques. Échanger

[19] http://identifiers.org/combine.specifications/sbgn.pd.level-1.version-1.3

[20] http://identifiers.org/combine.specifications/sbml.level-3.version-1.core.release-1

[21] http://identifiers.org/combine.specifications/sbgn.er.level-1.version-1.2

[22] http://sbml.org/Documents/Specifications/SBML_Level_3/Packages/Multistate_and_Multicomponent_Species_(multi)

[23] http://identifiers.org/combine.specifications/sbgn.af.level-1.version-1.0

[24] http://sbml.org/Documents/Specifications/SBML_Level_3/Packages/Qualitative_Models_(qual)

des fichiers vectoriels permet de pouvoir modifier les graphes, mais la sémantique de la notation est perdue. C'est pourquoi nous avons développé SBGN-ML (van Iersel *et al.*, 2012)[25]. Il s'agit d'un format XML permettant d'encoder un graph suivant un des trois langages SBGN. Les fichiers SBGN-ML peuvent alors être échangés entre différents logiciels.

[25] `http://www.sbgn.org/LibSBGN/Exchange_Format`

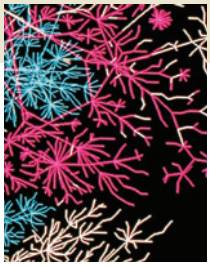
2

Directives pour l'annotation des modèles

Le Novère, N, Finney, A, Hucka, M, Bhalla, US, Campagne, F, Collado-Vides, J, Crampin, EJ, Halstead, M, Klipp, E, Mendes, P, Nielsen, P, Sauro, H, Shapiro, B, Snoep, JL, Spence, HD, Wanner, BL. Minimum information requested in the annotation of biochemical models (MIRIAM). *Nature Biotechnology* (2005), 23 : 1509-1515.

Résumé :

la plupart des modèles quantitatifs publiés en biologie sont perdus pour la communauté car soit ils ne sont pas mis librement à disposition, soit ils ne sont pas suffisamment caractérisés pour permettre leur réutilisation. L'absence d'un format de description standard, le manque de révision stricte et la négligence des auteurs sont les causes principales de descriptions incomplètes des modèles. Avec l'augmentation actuelle de l'intérêt pour les modèles biochimiques détaillées, il est nécessaire de définir une norme minimale de qualité pour l'encodage de ces modèles. Nous proposons un ensemble de règles pour la curation des modèles quantitatifs des systèmes biologiques. Ces règles définissent les procédures pour structurer et annoter les modèles représentés sous une forme lisible par un ordinateur. Nous pensons que leur application permettra aux utilisateurs (i) d'avoir confiance que ces modèles sont un reflet exact de leurs descriptions dans les documents de référence, (ii) d'interroger les collections de modèles avec précision, (iii) d'identifier rapidement les phénomènes biologiques qu'un modèle donné, ou l'une de ses composantes, représente, et (iv) de faciliter la réutilisation des modèles et leur composition dans des efforts plus larges.



Minimum information requested in the annotation of biochemical models (MIRIAM)

Nicolas Le Novère[1,15], Andrew Finney[2,15], Michael Hucka[3], Upinder S Bhalla[4], Fabien Campagne[5], Julio Collado-Vides[6], Edmund J Crampin[7], Matt Halstead[7], Edda Klipp[8], Pedro Mendes[9], Poul Nielsen[7], Herbert Sauro[10], Bruce Shapiro[11], Jacky L Snoep[12], Hugh D Spence[13] & Barry L Wanner[14]

Most of the published quantitative models in biology are lost for the community because they are either not made available or they are insufficiently characterized to allow them to be reused. The lack of a standard description format, lack of stringent reviewing and authors' carelessness are the main causes for incomplete model descriptions. With today's increased interest in detailed biochemical models, it is necessary to define a minimum quality standard for the encoding of those models. We propose a set of rules for curating quantitative models of biological systems. These rules define procedures for encoding and annotating models represented in machine-readable form. We believe their application will enable users to (i) have confidence that curated models are an accurate reflection of their associated reference descriptions, (ii) search collections of curated models with precision, (iii) quickly identify the biological phenomena that a given curated model or model constituent represents and (iv) facilitate model reuse and composition into large subcellular models.

[1]European Bioinformatics Institute, Hinxton, CB10 1SD, UK. [2]Physiomics PLC, Magdalen Centre, Oxford Science Park, Oxford, OX4 4GA,K. [3]Control and Dynamical Systems, California Institute of Technology,Pasadena, California 91125, USA. [4]National Centre for Biological Sciences, TIFR, UAS-GKVK Campus, Bangalore 560065, India. [5]Institute for Computational Biomedicine, Weill Medical College of Cornell University, New York, New York 10021, USA. [6]Center for Genomic Sciences, Universidad Nacional Autónoma de México, Av. Universidad s/n, Cuernavaca, Morelos, 62100, Mexico. [7]Bioengineering Institute and Department of Engineering Science, The University of Auckland, Private Bag 92019, Auckland, New Zealand. [8]Max-Planck Institute for Molecular Genetics, Berlin Center for Genome based Bioinformatics (BCB), Ihnestr. 73, 14195 Berlin, Germany. [9]Virginia Bioinformatics Institute, Virginia Tech, Washington St., Blacksburg, Virginia 24061-0477, USA. [10]Keck Graduate Institute, 535 Watson Drive, Claremont, California 91711, USA. [11]Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California 91109, USA. [12]Triple-J Group for Molecular Cell Physiology, Department of Biochemistry, Stellenbosch University, Private Bag X1, Matieland 7602, South Africa. [13]Department of Scientific Computing & Mathematical Modeling, GlaxoSmithKline Research & Development Limited, Medicines Research Centre, Gummels Wood Road, Stevenage, Herts, SG1 2NY, UK. [14]Purdue University, Department of Biological Sciences, Lilly Hall of Life Sciences, 915 W. State Street, West Lafayette, Indiana 47907-2054, USA. [15]These authors have contributed equally to the work. Correspondence should be addressed to N.L.N. (e-mail: lenov@ebi.ac.uk).

Published online 6 December 2005; doi:10.1038/nbt1156

During the genomic era we have witnessed a vast increase in availability of large amounts of quantitative data. This is motivating a shift in the focus of molecular and cellular research from qualitative descriptions of biochemical interactions towards the quantification of such interactions and their dynamics. One of the tenets of systems biology is the use of quantitative models (see **Box 1** for definitions) as a mechanism for capturing precise hypotheses and making predictions[1,2]. Many specialized models exist that attempt to explain aspects of the cellular machinery. However, as has happened with other types of biological information, such as sequences, macromolecular structures or

Box 1 Glossary

Some terms are used in a very specific way throughout the article. We provide here a precise definition of each one.

Quantitative biochemical model. A formal model of a biological system, based on the mathematical description of its molecular and cellular components, and the interactions between those components.

Encoded model. A mathematical model written in a formal machine-readable language, such that it can be systematically parsed and employed by simulation and analysis software without further human translation.

MIRIAM-compliant model. A model that passes all the tests and fulfills all the conditions listed in MIRIAM.

Reference description. A unique document that describes, or references the description of the model, the structure of the model, the numerical values necessary to instantiate a simulation from the model, or to perform a mathematical analysis of the model, and the results one expects from such a simulation or analysis.

Curation process. The process by which the compliance of an encoded model with MIRIAM is achieved and/or verified. The curation process may encompass some or all of the following tasks: encoding of the model, verification of the reference correspondence and annotation of the model.

Reference correspondence. The fact that the structure of a model and the results of a simulation or an analysis match the information present in the reference description.

Box 2 Case studies of MIRIAM-compliant models

To provide background to the motivations for MIRIAM, we provide here a number of case studies involving models encoded with the schemes described in this document.

User queries model database

In this scenario, a user wants to design a model of CDC2 function in the human cell-cycle. By interacting with a database consisting of models compliant with MIRIAM, this user searches all the models that contains CDC2 and represent cell cycle. Retrieving models of yeast and amphibian cell cycles, the user then reviews the models by reading the associated documentation and browsing other bioinformatics databases. By following links to databases of biochemical pathways, the user decides which model best describes what he/she knows about the function of CDC2 in the human cell cycle. The user then downloads this model and uses it as a basis for her/his own modeling work.

All of the above is possible if this proposal is applied, ensuring that models correspond to associated reference descriptions and are appropriately annotated.

Journal peer review: JWS Online

In this use case, we describe how a journal peer review process could incorporate MIRIAM, using the example of the procedure carried out by JWS Online[9] with its associated journals. When a manuscript describing a kinetic model is submitted to those journals, the authors are requested to submit the model description in electronic form (encoded in an accessible standard format). A curator parses the model using software that automatically checks its syntax (for instance, SBML and CellML validation tools), and if necessary, corrects the model. The curator then performs the verifications described in the section on reference correspondence. In particular, he/she attempts to reproduce the model results, as shown in the manuscript. If this fails, the curator contacts the authors in an attempt to correct the errors in the description or coding. After the curators and authors reach agreement on model description and simulation results, the model is made available to the reviewers and the authors, in a secure manner. A letter is sent to the reviewers with a set of instructions on how they can test the model remotely, running simulations at JWS Online directly from their web browsers. If the manuscript is accepted by the journal for publication, the model is moved to the public database of JWS Online. Some of the benefits of the procedure are:

- Readers would not have to re-encode models into an accessible format based on the article.
- The reviewers and authors could resolve issues relating to the correspondence between the encoded model and the model described in the article, before publication. Any differences could be eliminated.
- Modelers would be motivated to resolve correspondence issues because the publication of their article would depend on it.

Curation pipeline

The model curation process requires significant effort and this effort will in practice be shared between curators and/or teams of curators, often at different sites. The subdivision of MIRIAM into components is useful for defining the relationships between these individuals and groups. We anticipate that some groups will concentrate on encoding models that comply with the proposal for reference correspondence and the attribution scheme for annotations. Other groups will then continue the curation process by annotating these models so that they comply with the external data resources annotation scheme.

microarray data, quantitative models will be useful only if their access and reuse is made easy for all scientists. Moreover, the next step towards a more synergistic view of living systems is assembling models into larger entities, by module reuse and assembly or modeling across different spatial, temporal or physiological scales. Both model retrieval and model composition require formal descriptions of model structure and semantics. Our separate groups have been active in the development of standards for encoding biological models in machine-readable formats (e.g., CellML[3] and SBML[4,5]) and of public repositories of computational models (such as BioModels Database[6], Sigpath[7], EcoCyc[8], the CellML repository (http://www.cellml.org/examples/repository/), JWS Online[9], RegulonDB[10], DOQCS[11]). We firmly believe in the value of expressing computational models using standardized, structured formats as a means of enabling direct interpretation and manipulation of those models by software tools.

Box 3 Rules for reference correspondence

1. The model must be encoded in a public, machine-readable format, either standard such as SBML or CellML, or supported by specific software applications. Relevant examples include those aimed at biological modeling (GENESIS[44], XPP[45]) or generic scientific software packages (Mathematica, MatLab, SciLab, Octave)
2. The encoded model must comply with the standard in which it is encoded. The syntax of the language must be respected, and the model has to pass validation at curation time. The form of this validation will depend on the format in which the model is encoded. For the SBML and CellML standards, formal validation software should be used; see http://sbml.org/ and http://www.cellml.org/, respectively. For application-specific formats, the model must be parsed (loaded) successfully by the relevant application.
3. The model must be clearly related to a single reference description that describes or references a set of results that one can expect to reproduce using the model. If the model is associated with only part of a reference description, then that part must be clearly identified (although failure to do so does not preclude MIRIAM compliance). If a model is derived from several initial reference descriptions, there must still be a reference description associated with the derived/combined model.
4. The encoded model structure must reflect the biological processes listed in the reference description. For instance, one should be able to map a reaction network in the encoded form to a reaction graph in the associated description. It is not essential that the constituents of the encoded model correspond one-to-one with the constituents described in the associated reference description. The software used to build

Box 3 Rules for reference correspondence (continued)

the initial model and the standard format used to encode the model may impose constraints on the form of the model. For example, a modeler might have to add reactions to represent the creation or removal of mass. A ligand in excess may be represented either as an independent constituent, or as an event modifying parameters.

5. The encoded model must be instantiated in a simulation. This means that quantitative attributes of the model have to be defined. Therefore, the model must contain, or be associated with, values (or ranges of values) for all initial conditions and parameters, as well as kinetic expressions for all reactions. These values can be provided as a separate file from the model itself. If the model was not submitted as an adjunct to the original description, then one should be able to trace all quantities in the encoded form to quantities enumerated in the reference description. The values of quantitative variables and their units must be equivalent to the values listed in the reference description. Any missing values have to be added (perhaps by contacting the authors) before the model can be claimed to be MIRIAM compliant
6. The model, when instantiated within a suitable simulation environment, must be able to reproduce all relevant results given in the reference description that can readily be simulated. Not only does the simulation have to provide results qualitatively similar to the reference description, such as oscillation, bistability, chaos, but the quantitative values of variables, and their relationships (e.g., the shape of the phase portrait) must be reproduced within some epsilon, the difference being attributable to the algorithms used to run the simulation, and the roundup errors. Some software exists that can help to compare qualitatively the results of a simulation with a benchmark; see for instance BIOCHAM[46].

Databases of quantitative models are valuable resources only if researchers can trust the quality of their content. Similarly, repositories are not useful unless users can search for specific models and then relate model constituents to other data sets such as bioinformatics databases and controlled vocabularies. To meet these needs, we believe four complementary aspects of the quality of an encoded model must be addressed: (i) the quality of the documentation (e.g., journal article) associated with the encoded model, (ii) the degree of correspondence between the encoded model and the documentation, (iii) the accuracy and extent of the annotations of the encoded model and (iv) whether the model is encoded in a machine-readable format, that is, a format that can be immediately and unambiguously parsed by software to perform simulations and analysis.

Most of the encoded models available in scientific publications or on the Internet are not in a standard format. Of those that are encoded in a standard format, it turns out that most actually fail compliance tests developed for these standards. Failures occur for a variety of reasons, ranging from minor syntactic errors to significant conceptual problems, including the incorrect specification of units. Even deeper semantic inaccuracies can lie in the structure of the model itself. Finally, there is no standard naming scheme for the model constituents, so the precise identification of constituents depends on the associated documentation/annotation. Most models available today are not annotated, and as a result, users are faced with such things as a reaction 'X' between the constituents 'A' and 'B,' producing 'C' and modulated by 'M.' As a consequence, models frequently have to be re-encoded in order to be reused, a process that in practice is often performed by a different person from the original author.

These quality issues must be addressed when curating model collections for public use, just as it is done for other type of biological data. One crucial step is the development of interchange standards[12], such as those developed for microarray data[13], protein interactions[14] or metabolic analyses[15]. By 'curation,' we mean the processes of collecting models, verifying them to some degree and annotating them with metadata.

Box 4 Annotation that must be included with a quantitative model to achieve MIRIAM compliance

1. The preferred name of the model, in order to facilitate discussions about it.
2. A citation of the reference description with which the model is associated. This citation can be a complete bibliographic record, a unique identifier such as a Digital Object Identifier (http://www.doi.org/), a PubMed identifier (http://www.pubmed.gov/) or, in the last resort, an unambiguous URL pointing to the description itself (but not a generic URL, for instance of an archive containing the description). The main point is that the citation should provide access to the complete description of the model and should make possible the identification of the authors of the reference description. These authors should be contacted if there are concerns with the biological basis of the model (such as the presence of an interaction undocumented in the scientific literature).
3. Name and contact information for the model creators, that is, the people who actually contributed to the encoding of the model in its present form. In many cases, there will be many creators who either encoded the model from scratch, or debugged it. For instance, the semantic curators of a database would be creators. The creators should be contacted if there are problems with the structure of the model (initial conditions, kinetics parameters, reaction scheme).
4. The date and time of creation, and the date and time of last modification. This is particularly important in order to know if a model has been modified since its creation, and to compare various versions of the same model. A history of the modifications could be useful, but is not required for MIRIAM compliance. A checksum could be useful to identify a specific version of a model, but is not required for MIRIAM compliance.
5. A precise statement about the terms of distribution. The statement can be anywhere from 'public domain' to 'copyrighted' and 'freely distributable' to 'confidential.' It is important to note that MIRIAM itself does not require free distribution, whether in the sense of 'freedom of use' or 'no cost.' However, MIRIAM is intended to allow models to be communicated better, and stipulating the terms of distribution are essential for that purpose.

Table 1 Possible sources of annotation for different model constituents[a]

Constituent	Resources
Model	Digital Object Identifier, Medline, PubMed, Gene Ontology[30] (BP, MF, CC), International Classification of Disease, Online Mendelian Inheritance in Man[31] (OMIM), Taxonomy[32,33]
Physical compartment	Gene Ontology (CC), Taxonomy
Reacting entity	BIND complex, Chemical Entities of Biological Interest (ChEBI), Ensembl[29], Gene Ontology (MF, CC), InterPro[34], KEGG[35] compound, OMIM, Protein DataBank (PDB), PIRSF[36], Reactome[37], UniProt[28] ...
Reaction	BIND interaction[38], EC code, Gene Ontology (BP, MF), KEGG reaction, IntAct[39]

[a]This list is by no means exhaustive, but rather represents the diversity of available resources. BP, biological process; MF, molecular function; CC, cellular component.

We propose to standardize an approach to the curation of model collections and the encoding of models using a framework of rules we call MIRIAM, the Minimum Information Requested In the Annotation of Models. MIRIAM aims to define processes and schemes that will instill confidence in model collections, enable the assembly of meta-collections of models at the same high level of quality and allow the curation process to be shared among teams at different sites and institutions. The standard we propose is designed to cover encoding processes that may be conducted either up front by the model author or *post hoc* by a curator. However, we do not believe that the *post hoc* approach is particularly efficient, and prefer modelers to make their models available in standard formats. **Box 2** describes some uses of MIRIAM.

Scope of MIRIAM

MIRIAM applies only to models linked to a unique reference description. MIRIAM does not address directly issues of quality of documentation (although sufficiently poor documentation can make a model impossible to curate). The assessment of the quality of documentation is well established in the scientific community. We expect that, by assessing the documentation describing quantitative models, peer reviewers (not the model curators) will assess the models' ability to represent and predict the quantitative behavior of biological systems and/or make an important theoretical contribution. Instead, MIRIAM focuses on the correspondence of an encoded model to its associated description and how the encoded model is annotated. In other words, even if it is MIRIAM compliant, a model may not necessarily make sense in biological terms. Conversely, many models that cannot be declared MIRIAM compliant may still be of high scientific interest.

We expect MIRIAM to apply mainly to quantitative models that can be simulated over a range of parameter values and provide numerical results. This encompasses not only models that can be integrated or iterated forwards in time, such as ordinary and partial differential equation models and differential algebraic equation models, but also other quantitative approaches such as steady-state models (e.g., Metabolic Control Analysis[16], Flux Balance Analysis[17]). Discrete approaches, such as logical modeling[18–20] or stochastic and hybrid Petri Net[21], can also be considered when they can lead to specific numerical results. Although we are aware that this means we can cover only part of the modeling field, we make this our initial focus because only these models can lead to quantitative numerical results providing refutable predictions. The comparison of these predictions with the reference description of the model is a crucial test of MIRIAM compliance.

Overview of the proposal

MIRIAM is divided into two parts. The first is a proposed standard for reference correspondence dealing with the syntax and semantics of the model, whereas the second is a proposed annotation scheme that specifies the documentation of the model by external knowledge.

Standard for reference correspondence

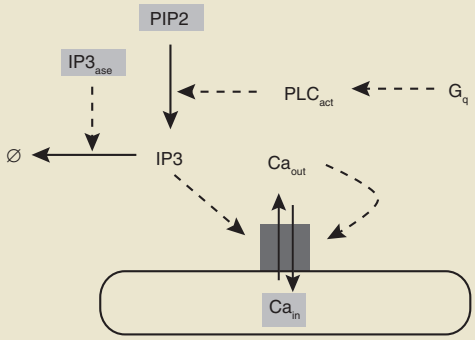
The aim of this proposal is to ensure that the model is properly associated with a reference description and is consistent with that reference description. To be declared MIRIAM compliant, a quantitative model must fulfill a set of rules dealing with its encoding, its structure and the results it should provide when instantiated in simulations. These rules are detailed in **Box 3**.

Table 2 Examples of different physical locations related to the same URIs expressed as a URL or a LSID

URI	Example of alternative physical locations
Taxonomy	
http://www.ncbi.nlm.nih.gov/Taxonomy/#9606	http://www.ncbi.nlm.nih.gov/Taxonomy/Browser/wwwtax.cgi?id=9606 (ref. 32)
urn:lsid:ncbi.nlm.nih.gov:Taxonomy:9606	http://www.ebi.ac.uk/newt/display?search=9606 (ref. 33)
Gene Ontology	
http://www.geneontology.org/#GO:0045202	http://www.ebi.ac.uk/ego/DisplayGoTerm?id=GO:0045202
urn:lsid:geneontology.org:GO:0045202	http://www.godatabase.org/cgi-bin/amigo/go.cgi?view=details&query=GO:0045202
UniProt	
http://www.uniprot.org/#P62158	http://www.ebi.uniprot.org/entry/P62158 (ref. 28)
urn:lsid:uniprot.org:P62158	http://us.expasy.org/uniprot/P62158 (ref. 40)
	http://www.pir.uniprot.org/cgi-bin/upEntry?id=P62158 (ref. 41)
EC code	
http://www.ebi.ac.uk/intenz/EC 1.1.1.1	http://www.ebi.ac.uk/intenz/query?cmd=SearchEC&ec=1.1.1.1 (ref. 42)
urn:lsid:ebi.ac.uk:intenz:EC 1.1.1.1	http://www.genome.jp/dbget-bin/www_bget?ec:1.1.1.1 (ref. 35)
	http://www.chem.qmul.ac.uk/iubmb/enzyme/EC1/1/1/1.html
	http://us.expasy.org/cgi-bin/nicezyme.pl?1.1.1.1 (ref. 43)

Table 3 Example of a small curated and annotated model

Creators	Joe User (juser@eden.com), Anne Other (aother@eden.com)
Creation date	01 January 2000
Last modification	31 May 2005



Constituent	Data type	Identifier	Qualifier	Meaning
Model	http://www.pubmed.gov/	0000000		
	http://www.ncbi.nlm.nih.gov/Taxonomy/	9606		*Homo sapiens*
	http://www.geneontology.org/	GO:0007204	IsVersionOf	Positive regulation of cytosolic [Ca^{2+}]
	http://www.geneontology.org/	GO:0051279	IsVersionOf	Regulation of release of sequestered Ca^{2+} into cytoplasm
	http://www.genome.jp/kegg/pathway/	hsa04020	IsPartOf	Calcium signaling pathway, *H. sapiens*
	http://www.genome.jp/kegg/pathway/	hsa04070	IsPartOf	Phosphatidylinositol signaling system, *H sapiens*
Compartment ER	http://www.geneontology.org/	GO:0005790		Smooth endoplasmic reticulum
Reactant Ca_{in}	http://www.ebi.ac.uk/chebi/	CHEBI:29108		$Calcium^{2+}$
Cytoplasm	http://www.geneontology.org/	GO:0005737		Cytoplasm
Reactant Ca_{out}	http://www.ebi.ac.uk/chebi/	CHEBI:29108		$Calcium^{2+}$
Reactant IP3	http://www.ebi.ac.uk/chebi/	CHEBI:16595		1D-myo-inositol 1,4,5-tris (dihydrogen phosphate)
Reactant PIP2	http://www.ebi.ac.uk/chebi/	CHEBI:18348		1-phosphatidyl-1D -myo-inositol 4,5-bisphosphate
Reactant IP3R	http://www.uniprot.org/	Q14643	HasVersion	Inositol 1,4,5-trisphosphate receptor type 1
	http://www.uniprot.org/	Q14571	HasVersion	Inositol 1,4,5-trisphosphate receptor type 2
	http://www.uniprot.org/	Q14573	HasVersion	Inositol 1,4,5-trisphosphate receptor type 3
Reactant PLC_{act}	http://www.uniprot.org/	Q9NQ66	IsVersionOf	PIP2 phosphodiesterase β-1
Reactant PLC_{tot}	http://www.uniprot.org/	Q9NQ66		PIP2 phosphodiesterase β-1
Reactant $IP3_{ase}$	http://www.uniprot.org/	Q14642		Type I inositol-1,4,5-trisphosphate 5-phosphatase
Reactant G_q	http://www.uniprot.org/	Q6NT27		Guanine nucleotide binding protein Gq
Reaction $Ca_{release}$	http://www.geneontology.org/	GO:0005220		IP3-sensitive calcium-release channel activity
	http://www.geneontology.org/	GO:0008095	IsVersionOf	IP3 receptor activity
Reaction $IP3_{production}$	http://www.geneontology.org/	GO:0004435	IsVersionOf	Phosphoinositide phospholipase C activity
	http://www.ebi.ac.uk/intenz/	3.1.4.11	IsVersionOf	Phosphoinositide phospholipase C
Reaction $IP3_{degradation}$	http://www.ebi.ac.uk/intenz/	3.1.3.56	IsVersionOf	Inositol-polyphosphate 5-phosphatase
Reaction $PLC_{activation}$	http://www.geneontology.org/	GO:0007200		G-protein signaling coupled to IP3 second messenger

$$k_1 = k_2 = k_3 = 1\ s^{-1}$$

$$Km = 10^{-7}\,M, Km_2 = 10^{-8}\,\mathrm{M}, Km_3 = 2.10^{-6}\,M$$

$$K_A = 10^{-11}, m = 4, n = 3, \alpha = 0.001$$

$$[Ca_{in}] = [IP3R] = [PLC_{tot}] = [PIP2] = [IP3_{ase}] = 0.001\,M$$

$$[G_q] = 0.01\,M,\ [Ca_{out}] = [IP3] = [PLC_{act}] = 0\,M$$

$$\frac{d[Ca_{out}]}{dt} = \frac{k_1[IP3R] * ([Ca_{in}] - [Ca_{out}])}{Km_1 + |[Ca_{in}] - [Ca_{out}]|} * \frac{[IP3]^m}{K_A + [IP3]^m}$$

$$\frac{d[IP3]}{dt} = \frac{k_2[PLC_{act}] * [PIP2]}{Km_2 + [PIP2]} - \frac{k_3[IP3_{ase}] * [IP3]}{Km_3 + [IP3]}$$

$$\frac{d[PLC_{act}]}{dt} = \frac{[G_q]^n}{\alpha + [G_q]^n} * [PLC_{tot}]$$

The model describes the release of calcium from the endoplasmic reticulum, regulated by cytoplasmic calcium and the Inositol 1,4,5-trisphosphate produced in response to G-protein–coupled receptor activation. Note that although working, this model is only meant to provide a large number of example annotations.

To pass the various tests, and in particular the reproduction of described results, a modeler could be required to make minor changes to a model until it is truly consistent with the results given in the associated reference description. If the modeler is not one of the authors, ideally he/she should perform these modifications in collaboration with the authors. Examples include changing a few parameter and/or initial condition values.

When the model given in the text of the reference description is significantly different from the encoded model used to generate the results given in this text, the model cannot be curated and MIRIAM cannot be applied. For example, MIRIAM cannot be applied if a significant number of parameter values are different between the two models (the significance being judged by the curators). The original authors of the model should be encouraged to publish an erratum detailing the correct values.

Annotation schemes

The scheme for annotation is composed of two complementary components: attribution, covering the absolute minimum information that is required to associate the model with both a reference description and an encoding process, and external data resources, covering information required to relate the constituents of quantitative models to established data resources or controlled vocabularies.

The annotations must always be transferred with the encoded model. The ideal case is incorporating these annotations in the same file as the model itself, in a structured form such as the CellML metadata[22] or the SBML simple annotation scheme[23]. However, annotations could also be joined in another form, such as one or several accompanying files, in various formats, textual or graphical.

Attribution annotation

To be confident in being able to reuse an encoded model, one must be able to trace its origin and the people who were involved in its creation. In particular, the reference description has to be identified, as well as the authors and creators of the model. The information that must always be joined with an encoded model is listed in **Box 4**.

External data resources annotation

The aim of this scheme is to link model constituents to corresponding structures in existing and future open access bioinformatics resources. Such data resources can be, for instance, database or controlled vocabularies. This will permit the identification of model constituents and the comparison of model constituents between different models, but also the execution of queries on models to recover specific constituents in models. Possible sources of annotation for various types of constituents are listed in **Table 1**.

This annotation must permit a piece of knowledge to be unambiguously related to a model constituent. The structure of an atomic element of the annotation is similar to the relationshipXref element of BioPAX (http://www.biopax.org/). The referenced information should be described using a triplet {"data-type," "identifier," "qualifier"}. The "data-type" is a unique, controlled description of the type of data. The "identifier," within the context of the "data-type," points to a specific piece of knowledge. The "qualifier" is a string that serves to refine the relation between the referenced piece of knowledge and the described constituent. Example of qualifiers are "has a," "is version of," "is homolog to." The qualifier is optional, and its absence does not preclude MIRIAM compliance. When a qualifier is absent, one assumes the relation to be "is."

The "data-type" should be written as a Unique Resource Identifier[24]. This URI can be a Uniform Resource Locator[25] or a Uniform Resource Name[26]. The URL or URN does not have to describe an actual physical location. It is up to the software tool reading the model to decide what to do with this URI. This software can, for instance, use the "identifier" with a search engine built on a database mirroring the "data-type." Alternatively, a reading tool translating the model can build a hyperlink using the "identifier" and another URL related to the "data-type."

The "data-type" and the "identifier" can be combined into a single URL, such as http://www.myResource.org/#myIdentifier or as a URN, for instance using the LSID scheme[27] of urn:lsid:myResource.org:myIdentifier.

To enable interoperability, the community will have to agree on a set of standard, valid URIs. An online resource will be established to catalog the URIs and the corresponding physical URLs of the agreed-upon "data-types," whether these are controlled vocabularies or databases. This catalog will simply list the URIs and for each one, provide a corresponding summary of the syntax for the "identifier." An application programming interface (API) can be created so that software tools can retrieve valid URL(s) corresponding to a given URI. **Table 2** shows a small subset of this forthcoming list. Note that although MIRIAM compliance does not require such a list to exist, it is considered crucial to actually enforce MIRIAM usage, and to make it truly useful. The list will also have to evolve with the data resources.

It is important that model constituents be annotated with perennial identifiers. For example, the "entry name" field of UniProt[28] is not perennial but is modified on a regular basis to reflect the classification of the protein. However, the "accession" field of UniProt is perennial. Consider a model with an entity representing the protein calmodulin. An annotation of this entity referring to the UniProt record for calmodulin should therefore use a URI containing the "accession" field value for calmodulin "P62158" rather than the "entry name" field value "CALM_HUMAN."

Quite often, several identified biological entities, physical components or reactions are lumped in a single constituent of the model. For instance, successive reactions of a pathway may be merged into one reaction, or a set of different molecules is represented by one pool. The annotation must reflect this situation, either by enumerating the biological entities, or with a carefully chosen term from a controlled vocabulary (an example of a curated and annotated model is presented in **Table 3**).

Conclusions

We believe that through the standardization of the model curation process, it will be possible to create resources that are as significant to systems biology as resources like Ensembl[29] are to genomics. Pursuing this proposal will in the short term allow us to establish collections of models of sufficient quality to gain the confidence of the systems biology community. To pave the way, the resources handled by the authors of this manuscript (BioModels Database, CellML repository, DOQCS, SigPath) endorse the standard, and will undertake efforts to make them MIRIAM compliant. In the longer term, the application of MIRIAM will enable the peer review process to become more efficient and its products more accessible. We also hope the standard will be adopted by publishers of scientific literature, as was the case with other standards such as MIAME[13].

COMPETING INTERESTS STATEMENT
The authors declare that they have no competing financial interests.

Published online at http://www.nature.com/naturebiotechnology/
Reprints and permissions information is available online at http://npg.nature.com/reprintsandpermissions/

1. Kitano, H. Computational systems biology. *Nature* **420**, 206–210 (2002).
2. Crampin, E. *et al.* Computational physiology and the physiome project. *Exp. Physiol.* **89**, 1–26 (2004).

3. Lloyd, C., Halstead, M. & Nielsen, P. CellML: its future, present and past. *Prog. Biophys. Mol. Biol.* **85**, 433–450 (2004).
4. Hucka, M. *et al.* The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics* **19**, 524–531 (2003).
5. Finney, A. & Hucka, M. Systems biology markup language: level 2 and beyond. *Biochem. Soc. Trans.* **31**, 1472–1473 (2003).
6. Le Novère, N., *et al.* BioModels Database: A free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. *Nucleic Acids Res.* **34**, (2006).
7. Campagne, F. *et al.* Quantitative information management for the biochemical computation of cellular networks. *Sci. STKE* **248**, PL11 (2004).
8. Keseler, I. *et al.* EcoCyc: a comprehensive database resource for *Escherichia coli*. *Nucleic Acids Res.* **33**, D334–D337 (2005).
9. Olivier, B. & Snoep, J. Web-based kinetic modelling using JWS Online. *Bioinformatics* **20**, 2143–2144 (2004).
10. Salgado, H. *et al.* RegulonDB (version 4.0): transcriptional regulation, operon organization and growth conditions in *Escherichia coli* K-12. *Nucleic Acids Res.* **32**, D303–D306 (2004).
11. Sivakumaran, S., Hariharaputran, S., Mishra, J. & Bhalla, U. The database of quantitative cellular signaling: management and analysis of chemical kinetic models of signaling networks. *Bioinformatics* **19**, 408–415 (2003).
12. Quackenbush, J. Data standards for 'omic' science. *Nat. Biotechnol.* **22**, 613–614 (2004).
13. Brazma, A. *et al.* Minimum information about a microarray experiment (MIAME)-toward standards for microarray data. *Nat. Genet.* **29**, 365–371 (2001).
14. Hermjakob, H. *et al.* The HUPO PSI's molecular interaction format-a community standard for the representation of protein interaction data. *Nat. Biotechnol.* **22**, 177–183 (2004).
15. Lindon, J. *et al.* Summary recommendations for standardization and reporting of metabolic analyses. *Nat. Biotechnol.* **23**, 833–838 (2005).
16. Kacser, H. & Burns, J. The control of flux. *Symp. Soc. Exp. Biol.* **27**, 65–104 (1973).
17. Savinell, J. & Palsson, B. Optimal selection of metabolic fluxes for *in vivo* measurement. I. Development of mathematical methods. *J. Theor. Biol.* **155**, 201–214 (1992).
18. Thomas, R. Boolean formalisation of genetic control circuits. *J. Theor. Biol.* **42**, 565–583 (1973).
19. Sánchez, L. & Thieffry, D. Segmenting the fly embryo: a logical analysis of the pair-rule cross-regulatory module. *J. Theor. Biol.* **224**, 517–537 (2003).
20. Laubenbacher, R. & Stigler, B. A computational algebra approach to the reverse engineering of gene regulatory networks. *J. Theor. Biol.* **229**, 523–537 (2004).
21. Doi, A., Fujita, S., Matsuno, H., Nagasaki, M. & Miyano, S. Constructing biological pathway models with hybrid functional petri nets. *In Silico Biol.* **4**, 271–291 (2003).
22. Cuellar, A., Nelson, M. & Hedley, W. The CellML metadata 1.0 specification. http://www.cellml.org/specifications/metadata/.
23. Le Novère, N. & Finney, A. A simple scheme for annotating SBML with references to controlled vocabularies and database entries. http://www.ebi.ac.uk/compneur-srv/sbml/proposals/AnnotationURI.pdf.
24. Berners-Lee, T., Fielding, R. & Masinter, L. Uniform resource identifier (URI): Generic syntax. http://www.gbiv.com/protocols/uri/rfc/rfc3986.html.
25. Berners-Lee, T. Uniform resource locators (URL): a syntax for the expression of access information of objects on the network. http://www.w3.org/Addressing/URL/url-spec.txt.
26. Moats, R. URN syntax. http://www.ietf.org/rfc/rfc2141.txt.
27. Martin, S., Niemi, M. & Senger, M. Life sciences identifiers RFP response. http://www.omg.org/technology/documents/formal/life_sciences.htm
28. Apweiler, R. *et al.* UniProt: the universal protein knowledgebase. *Nucleic Acids Res.* **32**, D115–D119 (2004).
29. Hubbard, T. *et al.* The Ensembl genome database project. *Nucleic Acids Res.* **30**, 38–41 (2002).
30. Ashburner, M. *et al.* Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat. Genet.* **25**, 25–29 (2000).
31. Hamosh, A., Scott, A., Amberger, J., Bocchini, C. & McKusick, V. Online mendelian inheritance in man ({OMIM}), a knowledgebase of human genes and genetic disorders. *Nucleic Acids Res.* **33**, D514–D517 (2005).
32. Wheeler, D. *et al.* Database resources of the national center for biotechnology information. *Nucleic Acids Res.* **28**, 10–14 (2000).
33. Phan, I., Pilbout, S., Fleischmann, W. & Bairoch, A. NEWT, a new taxonomy portal. *Nucleic Acids Res.* **31**, 3822–3823 (2003).
34. Mulder, N.J. *et al.* InterPro, progress and status in 2005. *Nucleic Acids Res.* **33**, 201–205 (2005).
35. Kanehisa, M., Goto, S., Kawashima, S., Okuno, Y. & Hattori, M. The KEGG resource for deciphering the genome. *Nucleic Acids Res.* **32**, D277–D280 (2004).
36. Wu, C. *et al.* PIRSF: family classification system at the protein information resource. *Nucleic Acids Res.* **32**, D112–D114 (2004).
37. Joshi-Tope, G. *et al.* The genome knowledgebase: A resource for biologists and bioinformaticists. *Cold Spring Harb. Symp. Quant. Biol.* **68**, 237–243 (2003).
38. Bader, G. & Hogue, C. BIND—a data specification for storing and describing biomolecular interactions, molecular complexes and pathways. *Bioinformatics* **16**, 465–477 (2000).
39. Hermjakob, H. *et al.* IntAct—an open source molecular interaction database. *Nucleic Acids Res.* **32**, D452–D455 (2004).
40. Boeckmann, B. *et al.* The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. *Nucleic Acids Res.* **31**, 365–370 (2003).
41. Wu, C. *et al.* Update on genome completion and annotations: protein information resource. *Nucleic Acids Res.* **31**, 345–347 (2003).
42. Fleischmann, A. *et al.* IntEnz, the integrated relational enzyme database. *Nucleic Acids Res.* **32**, D434–D437 (2004).
43. Bairoch, A. The ENZYME database in 2000. *Nucleic Acids Res.* **28**, 304–305 (2000).
44. Bower, J. & Beeman, D. *The Book of GENESIS* (Springer-Verlag, New York, 1998).
45. Ermentrout, B. *Simulating, Analyzing, and Animating Dynamical Systems: A Guide to XPPAUT for Researchers and Students* (Society for Industrial & Applied Math, Philadelphia, PA, 2002).
46. Chabrier, N. & Fages, F. Symbolic model checking of biochemical networks. in *International Workshop on Computational Methods in Systems Biology* (Springer-Verlag, New York, 2003).

3

Directives pour la description des simulations

Waltemath D, Adams R, Beard DA, Bergmann FT, Bhalla US, Britten R, Chelliah V, Cooling MT, Cooper J, Crampin E, Garny A, Hoops S, Hucka M, Hunter P, Klipp E, Laibe C, Miller A, Moraru I, Nickerson D, Nielsen P, Nikolski M, Sahle S, Sauro HM, Schmidt H, Snoep JL, Tolle D, Wolkenhauer O, Le Novère N. Minimum Information About a Simulation Experiment (MIASE). *PLoS Computational Biology* (2011), 7(4) : e1001122

Résumé :

La reproductibilité des expériences est une exigence de base en recherche scientifique. Les directives minimales (MI) en biologie se sont montrées efficaces pour faciliter la réutilisation de travaux existants. MIRIAM promeut l'échange et la réutilisation des modèles informatiques en biologie. Cependant, l'information relative au modèle n'est pas suffisante par elle-même pour permettre sa réutilisation efficace. Les algorithmes numériques avancés ainsi que les workflows complexes de modélisation utilisés de nos jours en biologie rendent la reproduction des simulations difficiles. Il est de ce fait essentiel de définir l'information clé nécessaire pour simuler ces modèles. L'information minimale pour la description des expériences de simulation (MIASE) décrit l'ensemble minimal d'information qui doit être fournie afin de rendre la description d'une expérience de simulation utilisable par d'autres chercheurs. Elle comprend la liste des modèles et leurs modifications, toutes les procédures de simulation à appliquer et leur ordre, le traitement des informations numériques brutes, et la description de la sortie numérique finale. MIASE est applicable à toute expérience de simulation. Cette information, plus l'ensemble des modèles requis, garantit que l'expérience de simulation représente l'intention des auteurs initiaux. Le respect des directives MIASE améliorera la qualité de la publication scientifique, et permettra des efforts collaboratifs, plus distribués, autour de la modélisation et de la simulation des processus biologiques.

OPEN ACCESS Freely available online

PLoS COMPUTATIONAL BIOLOGY

Perspective

Minimum Information About a Simulation Experiment (MIASE)

Dagmar Waltemath[1], Richard Adams[2,3], Daniel A. Beard[4], Frank T. Bergmann[5,6], Upinder S. Bhalla[7], Randall Britten[8], Vijayalakshmi Chelliah[9], Michael T. Cooling[8], Jonathan Cooper[10], Edmund J. Crampin[8], Alan Garny[11], Stefan Hoops[12], Michael Hucka[13], Peter Hunter[8], Edda Klipp[14], Camille Laibe[9], Andrew K. Miller[8], Ion Moraru[15], David Nickerson[8], Poul Nielsen[8], Macha Nikolski[16], Sven Sahle[17], Herbert M. Sauro[5], Henning Schmidt[18,19], Jacky L. Snoep[20], Dominic Tolle[9], Olaf Wolkenhauer[18], Nicolas Le Novère[9]*

1 Database and Information Systems, Graduate Research School dIEM oSiRiS, Rostock University, Rostock, Mecklenburg-Vorpommern, Germany, **2** Centre for Systems Biology at Edinburgh, University of Edinburgh, Edinburgh, United Kingdom, **3** Informatics Life-Sciences Institute, School of Informatics, University of Edinburgh, Edinburgh, United Kingdom, **4** Biotechnology and Bioengineering Center, Department of Physiology, Medical College of Wisconsin, Milwaukee, Wisconsin, United States of America, **5** Department of Bioengineering, University of Washington, Seattle, Washington, United States of America, **6** Keck Graduate Institute, Claremont, California, United States of America, **7** National Centre for Biological Sciences, Tata Institute of Fundamental Research, Bangalore, India, **8** Auckland Bioengineering Institute, The University of Auckland, Auckland, New Zealand, **9** EMBL-EBI, Wellcome-Trust Genome Campus, Hinxton, United Kingdom, **10** Oxford University Computing Laboratory, University of Oxford, Oxford, United Kingdom, **11** Cardiac Electrophysiology Group, Department of Physiology, Anatomy and Genetics, University of Oxford, Oxford, United Kingdom, **12** Virginia Bioinformatics Institute, Virginia Polytechnic Institute and State University, Blacksburgh, Virginia, United States of America, **13** Engineering and Applied Science, The California Institute of Technology, Pasadena, California, United States of America, **14** Theoretical Biophysics, Humboldt Universität zu Berlin, Berlin, Germany, **15** Department of Cell Biology, University of Connecticut Health Center, Farmington, Connecticut, United States of America, **16** Laboratoire Bordelais de Recherche en Informatique, Universite Bordeaux 1, Bordeaux, France, **17** BIOQUANT, University of Heidelberg, Heidelberg, Germany, **18** Systems Biology & Bioinformatics Group, University of Rostock, Rostock, Germany, **19** Novartis Pharma AG, Novartis Campus, Basel, Switzerland, **20** Department of Biochemistry, Stellenbosch University, Matieland, South Africa

Reproducibility of experiments is a basic requirement for science. Minimum Information (MI) guidelines have proved a helpful means of enabling reuse of existing work in modern biology. The Minimum Information Required in the Annotation of Models (MIRIAM) guidelines promote the exchange and reuse of biochemical computational models. However, information about a model alone is not sufficient to enable its efficient reuse in a computational setting. Advanced numerical algorithms and complex modeling workflows used in modern computational biology make reproduction of simulations difficult. It is therefore essential to define the core information necessary to perform simulations of those models. The Minimum Information About a Simulation Experiment (MIASE, Glossary in Box 1) describes the minimal set of information that must be provided to make the description of a simulation experiment available to others. It includes the list of models to use and their modifications, all the simulation procedures to apply and in which order, the processing of the raw numerical results, and the description of the final output. MIASE allows for the reproduction of any simulation experiment. The provision of this information, along with a set of required models, guarantees that the simulation experiment represents the intention of the original authors. Following MIASE guidelines will thus improve the quality of scientific reporting, and will also allow collaborative, more distributed efforts in computational modeling and simulation of biological processes.

Needs for a Standard Description of Simulations Experiments

The rise of systems biology as a new paradigm of biological research has put computational modeling under the spotlight. In cell biology [1], physiology [2], and more recently in synthetic biology [3], mathematical modeling and simulation have become parts of a researcher's toolkit. Following Cellier [4], we consider "a model (M) for a system (S) and an experiment (E) is anything to which E can be applied in order to answer questions about S" and "a simulation is an experiment performed on a model". Zeigler [5] emphasized the importance of separating the descriptions of the experimental frame (e.g., the initial conditions), the model, and the simulation.

Although generic, this framework for modeling and simulation applies well to the field of computational modeling and simulation of biological processes, where

Citation: Waltemath D, Adams R, Beard DA, Bergmann FT, Bhalla US, et al. (2011) Minimum Information About a Simulation Experiment (MIASE). PLoS Comput Biol 7(4): e1001122. doi:10.1371/journal.pcbi.1001122

Editor: Philip E. Bourne, University of California San Diego, United States of America

Published April 28, 2011

Funding: The discussions that led to the definition of MIASE benefited from the support of a Japan Partnering Award by the UK Biotechnology and Biological Sciences Research Council. DW was supported by the Marie Curie program and by the German Research Association (DFG Research Training School "dIEM oSiRiS" 1387/1). This publication is based on work (EJC) supported in part by Award No KUK-C1-013-04, made by King Abdullah University of Science and Technology (KAUST). FTB acknowledges support by the NIH (grant 1R01GM081070-01). JC is supported by the European Commission, DG Information Society, through the Seventh Framework Programme of Information and Communication Technologies, under the VPH NoE project (grant number 223920). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: lenov@ebi.ac.uk

The publication of this Perspective is not an endorsement by PLoS of MIASE, but rather encouragement to have an active dialog around the development of a standard.

Box 1. Glossary

MIASE *Minimum Information About a Simulation Experiment*. Reporting guidelines specifying the information to be provided with the description of a simulation in order to permit its correct interpretation and reproduction.

MIASE compliant A simulation description that provides all information listed by the MIASE guidelines.

MIRIAM *Minimum Information Required in the Annotation of Models*. Reporting guidelines specifying the information to be provided with an encoded model in order to permit its correct interpretation and re-use.

Model A mathematical representation of a biological system that can be manipulated and experimented upon (simulated).

Model description Set of formal statements describing the structure of the components of a modeled system, whether entities or events, encoded in a computer-readable form.

Repeatability The closeness between independent simulations performed with the same methods on identical models with the same experimental setup.

Reproducibility The closeness between independent simulations performed with the same methods on identical models but with a different experimental setup.

Simulation A numerical procedure performed on a model that aims to reproduce the spatial and temporal evolution (the behavior) of the system represented by the model, under prescribed conditions.

Simulation experiment A set of procedures, including simulations, to be performed on a model or a group of models, in order to obtain a certain set of given numerical results.

models are created and simulated as testable hypotheses in order to determine whether or not they are compatible with experimental data or expected future observations; their analysis supports the design of additional experiments and helps in the synthesis of engineered biological systems. The acceptance of the computationally aided systems biology approach has led to the creation of models at an ever increasing rate, as shown by the rapid growth of model databases. Because of the size of the systems considered, and their multi-scale aspects (both temporal and spatial), modeling activity in integrative systems biology requires researchers to leverage new approaches from prior work. Initiatives to establish standards for describing models and simulations have already been advocated in 1969, e.g., to "establish a standard form of what a model should be like, how it should be described and documented […]. This is intended in part to facilitate communication of information about models, which may be difficult owing to their complexity" [6].

Such an endeavor requires the model descriptions (specifying the mathematical expressions and parameters for a given model) to be stored and exchanged in a way that allows for their efficient reuse [7,8]. Once the model descriptions are retrieved, the user typically wants to test existing simulation protocols on them to obtain a desired output. Currently, most users do so by reading the simulation description in the corresponding publication. This is, however, not only time-consuming, but also error prone. In some cases the published description of a simulation experiment is incomplete, or even wrong, and it requires educated guesswork to reconstruct the original experiment. Examples for such guesses include the initial conditions of simulation, the determination of a starting point for bifurcation diagrams, or the normalization of raw simulation output. Incomplete or erroneous descriptions impede reuse and replication of existing work, and hamper the use of models for educational purposes. Conversely, making this information available to others leads to a greater reuse of existing models.

Standardization plays a central role in facilitating the exchange and interpretation of the outcomes of scientific research, and in particular of computational modeling [9]. Defining which information must be provided when describing an experimental procedure is the task of reporting guidelines, federated in the global project Minimum Information for Biological and Biomedical Investigations (MIBBI) [10]. Those reporting guidelines generally result from consultations with a large community and are carefully thought out. To facilitate reuse of models, MIRIAM [11] was defined in 2005. MIRIAM is a set of rules describing the information that must be provided with a mathematical model in order to allow its effective reuse. Most of the MIRIAM rules deal with the origin and structure of the model, and the precise identification of its components. But the MIRIAM guidelines also state that:

> The model, when instantiated within a suitable simulation environment, must be able to reproduce all relevant results given in the reference description that can readily be simulated.

While mentioning the need for result reproducibility, MIRIAM does not set out to cover the information needed to simulate the models.

As a consequence, it is still necessary to define the core information that needs to be made available to the users of existing models, so that they can perform defined simulations on those models. Once encoded in a computer readable format, these simulation experiment recipes can be downloaded along with the models, either from public resources or publisher Web sites. This will not only allow one to store descriptions of simulation experiments and reproduce them, but also foster their exchange between co-workers, research groups, and even between simulation tools. In this paper, we describe the minimum information that must be provided to make the description of a simulation experiment available to others. Experiment descriptions that provide all necessary information specified in the guidelines are considered MIASE compliant.

Scope of MIASE

MIASE sets out to define minimum requirements for simulation descriptions. It covers the simulation procedures, and allows for the experiments to be reproduced. The particular focus of MIASE is on life science applications.

MIASE Covers Simulation Procedures

One of the difficulties in applying common guidelines to multiple simulation

methods is that the definitions of model and simulation vary, and there is an ill-defined line between the two concepts. This conceptual entanglement is sometimes at the core of mathematical and computational approaches, as with executable biology [12], where the model *is* the simulation algorithm itself. When the description of biological processes builds on numerical integration, there is often a clear conceptual distinction between a model definition and its numerical simulation over space and time. Both concepts are nevertheless sometimes merged at the level of the description formats. Experienced modelers use this feature to run advanced simulations that may even involve the combination of several models. However, for the purpose of the present discussion, the term "simulation" stands for any calculation performed on a model and describing evolutions of the biological system represented, for instance, over spatial and/or temporal dimensions. This includes, but is not limited to, time series simulations (describing the evolution of model variables over time), parameter scans (iterating a given simulation for a range of parameter combinations), sensitivity analyses (variation of parameters or other model properties according to some algorithm, with additional post-processing such as statistical analysis of results), and bifurcation analyses (experiments to study and find stable and unstable steady states). Every necessary piece of information contributing to the unambiguous description of such a simulation is part of the MIASE guidelines. Conversely, information required for the description of the model structure (covered by MIRIAM) for the determination of the model's parameterization, and the specifics of simulation experimental setups, are not part of the MIASE guidelines.

MIASE Is a Reporting Guideline

Reporting guidelines describe how to report clearly and unambiguously what has been done, by describing the entities involved in the experiment. They are not, on the contrary, meant to describe which experimental approaches are correct, or how an experiment should be performed [13]. MIASE is a therefore neither a standard operating procedure nor a description of correct experimental approaches. As such, MIASE does not cover assumptions made during model design or simulation procedure. As mentioned above, information needed for the model description itself is listed in the MIRIAM guidelines. MIRIAM specifies the information necessary to correctly interpret the model, but does not require the explicit statement as to why this model was chosen to represent a particular biological process. Similarly, the reasons behind the choice of a particular simulation approach, e.g., using a stochastic rather than a deterministic algorithm, are not necessary for a MIASE-compliant simulation description. Also, MIASE does not require any statement about the correctness or the scope of a simulation experiment. Whether or not the simulation results match biological reality and whether or not an experiment should be conducted on a certain model is outside MIASE's mission. Nevertheless, a MIASE-compliant description should be detailed enough to allow others to investigate and discuss whether the experiment setup is correct.

MIASE Enables the Reproduction on Different Experimental Setup

The scope of MIASE is limited to the *reproducibility* of the simulation experiment,

Box 2. Rules for MIASE-Compliant Description of a Simulation Experiment

1. All models used in the experiment must be identified, accessible, and fully described.
   - A. The description of the simulation experiment must be provided together with the models necessary for the experiment, or with a precise and unambiguous way of accessing those models.
   - B. The models required for the simulations must be provided with all governing equations, parameter values, and necessary conditions (initial state and/or boundary conditions).
   - C. If a model is not encoded in a standard format, then the model code must be made available to the user. If a model is not encoded in an open format or code, its full description must be provided, sufficient to re-implement it.
   - D. Any modification of a model (pre-processing) required before the execution of a step of the simulation experiment must be described.
2. A precise description of the simulation steps and other procedures used by the experiment must be provided.
   - A. All simulation steps must be clearly described, including the simulation algorithms to be used, the models on which to apply each simulation, the order of the simulation steps, and the data processing to be done between the simulation steps.
   - B. All information needed for the correct implementation of the necessary simulation steps must be included through precise descriptions or references to unambiguous information sources.
   - C. If a simulation step is performed using a computer program for which source code is not available, all information needed to reproduce the simulation, and not just repeat it, must be provided, including the algorithms used by the original software and any information necessary to implement them, such as the discretization and integration methods.
   - D. If it is known that a simulation step will produce different results when performed in a different simulation environment or on a different computational platform, an explanation must be given of how the model has to be run with the specified environment/platform in order to achieve the purpose of the experiment.
3. All information necessary to obtain the desired numerical results must be provided.
   - A. All post-processing steps applied on the raw numerical results of simulation steps in order to generate the final results have to be described in detail. That includes the identification of data to process, the order in which changes were applied, and also the nature of changes.
   - B. If the expected insights depend on the relation between different results, such as a plot of one against another, the results to be compared have to be specified.

rather than its *repeatability*. Reproducibility deals with the replication of experiments, possibly with a different simulation set up, such as using different simulation tools, while repeatability requires the possibility of replicating a simulation experiment on the same models within the very same simulation environment. Furthermore, MIASE's scope does not include the reproduction of identical numerical results of such an experiment. However, while MIASE does not deal with correctness of simulation results, we encourage modelers to provide means to check that the reproduced simulation experiment provides adequate results, e.g., by providing unique identifiers to the original result.

MIASE Applies to Any Simulation Procedure in Life Science

The MIASE guidelines apply to simulation descriptions of biological systems that could be (but are not necessarily) written with ordinary and partial differential equations. For the time being, and as a consequence of the fact that the effort was launched in the systems biology community, the MIASE guidelines are applicable to the simulation of mathematical models of biochemical and physiological systems. However, MIASE principles are general and should appeal to other communities. It can be expected that MIASE compliance will be directly applicable to a wider range of simulation experiments, such as the ones performed in computational neuroscience or ecological modeling. MIASE could even be extended to cover other areas of mathematical modeling in the life sciences, e.g., process algebra.

The MIASE Guidelines

MIASE is composed of rules, summarized in Box 2, that fall into three categories. Rules 1A to 1D list the information that must be provided about the models to be used in the simulation experiment. All models must be listed or described in a manner that enables the reproduction of the experiment. Rules 2A to 2D specify how to describe the simulation experiment itself. All information necessary to run any step of the experiment must be provided. Finally, rules 3A and 3B deal with the output returned from the experiment. A publication describing a simulation experiment must obey the three levels of rules for the description to be declared MIASE compliant. Detailed explanations of the rules and the rationale behind them is provided in Text S1, and also on the MIASE Web site (http://biomodels.net/miase/). Three examples showing the application of the MIASE rules are described in Text S2.

Conclusion and Perspectives

Biomedical sciences are witnessing the birth of a new era, comparable to physical engineering two centuries ago. The practice of systems biology, and its applied siblings synthetic biology and cell reprogramming, will require the use of modeling and simulations as a routine procedure. Investigations into the behavior of complex biological systems are increasingly predicated on comparing simulations to observations. The simulations must be reproduced and/or modified in controlled ways. Precise descriptions of the procedures involved is the first and mandatory step in any standardization effort.

Scientists involved in the simulation of biological processes at different scales and with different approaches, together with maintainers of standards in systems biology, developed MIASE through several physical meetings and online discussions (see http://biomodels.net/miase/). It is expected that such discussions will continue to develop as other life science communities join them. Efforts have been started to create software tools that can help users to apply MIASE rules. An example is the Simulation Experiment Description Markup Language (SED-ML; [14], http://biomodels.net/sed-ml/). Application programming interfaces are under development in various communities to facilitate the support of SED-ML by simulation tools.

The systematic application of MIASE rules will allow the reproduction of simulations, and therefore the verification of simulation results. Such transparency is necessary to evaluate the quality of scientific activity. It will also improve the sharing of simulation procedures and promotion of the collaborative development and use of models.

Supporting Information

Text S1 Detailed description of the MIASE Guidelines, with a discussion of all the rules, and a workflow depicting the description of the different steps of a simulation experiment.
Found at: doi:10.1371/journal.pcbi.1001122.s001 (0.19 MB PDF)

Text S2 Three examples of MIASE-compliant descriptions of different simulation experiments ran on the same model.
Found at: doi:10.1371/journal.pcbi.1001122.s002 (0.48 MB PDF)

Acknowledgments

Authors are grateful to James Bassingthwaighte, Igor Goryanin, Fedor Kolpakov, and Benjamin Zaitlen for discussions and comments on the manuscript.

References

1. Fall CP, Marland ES, Wagner JM, Tyson JJ (2002) Computational cell biology. Math Med Biol 20: 131–133.
2. Hunter P, Nielsen P (2005) A strategy for integrative computational physiology. Physiology 20: 316–325.
3. Barrett CL, Kim TY, Kim HU, Palsson BØ, Lee SY (2006) Systems biology as a foundation for genome-scale synthetic biology. Curr Opin Biotechnol 17: 488–492.
4. Cellier FE, Greifeneder J (1991) Continuous system modeling. First edition. New York: Springer-Verlag. 755 p.
5. Zeigler BP, Praehofer H, Kim TG (2000) Framework for modeling and simulation. In: Theory of modeling and simulation. Second edition. San Diego: Academic Press. pp 25–36.
6. Garfinkel D (1969) Construction of biochemical computer models. FEBS Lett 2 Suppl 1: S9–S13.
7. [No authors listed] (2005) In pursuit of systems. Nature 435: 1.
8. Le Novère N (2006) Model storage, exchange and integration. BMC Neuroscience 7 Suppl 1: S11.
9. Klipp E, Liebermeister W, Helbig A, Kowald A, Schaber J (2007) Systems biology standards – the community speaks. Nat Biotechnol 25: 390–391.
10. Taylor CF, Field D, Sansone SA, Aerts J, Apweiler R, et al. (2008) Promoting coherent minimum reporting guidelines for biological and biomedical investigations: the MIBBI project. Nat Biotechnol 26: 889–896.
11. Le Novère N, Finney A, Hucka M, Bhalla US, Campagne F, et al. (2005) Minimum information requested in the annotation of biochemical models (MIRIAM). Nat Biotechnol 23: 1509–1515.
12. Fisher J, Henzinger TA (2007) Executable cell biology. Nat Biotechnol 25: 1239–1249.
13. Sherman DJ (2009) Minimum information requirements: neither bandits in the Attic nor bats in the belfry. N Biotechnol 25: 173–174.
14. Köhn D, Le Novère N (2008) SED-ML - An XML format for the implementation of the MIASE guidelines. Lect Notes Comput Sci 5307: 176–190.

Supporting information S1: **Details on the MIASE Guidelines**

The following sections explain the conditions for a simulation experiment description to be MIASE compliant (summarized in Box 2 of the main text). A workflow for the setup of such a simulation description is given in Figure S1. Examples of MIASE compliant simulation descriptions are provided in supporting information S2.

Information about the models to use

An essential step is the precise specification of the model(s) used in the simulation experiment (**see Box 2 of the main text, Rule 1**). In order to be MIASE compliant, a simulation experiment description must identify any and all models used throughout the experiment. These models can be joined with the experiment description, or be made available via a link provided. If models are derived from existing models, the procedures used to derive them have to be precisely described (**Rule 1A**).

Simulation experiments need not be restricted to any one specific model; a simulation experiment description may apply to a number of models, possibly after minor adjustments. It is in fact expected that the same simulation steps may be run on different models, for instance to compare their behaviors, or to cope with model refinement. If, however, the experiment does not reference models, then a MIASE compliant description must instead provide access to a complete description of all of those models (Rule 1B-C). A model for which the code or the description are inaccessible, e.g. provided as a binary black box, does not allow a user and/or a software package to understand its structure and therefore to interpret fully the simulation experiment. This most often precludes the reproduction of the experiment (although in certain cases, with adequate information, it may not preclude its repeatability). As such closed models make exchange problematic or even futile, the MIASE guidelines strongly recommend usage of open machine-readable model descriptions. The use of models available in community-developed standard formats (such as the SBML [2], CellML [3] or NeuroML [4]) and complying with the MIRIAM guidelines is encouraged, when available and suitable.

If a model had previously been made publicly available, it should be referred to using a reference to that public resource. However, the reference must only lead to an unambiguously identifiable model. Other, less favored, possibilities include databases of models in non-standard formats, or reference to an actual implementation in source-code. MIASE compliance does not restrict the encoding of a model to particular specified formats.

It is often necessary to modify a model prior to simulation, e.g. certain model parameters may need refinement in order for the model to show a particular behavior during simulation. Apart from such simple modifications, models may undergo more complex procedures such as the replacement of a model constituent, whether entity, process or mathematics. These may be implicit and iterative, for instance in the case of a parameter scan. MIASE compliance demands changes to be clearly described within the simulation experiment description (**Rule 1D**). For the example of a parameter scan, the range over which the parameter shall be scanned and the sampling procedure must be provided in the description.

Information about the simulation steps

A MIASE compliant simulation experiment description must contain the information necessary to enable simulations to be run (**see Box 2, Rule 2**). This comprises the types of simulation, any relevant information specific to the simulation types, on which model(s) to apply which simulation

type(s), and in which order, and any other information necessary to reproduce a particular simulation run.

The simulation algorithms should be identified or referred to in an unambiguous way, taking into account the particular algorithm variants and their implementations (**Rule 2A**). This is essential, as different algorithms yield different numerical results for the same theoretical trajectory of the system. For example, integration schemes with polynomial interpolation schemes of a different order will yield different results, and implicit integration schemes may give different results than explicit schemes. The use of controlled vocabularies is recommended; for example, although work is at an early state, using terms from the Kinetic Simulation Algorithm Ontology (KiSAO, http://biomodels.net/kisao/). This facilitates the identification of similar algorithms in case the original cannot be readily re-used. Simulation workflows including sequential and nested simulation experiments must be described. If the simulation experiment is a sequence of different simulations run on different models and using intermediate results, possibly produced by different software, the exact order of the particular steps has to be clearly identified.

All information relevant to a particular simulation procedure must be provided (**Rule 2B**), including the aforementioned simulation algorithms, the range of values and sampling procedure in the case of parameter scans etc. For stochastic simulations, the random number generator and the number of repetitions should be provided. The meshing method used for discretization in some spatial simulations must be provided, although the description of the actual meshing is not covered by MIASE.

It may be that some or all of the simulation steps used for the original experiments were performed with closed-source simulation software, effectively black-boxes for which precise details of the simulation algorithms may be unknown, nor the details of their implementation. If so, all information necessary to reproduce the simulation steps, and not solely to repeat them (i.e. using the same "black box" approach), must be provided (**Rule 2C**). In effect this enables the re-implementation of the black box, so as to run the same simulation experiment. MIASE is designed to be used by researchers willing to exchange their simulation descriptions. A simulation procedure that is impossible to be fully understood and reproduced is not covered by MIASE. We recommend the information required for MIASE compliance be encoded in a standard description format, where such a format exists, so that existing tools can verify the faithful reproduction of simulation experiments. Examples of such standardization efforts are the *Simulation Experiment Description Markup Language* (SED-ML, [5]) or *CellML Metadata* [6].

Sometimes certain hardware or specific software libraries are required to produce correct results. For some types of experiments information about global simulation processes such as hybrid integrators or distributed compute jobs may also be needed. In such cases, MIASE-compliance demands an explanation of the use of that particular setting (**Rule 2D**). However, it must be pointed out that such information cannot be provided in a standard format for the time being, nor can the authors see a solution for it in the foreseeable future. It is nevertheless recommended to encode the explanation in natural language, until standard representations exist.

MIASE's rules are restricted to the parts of the simulation experiment *specific to the scientific problem.* Conversely, the influences that a particular system running the simulation has on the simulation outcome, such as the type of CPU or operating system, are outside the scope of MIASE. In particular all issues arising from real number equality (inconsistency in floating point arithmetic [7]) are not addressed by MIASE. Another example are the seeds used in stochastic simulations. These influences might lead to similar yet not identical simulation values. However, the variations are artifacts and the technical details underlying them are not considered *minimal* information. Nevertheless, even if this information is not *required* for MIASE compliance, its addition to the

simulation description is encouraged if it is essential, or even helpful for later use of the simulation experiment.

Information about the output

A simulation experiment produces a defined set of results, which is presented for the benefit of the end user, whether human or software. The production of these results is part of a MIASE compliant simulation experiment description (**see Box 2, Rule 3**).

It may be that the numerical results obtained from the simulation steps used in the experiment do not constitute the final desired output. A MIASE compliant experiment description must include all necessary procedures required to be applied to the raw simulation results in order to obtain the appropriate result (**Rule 3A**). Examples for such post-processing are the conversion of units from different simulation runs, normalization of results, or transformation of a trajectory into a movie.

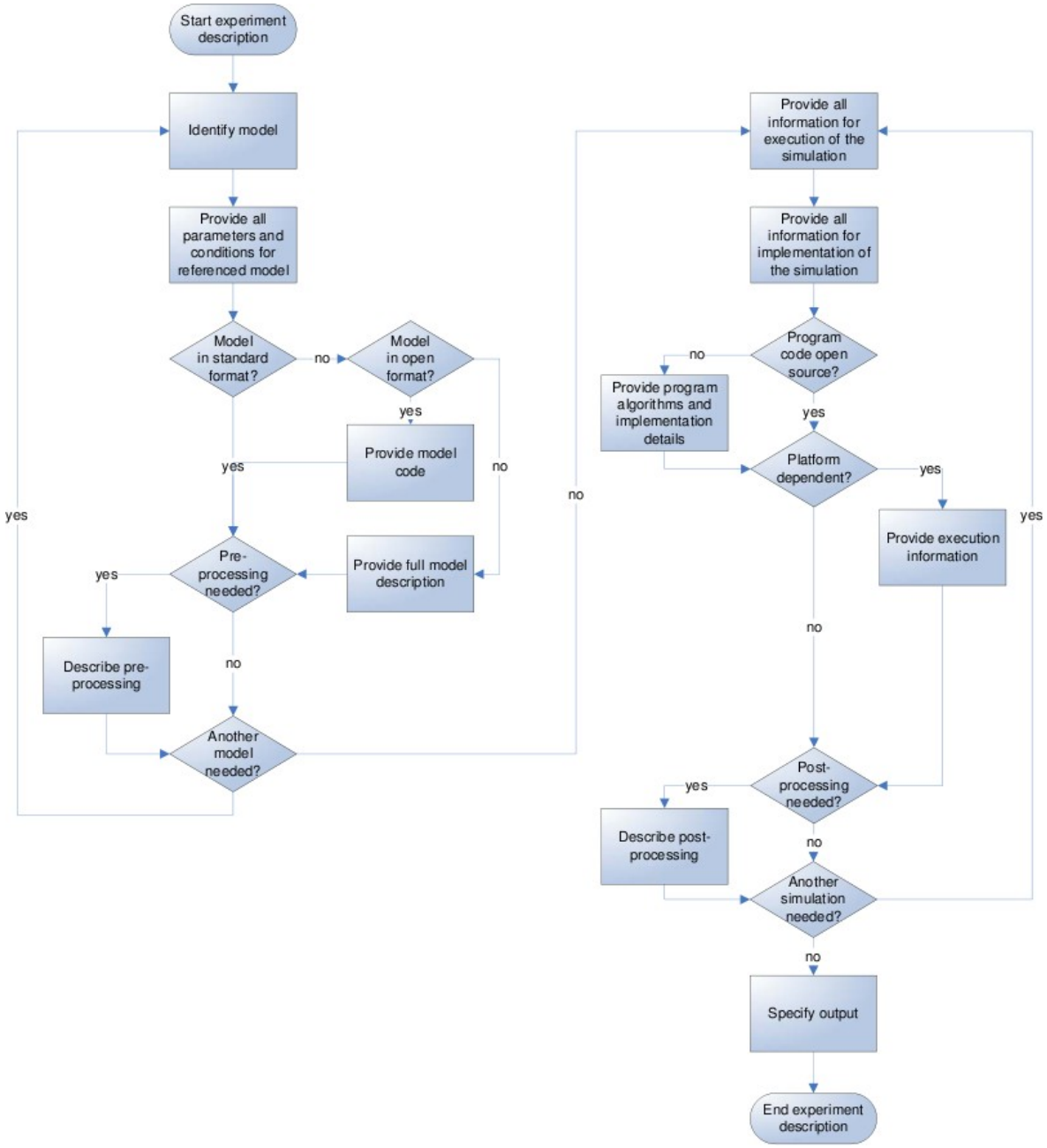
The output of the simulation experiment can be presented under different forms, e.g. textual, in a table or using descriptors, but also graphical, or in a movie. While detailed characteristics of specific output types need not be specified, the general format to present results should be described (**Rule 3B**). A time-course, where some model variables are plotted against time provides different insights than a phase portrait that plots different model variables against one another. While MIASE covers the description of output types, it does not address the exact visual rendering of the simulation results. The visual description, such as the type and appearance of curves, movies, the scaling, or the labels, are not part of the minimal description, since this information is not necessary to understand and reproduce the simulation procedure. The same principle applies to the definition of output tables – while the process of gaining the data and specifying the content of the single columns is within the scope of MIASE, the specification of output formats, such as how to format numbers or the order of columns, is not considered relevant for MIASE compliance.

References

1. International Organisation for Standardisation (ISO). Information processing - Documentation symbols and conventions; program and system flowcharts. ISO 5807.
2. Hucka M, Finney A, Sauro HM, Bolouri H, Doyle JC et al (2003) The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. Bioinformatics 19: 524–531.
3. Cuellar AA, Lloyd CM, Nielsen PF, Bullivant DP, Nickerson DP et al. (2003) An Overview of CellML 1.1, a Biological Model Description Language. Simulation 79: 740-747.
4. Goddard NH, Hucka M, Howell F, Cornelis H, Shankar K et al. (2001) Towards NeuroML: model description methods for collaborative modelling in neuroscience. Philos. Trans. R. Soc. London, Ser. B 356: 1209-1228.
5. Köhn D, Le Novère N (2008). SED-ML - An XML Format for the Implementation of the MIASE Guidelines. Lecture Notes in Bioinformatics, 5307: 176-190.
6. Nickerson DP, Corrias A, Buist ML (2008) Reference descriptions of cellular electrophysiology models. Bioinformatics 24: 1112-1114.
7. Chesneaux J (1994) The equality relations in scientific computing. Numerical Algorithms 7: 129-143.

Figure S1

Flowchart representing the rules (see Box 2) for a MIASE compliant simulation. Rectangles \represent processes, diamonds represent decision points [1].



Supporting text S2: **Examples of simulation experiments: Simulating the Repressilator**

The following examples are MIASE compliant descriptions of three simulation experiments performed on the well-known, simple Repressilator model with its rich and variable behavior. The Repressilator is a synthetic oscillating network of transcription regulators in Escherichia coli [1]. The network is composed of three repressor genes (lacI, tetR, and cI) and their promoters forming a cyclic negative-feedback loop. It is implemented in a simple mathematical model of coupled first-order differential equations which describe the interactions of the molecular species involved in the network. All six molecular species included in the network (three mRNAs, three repressor proteins) participate in creation (transcription/translation) and degradation processes. The model is used to determine the influence of the various parameters on the dynamic behavior of the system. In particular, parameter values were sought which would induce stable oscillations in the concentrations of the system components. Oscillations in the levels of the three repressor proteins are obtained by numerical integration.

The three introduced simulation experiments include a timecourse on the original model (**Example 1**), a timecourse on a perturbed version of the model (**Example 2**), and the creation of phase plane plot (**Example 3**).

Example 1

To run a time course simulation on the model as it is available from the BioModels Database [2], the following steps have to be followed (the execution will lead to Figure 1C of the original publication):

1. Import the model identified by the Unified Resource Identifier [3] "`urn:miriam:biomodels.db:BIOMD0000000012`" (NB: this is the reference for the model encoded in SBML and stored in BioModels Database. An equivalent reference for the model encoded in CellML and stored in the CellML repository [4] would be http://models.cellml.org/exposure/6ad4f33a31aa0b9aa81b6558979d72f5.) [**rules 1A+1B**]
2. Select a deterministic method "KISAO:0000035" (NB: this is the reference for a term of the Kinetic Simulation Algorithm Ontology) to run a simulation on that model. [**rule 2A**]
3. Run a uniform time course for the duration of 1000 minutes with an output interval of 1 min. [**rule 2B**]
4. Report the amount of Lactose Operon Repressor, Tetracycline Repressor and Repressor protein CI against time in a 2D Plot. [**rule 3B**]

The result of the simulation is shown in Figure S2.

Example 2

The fine-tuning of the model can be shown by changing parameters before simulation. The initial value of protein copies per promoter and leakiness in protein copies per promoter can be changed, which move the system's behavior from sustained oscillation to asymptotic steady-state. That change may be described as follows:

1. perform step 1 of Example 1.
2. Change the value of the parameter "tps_repr" from "0.0005" to "1.3e-05". [**rule 1D**]
3. Change the value of the parameter " tps_active" from "0.5" to "0.013". [**rule 1D**]

4. Select a deterministic method (KISAO:0000035) to run a simulation. [**rule 2A**]
5. Run a uniform time course for the duration of 1000 minutes with an output interval of 1 min. [**rule 2B**]
6. Report the amount of Lactose Operon Repressor, Tetracycline Repressor and Repressor protein CI against time in a 2D Plot. [**rule 3B**]

The result of the simulation is shown in Figure S3.

Example 3

The output of the simulation steps may be subjected to data post-processing before plotting it. In order to describe the production of normalized plots of the timecourse simulated above, representing the influence of one variable on another (in phase-planes), one would define the following steps.

1. Perform steps 1 to 3 of Example 1.
2. Collect the time series for lacI, tetR and cI, denoted as PX(t), PY(t) and (cI). [**rule 3A**]
3. Compute the value of the highest value for each of the repressor proteins, max(PX(t)), max(PY(t)), max(PZ(t)). [**rule 3A**]

4.Normalize the data for each of the repressor proteins by dividing each time point by the maximum value, i.e PX(t)/max(PX(t)), PY(t)/max(PY(t)) , and PZ(t)/max(PZ(t)). [**rule 3A**]

5.Report the normalized Lactose Operon Repressor protein as a function of the normalized Repressor protein CI, The normalized Repressor protein CI as a function of the normalized Tetracycline Repressor protein, and the normalized Tetracycline Repressor protein against the normalized Lactose Operon Repressor protein in a 2D plot. [**rule 3B**].

Figure S4 illustrates the result of the simulation after post-processing of the output data.

References

1. Elowitz MB, Leibler S (2000) A synthetic oscillatory network of transcriptional regulators. Nature 403: 335–338.
2. Le Novère N, Bornstein B, BroicherA, Courtot M, Donizello M et al. (2006) BioModels Database: a free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. Nucleic Acids Res 34: D689-D691.
3. Berners-Lee T, Fielding R, Masinter, L (2005) Uniform Resource Identifier (URI): Generic Syntax, http://www.ietf.org/rfc/rfc3986.txt
4. Lloyd CM, Lawson JR, Hunter PJ, Nielsen PF (2008) The CellML Model Repository. Bioinformatics 24: 2122-2123.
5. Hoops S, Sahle S, Lee C, Pahle J, Simus N et al. (2006) COPASI a complex pathway simulator. Bioinformatics 22: 3067–3074.

Figure S2

Time-course of the Repressilator model, imported from BioModels Database (BIOMD0000000012), simulated in COPASI [5], and plotted with Gnuplot (http://www.gnuplot.info/). The number of repressor proteins lacI, tetR and cI is shown as a function of the simulated time.

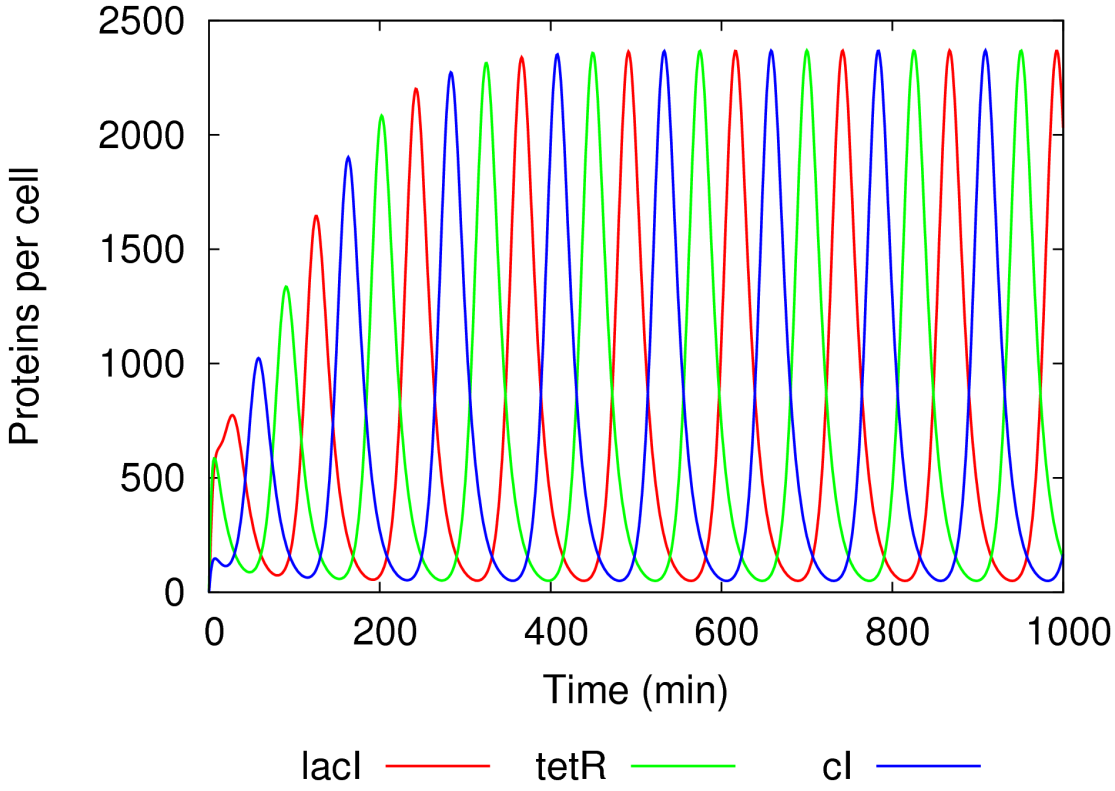


Figure S3

Timecourse of the Repressilator model, imported from BioModels Database (BIOMD0000000012), simulated in COPASI [29] after modification of the strength of the repressed and active promoters , and plotted with Gnuplot (http://www.gnuplot.info/). The number of repressor proteins lacI, tetR and cI is shown as a function of the simulated time.

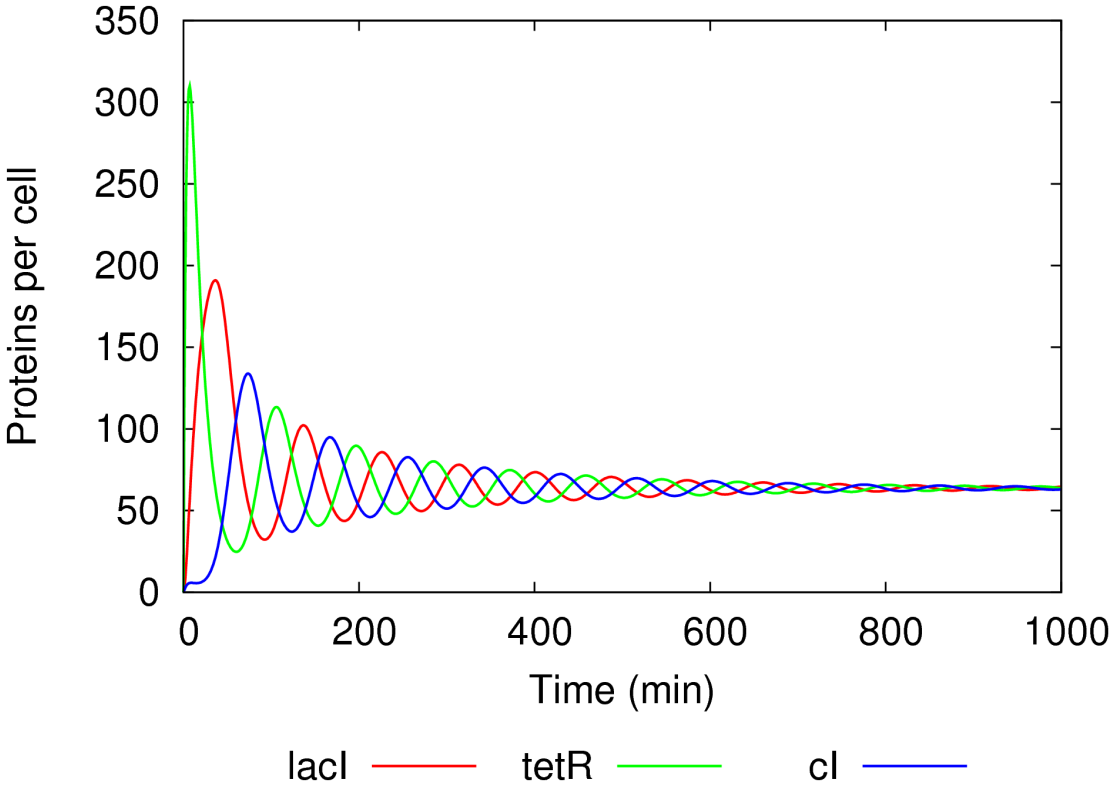
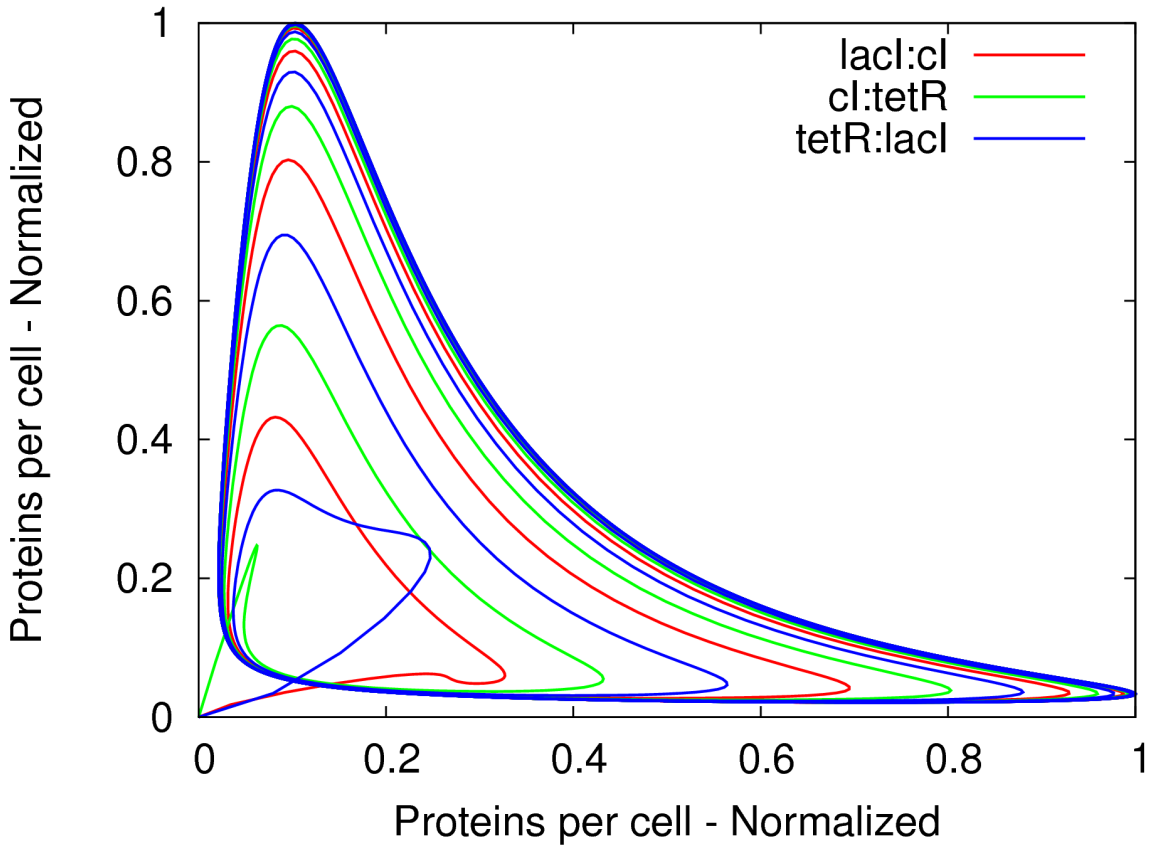


Figure S4

Timecourse of the Repressilator model, imported from BioModels Database (BIOMD0000000012), simulated in COPASI [29], and plotted with Gnuplot (http://www.gnuplot.info/), showing the normalized temporal evolution of repressor proteins lacI, tetR and cI in phase-plane.



4

Langage pour la description des simulations

Waltemath D, Adams R, Bergmann F T, Hucka M, Kolpakov F, Miller A, Moraru I I, Nickerson D, Sahle S, Snoep J L, Le Novère N. Reproducible computational biology experiments with SED-ML – The Simulation Experiment Description Markup Language. *BMC Systems Biology* (2011), 5 :198.

Résumé :

L'utilisation croissante d'expériences de simulation numérique par la recherche biologique moderne entraîne de nouvelles difficultés en ce qui concerne l'annotation, l'archivage, le partage et la reproduction de ces expériences. Les directives pour la description des expériences de simulation (MIASE) récemment publiées proposent un ensemble minimal d'informations qui devraient être fournies pour permettre la reproduction des expériences de simulation entre utilisateurs et outils de simulation. Dans cet article, nous présentons le *Simulation Experiment Description Markup Language* (SED-ML). SED-ML est un format d'échange informatique qui permet d'encoder l'information requise par MIASE pour permettre la reproduction des expériences de simulation. Le format a été développé par la communauté et est décrit dans une spécification technique détaillée ainsi qu'un schema XML. La version de SED-ML décrite dans cette publication est le niveau 1 version 1. Elle couvre la description du type de simulation le plus fréquent en biologie des systèmes, c'est à dire la simulation des comportements cinétiques. Un document SED-ML précise quels modèles utiliser dans une expérience, les modifications à apporter aux modèles avant de les utiliser, quelle procédure de simulation utiliser avec chaque modèle, quels résultats rapporter et comment les présenter. Ces descriptions sont indépendantes de l'implémentation sous-jacente des modèles. SED-ML encode la description des expériences de simulation indépendamment des logiciels. En particulier le format n'est pas spécifique de logiciels de simulation particuliers. On montre ici comment l'on pourra efficacement échanger des descriptions exécutables de simulation grâce à un support croissant de SED-ML. Avec SED-ML les logiciels peuvent échanger la description d'expériences de simulation, permettant la validation et la réutilisation de ces expériences de simulation entre différents logiciels. Les auteurs d'articles rapportant des expériences de simulation peuvent rendre leurs protocoles de simulation disponibles afin que d'autres scientifiques reproduisent leurs résultats. SED-ML étant agnostique quant aux langages utilisés pour encoder les modèles, des expériences utilisant des modèles venant de domaines de recherche différents peuvent être décrites et combinées.

Waltemath et al. BMC Systems Biology 2011, 5:198
http://www.biomedcentral.com/1752-0509/5/198

BMC Systems Biology

METHODOLOGY ARTICLE Open Access

Reproducible computational biology experiments with SED-ML - The Simulation Experiment Description Markup Language

Dagmar Waltemath[1], Richard Adams[2], Frank T Bergmann[3], Michael Hucka[3], Fedor Kolpakov[4], Andrew K Miller[5], Ion I Moraru[6], David Nickerson[5], Sven Sahle[7], Jacky L Snoep[8,9,10] and Nicolas Le Novère[11*]

Abstract

Background: The increasing use of computational simulation experiments to inform modern biological research creates new challenges to annotate, archive, share and reproduce such experiments. The recently published *Minimum Information About a Simulation Experiment* (MIASE) proposes a minimal set of information that should be provided to allow the reproduction of simulation experiments among users and software tools.

Results: In this article, we present the Simulation Experiment Description Markup Language (SED-ML). SED-ML encodes in a computer-readable exchange format the information required by MIASE to enable reproduction of simulation experiments. It has been developed as a community project and it is defined in a detailed technical specification and additionally provides an XML schema. The version of SED-ML described in this publication is *Level 1 Version 1*. It covers the description of the most frequent type of simulation experiments in the area, namely time course simulations. SED-ML documents specify which models to use in an experiment, modifications to apply on the models before using them, which simulation procedures to run on each model, what analysis results to output, and how the results should be presented. These descriptions are independent of the underlying model implementation. SED-ML is a software-independent format for encoding the description of simulation experiments; it is not specific to particular simulation tools. Here, we demonstrate that with the growing software support for SED-ML we can effectively exchange executable simulation descriptions.

Conclusions: With SED-ML, software can exchange simulation experiment descriptions, enabling the validation and reuse of simulation experiments in different tools. Authors of papers reporting simulation experiments can make their simulation protocols available for other scientists to reproduce the results. Because SED-ML is agnostic about exact modeling language(s) used, experiments covering models from different fields of research can be accurately described and combined.

Background

Reproducibility of results is a basic requirement for all scientific endeavors. This is not only true for experiments in the wet lab, but also for simulations of computational biology models [1]. Reproducibility of simulations (i. e., the closeness between the results of independent simulations performed with the same methods on identical models but with a different experimental setup [1]) saves time in modeling and simulation projects. The Minimum Information About a Simulation Experiment (MIASE, [1]) is a reporting guideline describing the minimal set of information that must be provided to make the description of a simulation experiment available to others. It includes the list of models to use and their modifications, all the simulation procedures to apply and in which order, the processing of the raw numerical results, and the description of the final output. MIASE is part of MIBBI [2], a project aiming at federating Minimum Information guidelines (MIs) in the life sciences. MIs are standards that specify which information should be provided as a minimum to ensure that published results of a given type can be understood,

* Correspondence: lenov@ebi.ac.uk
[11]EBI, Wellcome Trust Genome Campus, Hinxton, Cambridgeshire CB10 1SD, UK
Full list of author information is available at the end of the article

reused, and reproduced. MI standards focus on the information to be provided, but do not specify under which form it must be provided.

Different data formats have been developed to support the encoding of computational models of biological systems. Such model representation formats include, for example, SBML [3], CellML [4] and NeuroML [5]. However, while these formats are widely accepted and used to describe model structure, they do not cover the description of simulation, or analyses performed with the models. To address this need, we created the Simulation Experiment Description Markup Language (SED-ML, http://sed-ml.org/), an XML-based format for the encoding of simulation experiments performed on a set of computational models. Here, we describe SED-ML and its development process as a community project in detail.

Results

SED-ML encodes the description of simulation experiments in XML, in an exchangeable, reusable manner. Figure 1 shows how SED-ML could fit into a modeler's simulation workflow: Ideally, model authors will provide SED-ML files together with their publications, describing how to reproduce the presented simulation results. End-users will then be able to download models together with applicable simulation setups, enabling them to directly run the simulation in a simulation software. End-users might in addition share their own simulation experiment descriptions by exporting SED-ML from their simulation tool.

SED-ML is built of five main descriptive elements: (1) the models used in the experiment; (2) the simulation algorithms and their configurations; (3) the combination of algorithm and model into a numerical experiment; (4) post-processing of results; (5) and output of results. The relations between these elements are illustrated in Figure 2.

(1) Model elements

define the identity and location of the model(s) to be simulated and specify the model's native encoding format. The location is to be given as a Uniform Resource Identifier (URI), which enables software interpreting SED-ML to retrieve the model. In case of a relative URI, the base is the location of the referring SED-ML file. To share model and simulation descriptions together, we advise the use of the SED-ML archive format, described in the specification. To link the SED-ML file to remote model descriptions, we recommend using persistent,

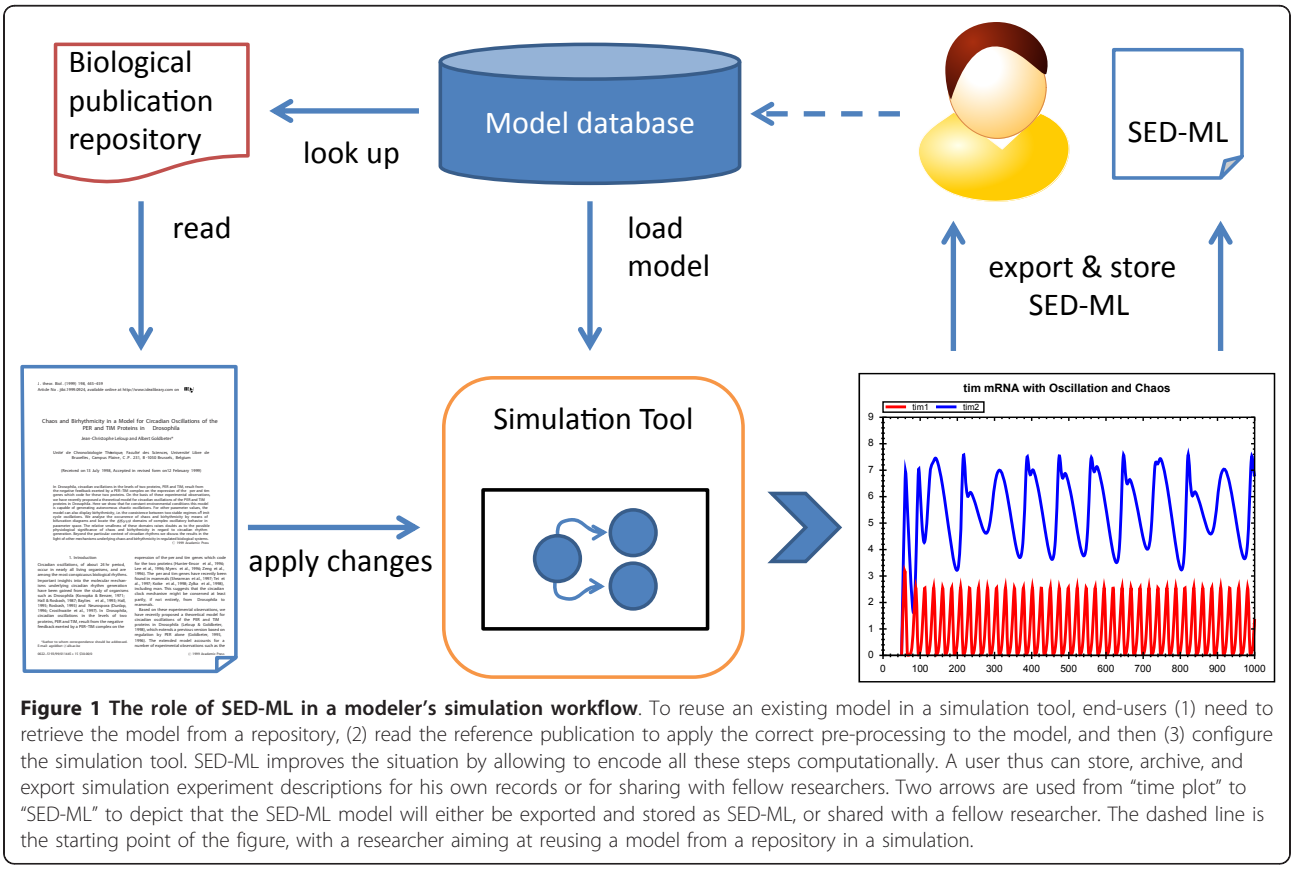


Figure 1 The role of SED-ML in a modeler's simulation workflow. To reuse an existing model in a simulation tool, end-users (1) need to retrieve the model from a repository, (2) read the reference publication to apply the correct pre-processing to the model, and then (3) configure the simulation tool. SED-ML improves the situation by allowing to encode all these steps computationally. A user thus can store, archive, and export simulation experiment descriptions for his own records or for sharing with fellow researchers. Two arrows are used from "time plot" to "SED-ML" to depict that the SED-ML model will either be exported and stored as SED-ML, or shared with a fellow researcher. The dashed line is the starting point of the figure, with a researcher aiming at reusing a model from a repository in a simulation.

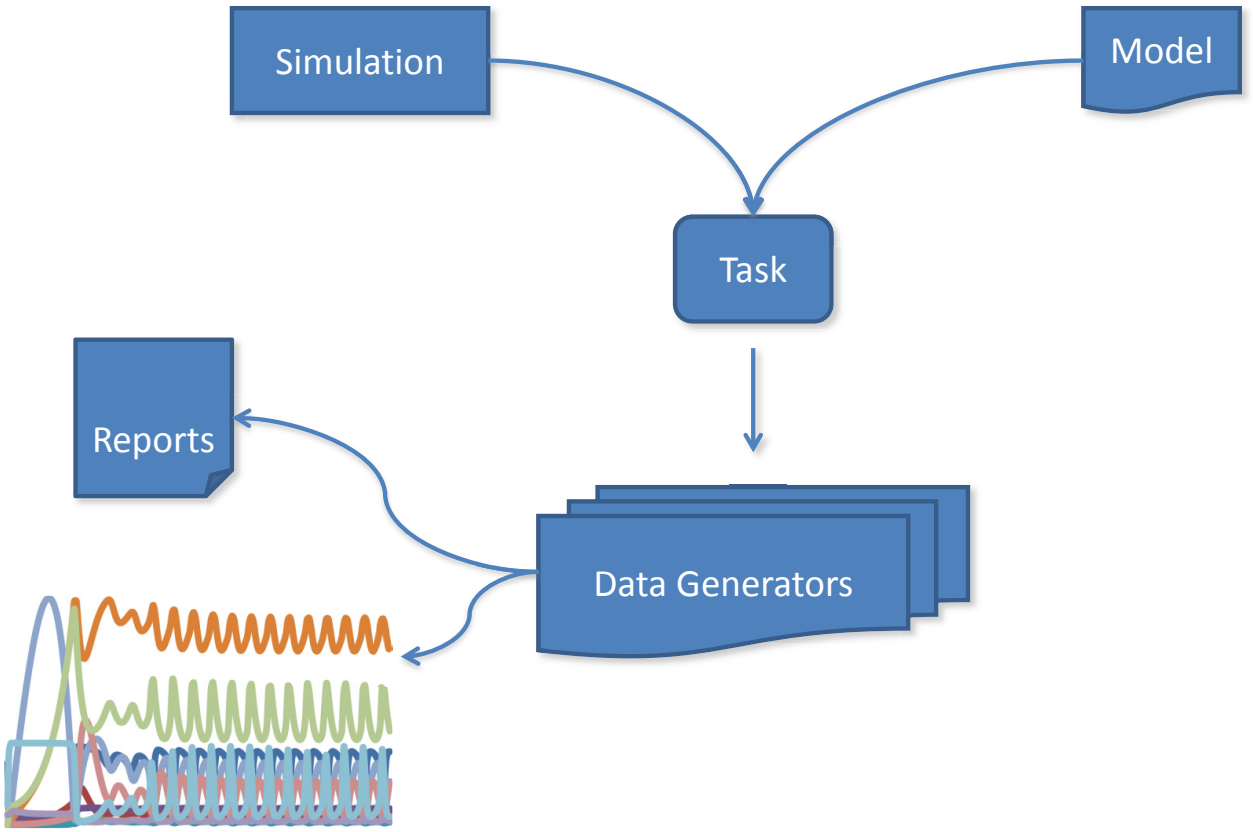


Figure 2 Main SED-ML elements. High level overview of the relations between the five major elements of a SED-ML document. Pairs of model and simulation elements are used in tasks. The dataGenerators allow to define the post-processing of simulation data to define the desired output (plots or reports).

consistent and accessible model resources. Persistent model resources include, for instance, repositories or databases having a MIRIAM URI [6]. We have restricted SED-ML to model encodings in XML-based languages (such as SBML, CellML, or NeuroML). In order to improve interoperability, the particular language a model is encoded in should be specified using one of the predefined SED-ML language Uniform Resource Name (URN); the list is available from the SED-ML website. Using URNs, one can specify a language precisely (e. g., "SBML Level 3, Version 1") or generically (e. g., "CellML (generic)"). Further languages can be registered via the SED-ML website.

In addition to defining the source model's location and encoding, SED-ML model elements can also list changes to be applied to a model before simulation. Such changes could be altering attribute values (e. g., a parameter value in an SBML model or the initial_value of a CellML variable) or changing the model structure. Attribute values may undergo a simple substitution or more complex calculation using content MathML 2.0 [7]. The model structure may be changed by adding or removing XML elements. XPath [8] expressions identify the target XML to which a change should be applied, thereby identifying model entities required for manipulation in SED-ML.

(2) Simulation elements

define the simulation algorithms to be used in the experiment and their configuration. Simulation algorithms are specified using terms from the Kinetic Simulation Algorithm Ontology (KiSAO, http://biomodels.net/kisao/, [9]). KiSAO classifies and characterizes kinetic simulation algorithms, such as those commonly used in systems biology. Furthermore, configuration details of the simulation can be described in SED-ML, such as the start and end times, or the number of time points to output. The current implementation supports the description of time course simulation setups. Extensions towards further experiment types are already being discussed and will be available in the next versions, including the description of steady-state analyses and nested simulations, such as parameter scans.

(3) Task elements
apply a particular simulation algorithm to a specific model. Because simulations and models are described independently, they can be combined in diverse ways. For example, the behavior of one model can be tested with a deterministic and a stochastic simulation algorithm, or a simulation can be applied to different versions of a model with varying parameterization (or other arbitrary model changes applied to the SED-ML model element)

(4) Data Generator elements
define transformations of raw simulation output generated by a task into the desired numerical form. For example, the simulation output might need normalization or scaling before output. Data generators can simply be references to a model variable, but may also be defined through complex mathematical expressions encoded using content MathML. Some variables used in an experiment are not explicitly defined in the model, but may be implicitly contained in it and therefore not addressable using XPath. The 'time' variable in SBML is a common example. To allow SED-ML to refer to such variables in a standard way, the notion of implicit variables has been incorporated into SED-ML. These so-called symbols are defined following the idea of MIRIAM URNs and using the aforementioned SED-ML URN scheme. To refer to the definition of SBML 'time' from a SED-ML file, for example, the URN is urn:sedml:symbol:time. The list of predefined symbols is available from the SED-ML website. From that source, a mapping of SED-ML symbols onto possibly existing concepts in the individual languages supported by SED-ML is provided.

(5) Output elements
describe how numerical data from the data generators are grouped together. In SED-ML Level 1 Version 1, one can relate two data streams or three data streams, allowing to generate 2D and 3D plots, or provide all the data streams as a set of unrelated arrays.

SED-ML documents can contain zero or more instances of the element types described above. A document describing several simulation experiments in a single file enables multiple simulations on the same set of models; for example, the output obtained from different simulation algorithms could be compared. Alternatively, a SED-ML document linking to several models enables the encoding of experiments to determine the influence of changes to models on the simulation output. Moreover, a SED-ML document describing several outputs provides the user with different views of the simulation results. Future versions of SED-ML may also allow the encoding of chained simulations (where several simulations are to be performed in a predefined order and results from one simulation are used to initialize a subsequent simulation).

All SED-ML elements can be complemented with human-readable notes written in XHTML, and machine-readable annotations. Furthermore annotations enable users to extend SED-ML to cover simulation and analysis procedures that are not (yet) part of the core language. The re-use of other standardized formats inside SED-ML annotations is encouraged; for example, simulation outputs can be annotated with terms from the Terminology for the Description of Dynamics (TEDDY, http://www.ebi.ac.uk/compneur-srv/teddy/, [9]). When annotating SED-ML elements with meta-information, MIRIAM URIs [6] should be used. In addition to providing the data type (e. g., PubMed) and the particular data entry inside that data type (e. g., 10415827), the annotation should be related to the annotated element using the standardized http://BioModels.net qualifiers. The list of qualifiers, as well as further information about their usage, is available from http://biomodels.net/qualifiers/.

Figure 3 shows a graphical representation of a SED-ML file, illustrating how the components described above can be used. In this example, a reference model, model1, is obtained from BioModels Database, while model2 is derived from the reference model by altering a parameter's value. Each model is simulated using identical solver configurations, and various outputs are derived from the main results.

Clearly, the interpretation and execution of SED-ML files will require software support. This is increasingly available, both in the form of application support for end-users wishing to execute simulations encoded in SED-ML, and as software libraries to facilitate the uptake of SED-ML support amongst application developers. To demonstrate the capability of SED-ML to facilitate the exchange of simulation experiment descriptions, we chose several freely available independent applications that support SED-ML: SED-ML Web tools (http://sysbioapps.dyndns.org/SED-ML_Web_Tools/), libSedML-Script (http://libsedml.sourceforge.net/) and SBSIVisual (http://www.sbsi.ed.ac.uk/). In SBSIVisual, we ran a simulation of a simple Circadian clock model [10] to produce oscillating behavior, and exported the simulation configuration in SED-ML format. We then edited this SED-ML file using libSedMLScript to describe how the model can produce chaotic behavior, and uploaded it to SED-ML Web Tools to execute and display both simulations. The workflow is shown in Figure 4.

Discussion

In this article, we describe SED-ML, a language to encode procedures performed during computational

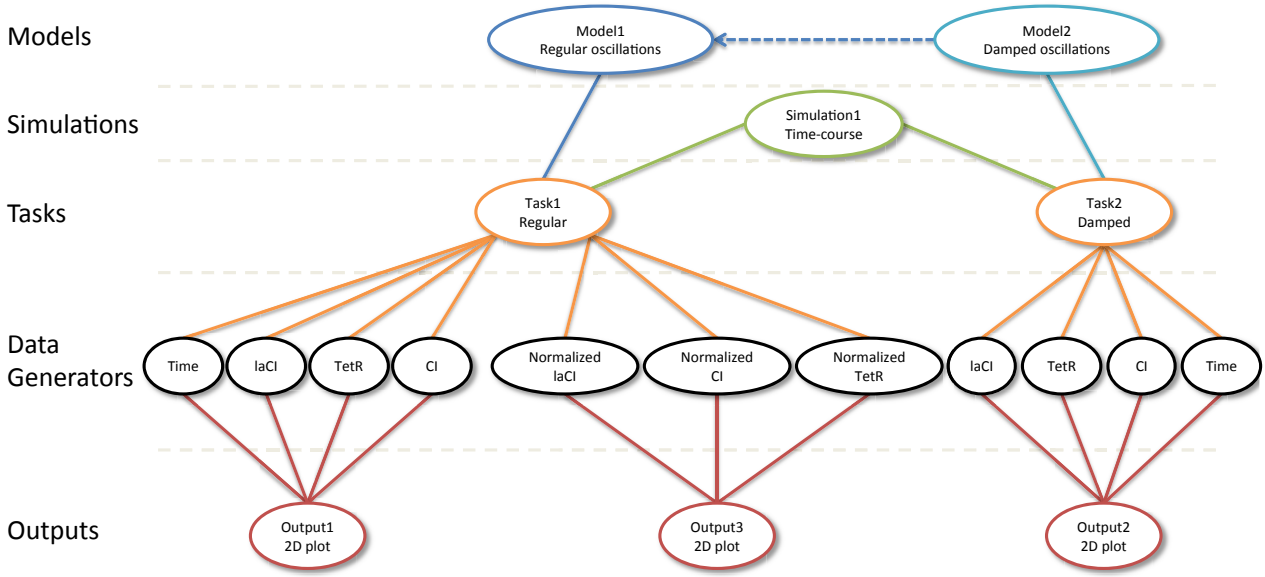


Figure 3 SED-ML document content. Graphical representation of the contents of the SED-ML document describing simulation experiments performed on the Repressilator model [25]. Three outputs are specified. Output 1 (left) displays a plot of levels of three proteins against time, showing regular oscillations. Output 2 (right) produces a plot of the same three protein levels against time after altering the model to produce damped oscillations. Output 3 (center) describes a plot of a simulation of regular oscillations, but after normalizing results and plotting the normalized amounts of each protein against each other. The simulations are described in more detail in Section 1.1 of the SED-ML specification [24], and the SED-ML document is available from the additional file 1.

simulation experiments, and its development process. The first version of SED-ML focuses on encoding uniform time-series experiments, since these are the most widely-used types of numerical model analysis in systems biology. They generally only require a model, and no additional resources such as experimental data.

We expect to extend future versions of SED-ML to include references to experimental data, as the standards and availability of relevant data develop. This is an essential first step towards encoding more complicated experiments such as nested simulations, parameter sweeps, parameter estimation, and sensitivity analysis. The limited scope of SED-ML Level 1 Version 1 lays a firm foundation from which to proceed, and any issues arising from its implementation can be dealt with better at an early stage. Moreover, an early release of a subset of the anticipated future functionality, with widespread community support, fosters participation and uptake amongst the modeling communities targeted by SED-ML.

As SED-ML evolves to describe more complex simulation experiments it will be increasingly useful to link models, simulation descriptions, and experimental data together in a machine-readable way. SED-ML describes the computational steps needed to reproduce particular results of a computational simulation, but it does not encode the simulation results themselves. The latter could be achieved, for instance, by the *Numerical Markup Language* (NuML, http://code.google.com/p/numl/). NuML initially had been part of the *Systems Biology Result Markup Language* (SBRML, [11]), a format to link a model with simulated and experimental datasets. SBRML used a free text 'Software' element to define the software tool, version and algorithm used to generate results. In addition, it will now provide the possibility to point towards a SED-ML file from the SBRML 'Method' element. Both SBRML and SED-ML will use NuML to store lists of numbers, either results or datasets.

SED-ML is agnostic about the underlying model representation formats and the software tool that gave rise to the experiment. The model variables that a SED-ML model needs to be aware of are addressed directly by XPath. SED-ML can thus encode simulation experiments involving models in different formats. Currently SED-ML is restricted to models encoded in XML-based formats. However, we envision that MIASE-compliant models may not always be XML-based and SED-ML should endeavor to address those formats in the future. Whilst many applications are tied to a particular modeling language, the increasing provision of simulation tools as web services [12] would enable a computational workflow to execute such a SED-ML description. The goals of SED-ML closely align with those of the earlier RDF-based CellML Simulation and Graphing Metadata specifications [13] and in the interests of developing a

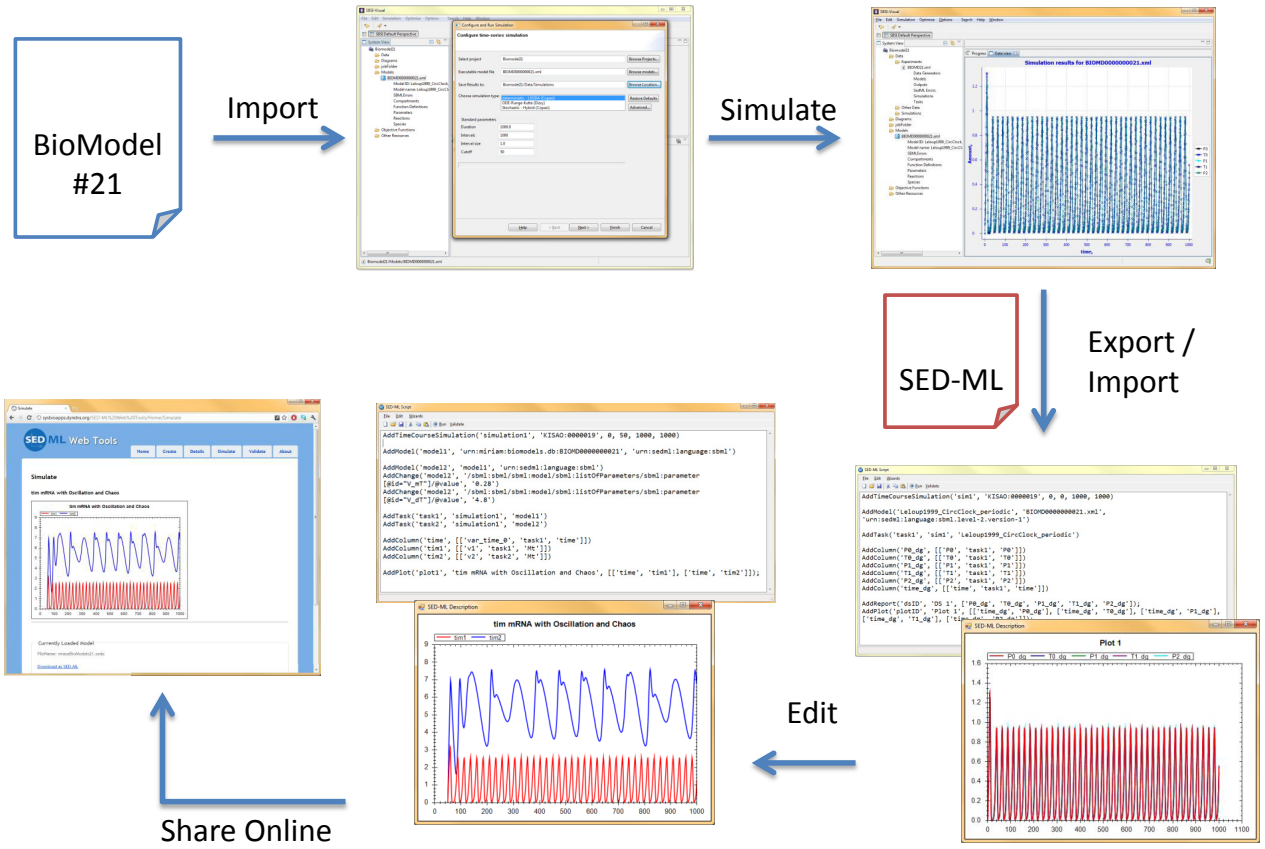


Figure 4 SED-ML demonstration. SED-ML facilitates the exchange of simulation experiments for a number of modeling languages and supports a number of existing simulators. In this sample setup, the SBML model file *BIOMD0000000021* (available from BioModels Database) was loaded into SBSIVisual, a simulation was created and then exported into the SED-ML Script editor (where the simulation can be reproduced). There the simulation was edited to carry out a simulation on a changed model. Finally, the resulting SED-ML experiment was saved as an archive and uploaded to the SED-ML Web Tools, where it was run to reproduce the simulation experiment.

common standard, development of those metadata specifications has been migrated to SED-ML.

While the contributors to the development of the language are primarily from the systems biology community, there is no reason why SED-ML could not be used in other domains that use computational simulation, such as environmental or agricultural modeling, neuroscience or pharmacometrics. Various communities, working on biological model representations, have already committed to the use and support of SED-ML, including SBML, CellML, and the Virtual Cell. Promotion of SED-ML in other realms of science and model representation communities (e. g., ISML, NeuroML, NineML, SimileXML ...) is an ongoing focus. Some of these communities have implemented software support for SED-ML in different tools, including SED-ML validators and a SED-ML visual editor. An up-to-date list is available at the SED-ML website.

The model changes specified in a SED-ML file result in implicit new models. These new models are only instantiated by the simulation environment interpreting the SED-ML file. This important feature of SED-ML allows the exploration of many different model structures to be stored in a compact way. Other methods have been proposed in the past, such as XML diff and patch [14]. This allows not only to change the parametrization of a model by changing the value of an XML attribute, but also to change the structure of the model by adding or removing XML construct. If a user then decides that the result of such changes is a new model, he may choose not to export a simulation description with that set of changes, but to store the modified version as a new model and use it as such in the simulation description. SED-ML is intended to be used by simulation software, as an export/import format. Therefore, the changes that are applicable to a model have to be specifiable within the software tool. As such, the software is responsible for only allowing valid model updates - and also for correctly translating them into SED-ML concepts. SED-ML itself does not restrict the

changes that can be applied to the models mentioned in a SED-ML file.

A number of software libraries have already been made available in C++, Java and .NET. We briefly describe a few of them in the following paragraphs.

libSedML http://libsedml.sf.net/ is a set of .NET libraries for supporting SED-ML. The core library libSedML supports reading, validating and writing of SED-ML descriptions, along with all necessary utility functions for resolving models and XPath expressions. Two additional libraries are included: libSedML-Runner, which allows to schedule and execute simulation experiments encoded in SED-ML files using either RoadRunner (http://roadrunner.sf.net/, [15]) or a variety of simulators exposed through the Systems Biology Workbench (SBW, [16]), such as iBioSim [17] and COPASI [18]. A third library, libSedMLScript, provides a script-based language for defining SED-ML experiments.

jlibsedml (http://sourceforge.net/projects/jlibsedml/) is a Java library for creating, manipulating, validating and working with SED-ML documents. It provides support for retrieval and pre-processing of models, by application of XPath expressions, and also post-processing of raw simulation results as specified by SED-ML dataGenerator elements. The jlibsedml application programming interface (API) follows a similar organization to that of libSBML [19], a successful and popular library for manipulation of SBML documents.

SProS (the SED-ML Processing Service) is an API described in interface definition language (IDL) for creating, reading and manipulating SED-ML documents, and so can be used by multiple software packages. The CellML API [20] provides an implementation of SProS. Future versions of SProS will also provide support for running simulations described in SED-ML and involving CellML models (using the simulation facilities already present in the CellML API).

We see an important role for SED-ML in the publication workflow, and in the enrichment it can bring to manuscripts containing mathematical models. Many journals currently require that models described in a manuscript be made available in electronic form, often in SBML, but software-specific formats are also accepted. Although reviewers would ideally test these models during the review process, this is often not done, perhaps due to time pressure or unfamiliarity with modeling software. As a consequence, many figures that show simulation results cannot be reproduced by the models linked to the manuscript, resulting in a labor-intensive curation step for model repositories, such as BioModels Database [21] and JWS Online model repository [22]. To aid in the reviewing process and prevent discrepancies between manuscript and model, JWS Online, to give one example, has set up a model reviewing workflow with a number of journals. The workflow consists of an initial check by the curators to reproduce simulations in a submitted manuscript. SED-ML will make this workflow significantly easier. Ideally, modelers would provide SED-ML scripts with their manuscript submission, these scripts can be run directly by the curator and make the curation job much easier. If the SED-ML scripts are not provided upon model submission they are generated by the curator and made available to the manuscript reviewer. The script loads the respective model and returns the model simulation. A SED-ML script can be linked to each simulation figure in the manuscript. This publication workflow is shown in Figure 5.

SED-ML Level 1 Version 1 provides a foundation for storing simulation experiment descriptions. It is designed to be easily extensible through the definition of further simulation (and analysis) types. The community is already discussing several such extensions, and in particular to cover nested simulation experiments (needed in parameter scans) and steady state experiments. In addition to new simulation types, another important extension is the ability to consume experimental data and directly address previously-performed simulation results. This will open the door to further analyses such as parameter fitting and optimization tasks. Eventually, this will make SED-ML the format of choice for a compact but comprehensive description of simulation experiments, allowing for the seamless exchange of model, experimental data and simulation results between software tools. We also are hopeful that SED-ML will be used by Taverna-based workflows such as those presented in [23].

Conclusions

Reproducibility of simulation procedures is a basic requirement when working with computational biology models. SED-ML provides structures to describe simulation procedures and allows to reproduce them. The provision of a SED-ML file together with publicly available models simplifies the models' reuse, as the simulation settings can be directly loaded into the simulation software. Together with SBML and SBGN to describe and represent the models, SED-ML is a new cornerstone of the edifice enabling to completely encode a computational systems biology project. Since SED-ML is independent of particular model formats, we believe its use will also play a role in bridging different communities towards integrative systems biology.

Methods

SED-ML Community development

SED-ML is a community effort that has been developed in cooperation with several modeling and simulation

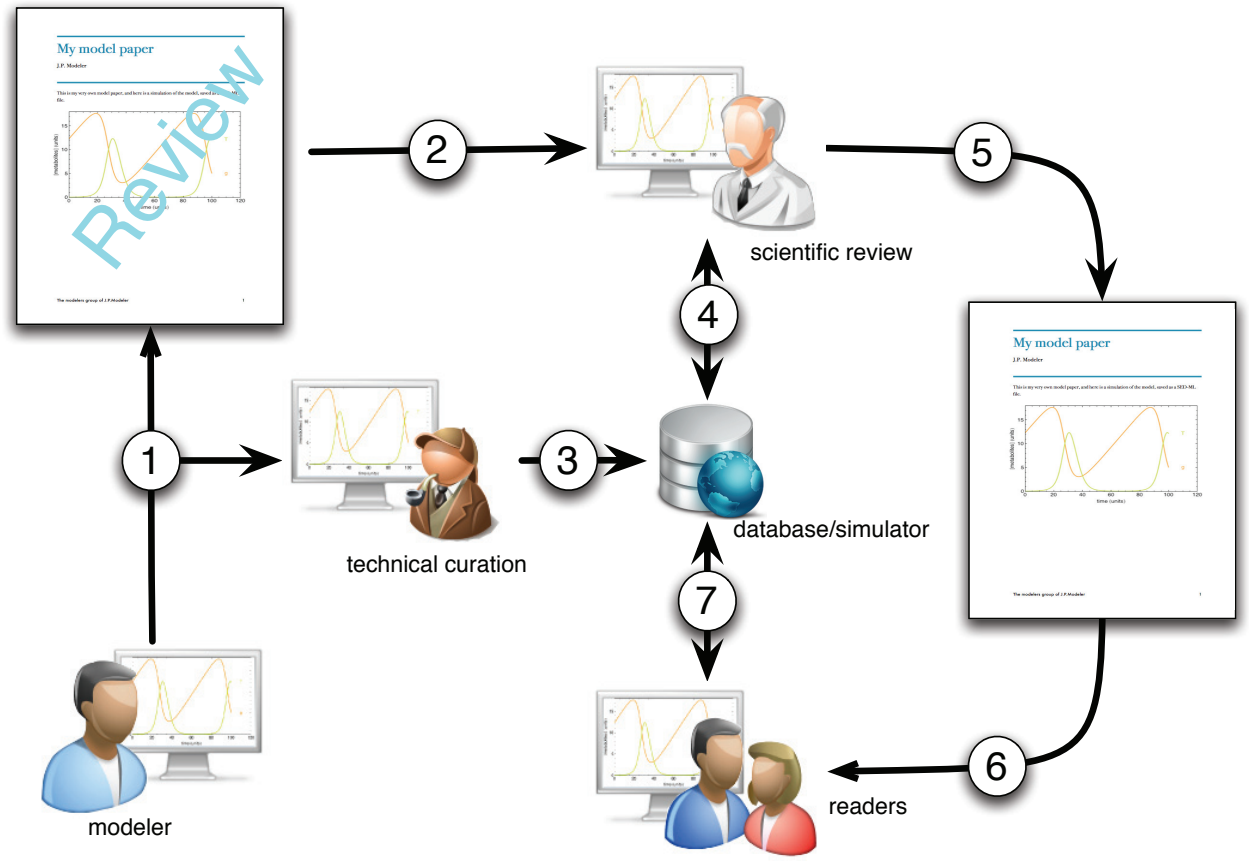


Figure 5 SED-ML, publications and model databases. Many Figures in journal publications cannot easily be reproduced by the curators of model databases and later end-users. The Figure shows how SED-ML can help model curators in reproducing simulations submitted together with a manuscript. A workflow for the publication process would involve the following steps: 1, a researcher submits a manuscript with a mathematical model to a scientific journal; 2, the manuscript is sent out for scientific review; the researcher can submit his model directly to a model database such as BioModels Database or JWS Online, either concurrently with step 1 or via the journal office. 3, Model curators perform a technical curation of the model, (i.e. check whether the model description is complete, whether the model can be simulated and whether the results shown in the manuscript can be reproduced); 4, if the model passes the technical curation it could be made available for the scientific reviewer on a secure site (as it is the case for instance with JWS Online); 5, after scientific review the manuscript might become acceptable and published; 6, after which readers can access the manuscript; and 7, the model is moved to the public database, and is accessible for simulation. SED-ML would greatly facilitate steps 3, 4, and 7."

groups in computational systems biology. The development of SED-ML was begun at the same time as MIASE and KiSAO during a PhD visit by DW in the group of NLN. The SED-ML project was first presented publicly at the 12th SBML Forum Meeting in 2007 and its main structure outlined at both the super-hackathon "Standards and Ontologies for Systems Biology" in 2008 and the combined "CellML-SBGN-SBO-BioPAX-MIASE workshop" in 2009. Since then SED-ML has been developed in collaboration with the communities forming the "computational modeling in biology network" (COMBINE, http://co.mbine.org/). Besides dedicated sessions at various meetings, the development of SED-ML benefits from community interactions on the sed-ml-discuss mailing list (https://lists.sourceforge.net/lists/listinfo/sed-ml-discuss/). Every update in the language, as well as current issues and proposals for language extensions are discussed and voted on in an open forum. The specification development, as well as versions of the UML diagrams and the XML Schema are available from the SED-ML website. The community can also make use of a tracker to report bugs in the language or its implementation. The first official version of the SED-ML specification was published in March 2011. Since then, the community has elected editors to coordinate SED-ML development. The SED-ML editorial board consists of five editors and one editorial advisor. Editors were elected for a duration of two to four years and will be replaced accordingly.

Language specification

SED-ML is described in full detail in the specification document, "Simulation Experiment Description Markup Language (SED-ML): Level 1 Version 1" published in Nature Preceedings in March 2011 [24] and available from the SED-ML website. The specification describes the language and also outlines the typical workflow of creating a SED-ML document; examples show the use of SED-ML with existing models. The SED-ML Level 1 Version 1 specification document is the normative document and an XML Schema and UML diagram are provided as aids for tool developers and SED-ML users. In SED-ML, major language revisions containing substantial changes result in a new "level" while minor revisions containing corrections and refinements of SED-ML elements lead to forthcoming "versions" [24].

SED-ML documents can be validated against the SED-ML XML schema. XML Schema http://www.w3.org/XML/Schema is a W3C standard for describing the structure and content of an XML document. Although the XML Schema describes the structure of SED-ML, some language restrictions described in the normative SED-ML specification document cannot be encoded in XML Schema due to its limited rule constructs. We also provide a UML representation of the language to facilitate its understanding. However, the UML diagrams shown in the SED-ML specification document only support the written text. They do not fully express the constraints of the language.

Interaction with existing standards and technologies

SED-ML re-uses existing standards, conventions and ontologies wherever possible in order to avoid duplication of effort. SED-ML encodes any pre-processing applied to the computational model, as well as post processing applied to the raw simulation results data before output, using MathML 2.0. MathML is an international standard for encoding mathematical expressions in XML. It is also used as a representation of mathematical expressions in other formats, such as SBML and CellML, two of the model representation languages supported by SED-ML. In order to identify nodes and attributes within the XML representations of biological models, SED-ML uses XPath, a language for finding information in an XML document [8]. To identify precisely the type of simulation algorithm in the simulation experiment, SED-ML uses KiSAO [9]. Tools can for instance, use this information to differentiate whether stochastic traces or continuous simulations are requested, or to relate simulation algorithms and substitute one integration method with an equivalent one. Tools can also retrieve the parameters necessary in the configuration of an algorithm, for instance, to automatically generate the corresponding graphical interface. SED-ML is now a core standard of COMBINE, and as such we will seek to keep the maximum interoperability with other standards in computational systems biology.

Additional material

Additional file 1: SED-ML examples file. Repressilator simulation described in SED-ML.

Acknowledgements

The authors thank the whole community of computational systems biology and in particular the members of the network "Computational Modeling in Biology" (COMBINE) for providing requirements and comments. DW received funding for this work from the Marie Curie program and the DFG research training school dIEM oSiRiS (grant 1387/1). AKM was funded partly by the VPH-Share Project and partly by the Maurice Wilkins Centre For Molecular Biodiscovery. RA is grateful for funding by the BBSRC grant BB/D019621/1. IIM was funded by the NIH grants P41-RR013186 and U54-RR022232. MH was funded by the NIH NIGMS grant GM070923. FTB was funded by the NIH/NIGMS grant GM081070.

Author details

[1]Department of Systems Biology & Bioinformatics, Institute of Computer Science, University of Rostock, D-18051 Rostock, Germany. [2]Centre for Systems Biology Edinburgh, CHWaddington Building, University of Edinburgh, Edinburgh EH9 3JD, UK. [3]California Institute of Technology, 1200 East California Blvd., Pasadena, CA 91125, USA. [4]Institute of Systems Biology Ltd., Detskiy proezd 15, Novosibirsk, 630090, Russia. [5]Auckland Bioengineering Institute, The University of Auckland, Auckland, New Zealand. [6]Center for Cell Analysis and Modeling, University of Connecticut Health Center, Farmington, CT 06030, USA. [7]BIOQUANT, University of Heidelberg, Im Neuenheimer Feld 267, Heidelberg, Germany. [8]Department of Biochemistry, Stellenbosch University, Privatebag X1, Matieland 7602, South Africa. [9]Manchester Centre for Integrative Systems Biology, Manchester Interdisciplinary Biocentre, the University of Manchester, 131 Princess Street Manchester, M1 7DN, UK. [10]Molecular Cell Physiology, VU University, Amsterdam, The Netherlands. [11]EBI, Wellcome Trust Genome Campus, Hinxton, Cambridgeshire CB10 1SD, UK.

Authors' contributions

DW and NL initiated the project. All authors participated in the discussions leading to the structure of SED-ML. DW, RA, FB and NL developed the first specification of the language. All authors participated to and approved the final manuscript's preparation.

Received: 2 September 2011 Accepted: 15 December 2011
Published: 15 December 2011

References

1. Waltemath D, Adams R, Beard DA, Bergmann FT, Bhalla US, Britten R, Chelliah V, Cooling MT, Cooper J, Crampin E, Garny A, Hoops S, Hucka M, Hunter P, Klipp E, Laibe C, Miller A, Moraru I, Nickerson D, Nielsen P, Nikolski M, Sahle S, Sauro H, Schmidt H, Snoep JL, Tolle D, Wolkenhauer O, Le Novère N: **Minimum Information About a Simulation Experiment (MIASE).** *PLoS Compututational Biology* 2011, **7(4)**:e1001122.
2. Taylor CF, Field D, Sansone SA, Aerts J, Apweiler R, Ashburner M, Ball CA, Binz PA, Bogue M, Booth T, Brazma A, Brinkman RR, Clark AM, Deutsch EW, Fiehn O, Fostel J, Ghazal P, Gibson F, Gray T, Grimes G, Hancock JM, Hardy NW, Hermjakob H, Julian RK, Kane M, Kettner C, Kinsinger C, Kolker E, Kuiper M, Le Novère N, *et al*: **Promoting coherent minimum reporting guidelines for biological and biomedical investigations: the MIBBI project.** *Nature Biotechnology* 2008, **26(8)**:889-896.
3. Hucka M, Finney A, Sauro H, Bolouri H, Doyle J, Kitano H, Arkin A, Bornstein B, Bray D, Cornish-Bowden A, Cuellar A, Dronov S, Gilles E, Ginkel M, Gor V, Goryanin I, Hedley W, Hodgman T, Hofmeyr JH, Hunter P, Juty N, Kasberger J, Kremling A, Kummer U, Le Novère N, Loew L, Lucio D,

Mendes P, Minch E, Mjolsness E, *et al*: **The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models.** *Bioinformatics* 2003, **19(4)**:524-31.
4. Cuellar AA, Lloyd CM, Nielsen PF, Bullivant DP, Nickerson DP, Hunter PJ: **An Overview of CellML 1.1, a Biological Model Description Language.** *SIMULATION* 2003, **79(12)**:740-747.
5. Gleeson P, Crook S, Cannon RC, Hines ML, Billings GO, Farinella M, Morse TM, Davison AP, Ray S, Bhalla US, Barnes SR, Dimitrova YD, Silver RA: **NeuroML: a language for describing data driven models of neurons and networks with a high degree of biological detail.** *PLoS Computational Biology* 2010, **6(6)**:e1000815+.
6. Laibe C, Le Novère N: **MIRIAM Resources: tools to generate and resolve robust cross-references in Systems Biology.** *BMC Systems Biology* 2007, **58.**
7. Ausbrooks R, Buswell S, Carlisle D, Dalmas S, Devitt S, Diaz A, Froumentin M, Hunter R, Ion P, Kohlhase M, Miner R, Poppelier N, Smith B, Soiffer N, Sutor R, Watt S: **Mathematical Markup Language (MathML) version 2.0. W3C recommendation.** *World Wide Web Consortium* 2003.
8. Clark J, DeRose S: *XML path language (XPath)* 1999.
9. Courtot M, Juty N, Knüpfer C, Waltemath D, Zhukova A, Dräger A, Dumontier M, Finney A, Golebiewski M, Hastings J, Hoops S, Keating S, Kell D, Kerrien S, Lawson J, Lister A, Lu J, Machne R, Mendes P, Pocock M, Ro-driguez N, Villeger A, Wilkinson D, Wimalaratne S, Laibe C, Hucka M, Le Novère N: **Controlled vocabularies and semantics in Systems Biology.** *Molecular Systems Biology* 2011, **7(543)**.
10. Leloup J, Goldbeter A: **Chaos and birhythmicity in a model for circadian oscillations of the PER and TIM proteins in Drosophila.** *Journal of theoretical biology* 1999, **198(3)**:445-459.
11. Dada JO, Spasić I, Paton NW, Mendes P: **SBRML: a markup language for associating systems biology data with models.** *Bioinformatics (Oxford, England)* 2010, **26(7)**:932-938.
12. Bhagat J, Tanoh F, Nzuobontane E, Laurent T, Orlowski J, Roos M, Wolstencroft K, Aleksejevs S, Stevens R, Pettifer S, Lopez R, Goble C: **BioCatalogue: a universal catalogue of web services for the life sciences.** *Nucleic acids research* 2010, **38(suppl 2)**:W689.
13. Beard DA, Britten R, Cooling MT, Garny A, Halstead MD, Hunter PJ, Lawson J, Lloyd CM, Marsh J, Miller A, Nickerson DP, Nielsen PM, Nomura T, Subramanium S, Wimalaratne SM, Yu T: **CellML metadata standards, associated tools and repositories.** *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences* 2009, **367(1895)**:1845-1867.
14. Saffrey P, Orton R: **Version control of pathway models using XML patches.** *BMC Sytems Biology* 2009, **3**:34.
15. Bergmann F, Vallabhajosyula R, Sauro H: **Computational tools for modeling protein networks.** *Current Proteomics* 2006, **3(3)**:181-197.
16. Bergmann FT, Sauro HM: **SBW - a modular framework for systems biology.** *WSC '06: Proceedings of the 38th conference on Winter simulation* 2006, 1637-1645.
17. Myers C, Barker N, Jones K, Kuwahara H, Madsen C, Nguyen N: **iBioSim: a tool for the analysis and design of genetic circuits.** *Bioinformatics* 2009, **25(21)**:2848.
18. Hoops S, Sahle S, Lee C, Pahle J, Simus N, Singhal M, Xu L, Mendes P, Kummer U: **COPASI - a COmplex PAthway SImulator.** *Bioinformatics* 2006, **22(24)**:3067-3074.
19. Bornstein BJ, Keating SM, Jouraku A, Hucka M: **LibSBML: an API Library for SBML.** *Bioinformatics* 2008, **24(6)**:880-881.
20. Miller A, Marsh J, Reeve A, Garny A, Britten R, Halstead M, Cooper J, Nickerson D, Nielsen P: **An overview of the CellML API and its implementation.** *BMC bioinformatics* 2010, **11**:178.
21. Le Novère N, Bornstein B, Broicher A, Courtot M, Donizelli M, Dharuri H, Li L, Sauro H, Schilstra M, Shapiro B, Snoep JL, Hucka M: **BioModels Database: a free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems.** *Nucleic Acids Research* 2006, **34(suppl 1)**:D689-D691.
22. Olivier B, Snoep J: **Web-based kinetic modelling using JWS Online.** *Bioinformatics* 2004, **20(13)**:2143.
23. Li P, Dada J, Jameson D, Spasic I, Swainston N, Carroll K, Dunn W, Khan F, Malys N, Messiha HL, Simeonidis E, Weichart D, Winder C, Wishart J, Broomhead DS, Goble CA, Gaskell SJ, Kell DB, Westerhoff HV, Mendes P, Paton NW: **Systematic integration of experimental data and models in systems biology.** *BMC bioinformatics* 2010, **11**:582.
24. Waltemath D, Bergmann FT, Adams R, Le Novère N: *Simulation Experiment Description Markup Language (SED-ML): Level 1 Version 1* 2011, [Available from Nature Precedings, http://dx.doi.org/10.1038/npre.2011.5846.1].
25. Elowitz M, Leibler S: **A synthetic oscillatory network of transcriptional regulators.** *Nature* 2000, **403(6767)**:335-338.

doi:10.1186/1752-0509-5-198
Cite this article as: Waltemath *et al.*: Reproducible computational biology experiments with SED-ML - The Simulation Experiment Description Markup Language. *BMC Systems Biology* 2011 **5**:198.

5

Terminologies pour la biologie des systèmes

Courtot M, Juty N, Knüpfer C, Waltemath D, Zhukova A, Dräger A, Dumontier M, Finney A, Golebiewski M, Hastings J., Hoops S., Keating S., Kell D.B., Kerrien S., Lawson J., Lister A., Lu J., Machne R., Mendes P., Pocock M., Rodriguez N., Villeger A., Wilkinson D.J., Wimalaratne S., Laibe C., Hucka M., Le Novère N. Controlled vocabularies and semantics in Systems Biology. *Molecular Systems Biology* (2011), 7 : 543.

Résumé :

L'utilisation de la modélisation informatique pour décrire et analiser les processus biologiques est au cœur de la biologie des systèmes. La structure des modèles, la description des simulations et les résultats numériques peuvent être encodés dans des formats structurés, mais il y a un besoin croissant d'une couche sémantique. L'information sémantique ajoute de la signification aux composants des descriptions structurées afin d'aider à leur identification et leur interpretation non-ambigüe. Les ontologies sont un des outils fréquemment utilisés dans ce but. Nous décrivons ici trois ontologies créées spécifiquement pour couvrir les besoins de la biologie des systèmes. La *Systems Biology Ontology* (SBO) fournie une information sémantique a propos des composants d'un modèle. La *Kinetic Simulation Algorithm Ontology* (KiSAO) procure une information à propos des algorithmes existant pour la simulation des modèles en biologie des systèmes, leur caractérisation et leur relation. La *Terminology for the Description of Dynamics* (TEDDY) catégorise les caractéristiques dynamiques des résultats de simulation et les comportements généraux des systèmes. L'existence d'information sémantique allonge la durée de vie d'un modèle et facilite sa réutilisation. Elle nourrie la genèse de nouvelles idées à propos des systèmes modélisés et peut être utilisée pour prendre des décisions éclairées sur les expériences de simulation à lancer.

Molecular Systems Biology 7; Article number 543; doi:10.1038/msb.2011.77
Citation: ***Molecular Systems Biology*** **7: 543**

www.molecularsystemsbiology.com

molecular systems biology

PERSPECTIVE

Controlled vocabularies and semantics in systems biology

Mélanie Courtot[1,19], Nick Juty[2,19], Christian Knüpfer[3,19], Dagmar Waltemath[4,19], Anna Zhukova[2,19], Andreas Dräger[5], Michel Dumontier[6], Andrew Finney[7], Martin Golebiewski[8], Janna Hastings[2], Stefan Hoops[9], Sarah Keating[2], Douglas B Kell[10,11], Samuel Kerrien[2], James Lawson[12], Allyson Lister[13,14], James Lu[15], Rainer Machne[16], Pedro Mendes[10,17], Matthew Pocock[14], Nicolas Rodriguez[2], Alice Villeger[10,17], Darren J Wilkinson[13], Sarala Wimalaratne[2], Camille Laibe[2], Michael Hucka[18] and Nicolas Le Novère[2,*]

[1] Terry Fox Laboratory, Vancouver, Canada,
[2] Department of Computational Neurobiology, EMBL European Bioinformatics Institute, Wellcome-Trust Genome Campus, Hinxton, UK,
[3] Institute of Computer Science, Friedrich-Schiller University, Jena, Germany,
[4] Bioinformatics and Systems Biology, Rostock University, Rostock, Germany,
[5] Center for Bioinformatics Tuebingen, University of Tuebingen, Tübingen, Germany,
[6] Department of Biology, Carleton University, Ottawa, Ontario, Canada,
[7] Ansys, Abingdon, UK,
[8] HITS gGmbH, Heidelberg, Germany,
[9] Virginia Bioinformatics Institute, Blacksburg, VA, USA,
[10] Manchester Interdisciplinary Biocentre, Manchester, UK,
[11] School of Chemistry, University of Manchester, Manchester, UK,
[12] Auckland Bioengineering Institute, University of Auckland, Auckland, New Zealand,
[13] Centre for Integrated Systems Biology of Ageing and Nutrition, Institute for Ageing and Health, Newcastle upon Tyne, UK,
[14] School of Computing Science, Newcastle University, Newcastle upon Tyne, UK,
[15] Biomolecular Signaling and Control Group, Automatic Control Laboratory, Swiss Federal Institute of Technology Zurich, Zurich, Switzerland,
[16] Theoretical Biochemistry Group, University of Vienna, Vienna, Austria,
[17] School of Computer Science, University of Manchester, Manchester, UK and
[18] Division of Engineering and Applied Science, California Institute of Technology, Pasadena, CA, USA
[19] These authors contributed equally to this work
* Corresponding author. Department of Computational Neurobiology, EMBL European Bioinformatics Institute, Wellcome-Trust Genome Campus, Hinxton CB10 1SD, UK. Tel.: +44 (0)1223 494521; Fax: +44 (0)1223 494468; E-mail: lenov@ebi.ac.uk

Received 28.3.11; accepted 7.9.11

The use of computational modeling to describe and analyze biological systems is at the heart of systems biology. Model structures, simulation descriptions and numerical results can be encoded in structured formats, but there is an increasing need to provide an additional semantic layer. Semantic information adds meaning to components of structured descriptions to help identify and interpret them unambiguously. Ontologies are one of the tools frequently used for this purpose. We describe here three ontologies created specifically to address the needs of the systems biology community. The Systems Biology Ontology (SBO) provides semantic information about the model components. The Kinetic Simulation Algorithm Ontology (KiSAO) supplies information about existing algorithms available for the simulation of systems biology models, their characterization and interrelationships. The Terminology for the Description of Dynamics (TEDDY) categorizes dynamical features of the simulation results and general systems behavior. The provision of semantic information extends a model's longevity and facilitates its reuse. It provides useful insight into the biology of modeled processes, and may be used to make informed decisions on subsequent simulation experiments.
Molecular Systems Biology **7**: 543; published online 25 October 2011; doi:10.1038/msb.2011.77
Subject Categories: bioinformatics; simulation and data analysis
Keywords: dynamics; kinetics; model; ontology; simulation

Introduction: semantics in computational systems biology

Models as abstract representations of observed or hypothesized phenomena are not new to the life sciences. They have long been used as tools for organizing and communicating information. However, the form those models take in systems biology has changed dramatically. Traditional representations of biomolecular networks have used natural language narratives augmented with block-and-arrow diagrams. While useful for describing hypotheses about a system's components and their interactions, those representations are increasingly recognized as inadequate vehicles for understanding complex systems (Bialek and Botstein, 2004). Instead, formal, quantitative models replace these static diagrams as integrators of knowledge, and serve as the centerpiece of the scientific modeling and simulation cycle. By systematically describing how biological entities and processes interrelate and unfold, and by the adoption of standards for how these are defined, represented, manipulated and interpreted, quantitative models can enable 'meaningful comparison between the consequences of basic assumptions and the empirical facts' (May, 2004).

The ease with which modern computational and theoretical tools can be applied to modeling is leading not only to a large increase in the number of computational models in biology, but also to a dramatic increase in their size and complexity. As an example, the number of models deposited in BioModels Database (Le Novère *et al*, 2006; Li *et al*, 2010a) is doubling roughly every 22 months while the average number of relationships between variables per model is doubling every 13 months. The models published with the first release of BioModels Database contained on average 30 relationships per

model, and this number rose to around 100 in the 17th release. Standardization of the encoding formats is required to search, compare or integrate such a large amount of models. We have argued that the standards used in descriptions of knowledge in life sciences can be divided into three broad categories: content standards, syntax standards and semantic standards (see for instance the matrix in Le Novère, 2008). Content standards provide checklists or guidelines as to *what* information should be stored for a particular data type or subject area. Examples of such Minimum Information checklists are hosted on the MIBBI portal (Taylor *et al*, 2008). Syntax standards provide *structures* for formatting the information requested in a content standard. Frequent examples are representation formats, for instance using an XML language. Semantic standards provide a unified, common *definition* for all words, phrases or vocabulary used to describe a particular data type or subject area. By using standards from these three categories in concert, model descriptions can achieve both human and computational usability, reusability and interoperability, and it has even been claimed that 'the markup is the model' (Kell and Mendes, 2008).

Computational models, expressed in representation formats such as the Systems Biology Markup Language (SBML; Hucka *et al*, 2003), CellML (Lloyd *et al*, 2004) and NeuroML (Gleeson *et al*, 2010), still require much human interpretation. While syntax standards define the format for expressing the mathematical structure of models (i.e. the variables and their mathematical relationships), they define neither what the variables and the mathematical expressions represent, nor how they were generated. Where this critical information is communicated through free-text descriptions or non-standard annotations, it can only—if at all—be computationally interpreted with complex text-mining procedures (and hardly even with those; Ananiadou *et al*, 2006). Existing modeling tools that work only with unannotated models are therefore restricted to a fraction of the overall model information available, omitting the crucial semantic portion encoded in non-standard annotations. Furthermore, textual descriptions of semantics can be ambiguous and error-prone. Subsequent activities such as model searching, validation, integration, analysis and sharing all suffer as a result; software tools are of limited use without standardized, machine-readable data. The extent of semantic information associated with models is potentially unlimited and susceptible to rapid evolution. Thus, to provide for maximum flexibility, semantic information should be defined independently of the standard formats used for model encoding. This allows for easy updates and extensions of the vocabulary as science evolves, without invalidating previously encoded models. Making use of ontologies, as one approach of encoding semantics, has gained momentum in life sciences over the last decade (Smith, 2003). Ontologies are formal representations of knowledge with definitions of concepts, their attributes and relations between them expressed in terms of axioms in a well-defined logic (Rubin *et al*, 2008). Ontologies include information about their terms, especially definitional knowledge, and provide a single identifier for each distinct entity, allowing unambiguous reference and identification. In addition, ontologies can be augmented with terminological knowledge such as synonyms, abbreviations and acronyms. Widely used and established examples include the Gene Ontology (Ashburner *et al*, 2000), the Foundational Model of Anatomy (Rosse and Mejino, 2003) and BioPAX (Demir *et al*, 2010). Ontologies used in conjunction with standard formats provide a rich, flexible, fast-evolving semantic layer on top of the stable and robust standard formats.

While existing ontologies adequately cover the biology encoded in models, we extend the idea to model-related information. We describe three ontology efforts to standardize the encoding of semantics for models and simulations in systems biology. These publicly available, free consensus ontologies are the *Systems Biology Ontology* (SBO), the *Kinetic Simulation Algorithm Ontology* (KiSAO) and the *Terminology for the Description of Dynamics* (TEDDY). Together, they provide stable and perennial identifiers, referencing machine-readable, software-interpretable, regulated terms. These ontologies define semantics for the aspects of models, which correspond to the three steps of the modeling and simulation process as shown in Figure 1. The efforts we introduce here are at different stages of development and have different levels of community support; SBO is a well-established software tool, KiSAO gathers increasing community support and TEDDY is as yet in its infancy, being primarily a research project. The purpose of our work is to provide practical tools for computational systems biology and as such, the development of the ontologies presented here is largely driven by the needs of the projects using them. However, their focus and coverage is not voluntarily restricted and any community requirements will, in general, be accommodated. All three ontologies aim to fill specific niches in the concept space covered by the Open Biomedical Ontology (OBO) foundry (Smith *et al*, 2007). The level of compliance with the OBO foundry principles is described for each of the three ontologies in Table I.

Model structure: SBO

SBO describes the entities used in computational modeling. It provides a set of interrelated concepts that can be used to specify, for instance, the type of component being represented in a model, or the role of those components in systems biology descriptions. Annotating entities with SBO terms allows for unambiguous and explicit understanding of the meaning of these entities. In addition, using SBO terms in different representation formats facilitates mapping between elements of models encoded in those formats. SBO is currently composed of seven vocabulary branches: systems description parameter, participant role, modeling framework, mathematical expression (whose constituent terms refer to the previous three branches), occurring entity representation, physical entity representation and metadata representation (Box 1). The concepts are related through ontological subsumption relationships (subclassing), as well as via mathematical constructs expressed in the Mathematical Markup Language (MathML) Version 2 (Ausbrooks *et al*, 2003). If an SBO term carries a mathematical expression then each symbol used within that expression has to be defined by another SBO term. This procedure increases the richness of the information obtained when using such terms, and lends itself to further computational processing.

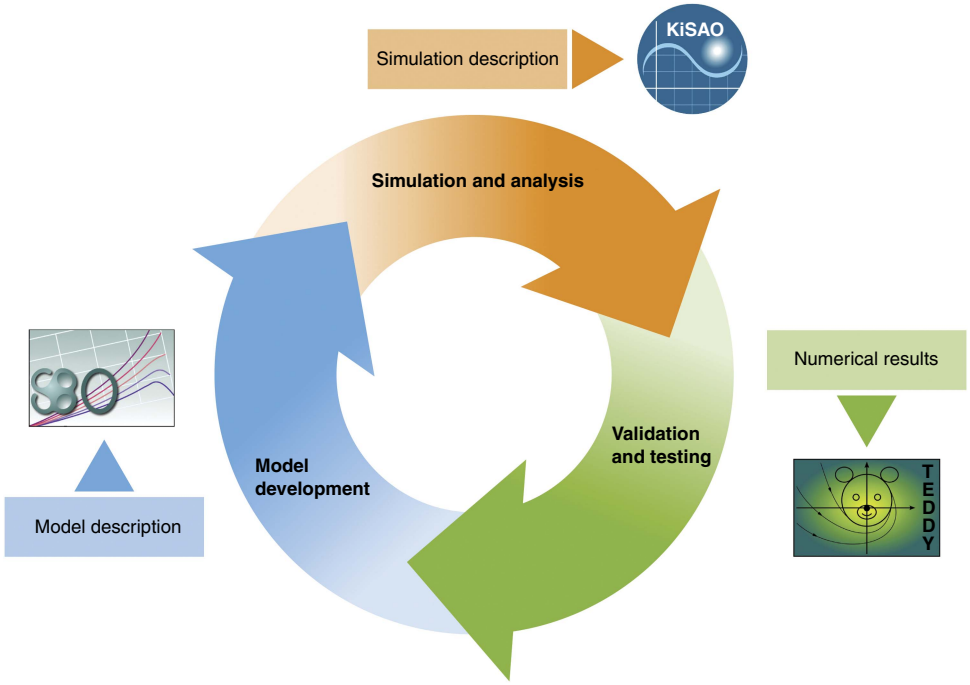


Figure 1 Flowchart depicting the role of SBO, KiSAO and TEDDY in the process of developing and analyzing models.

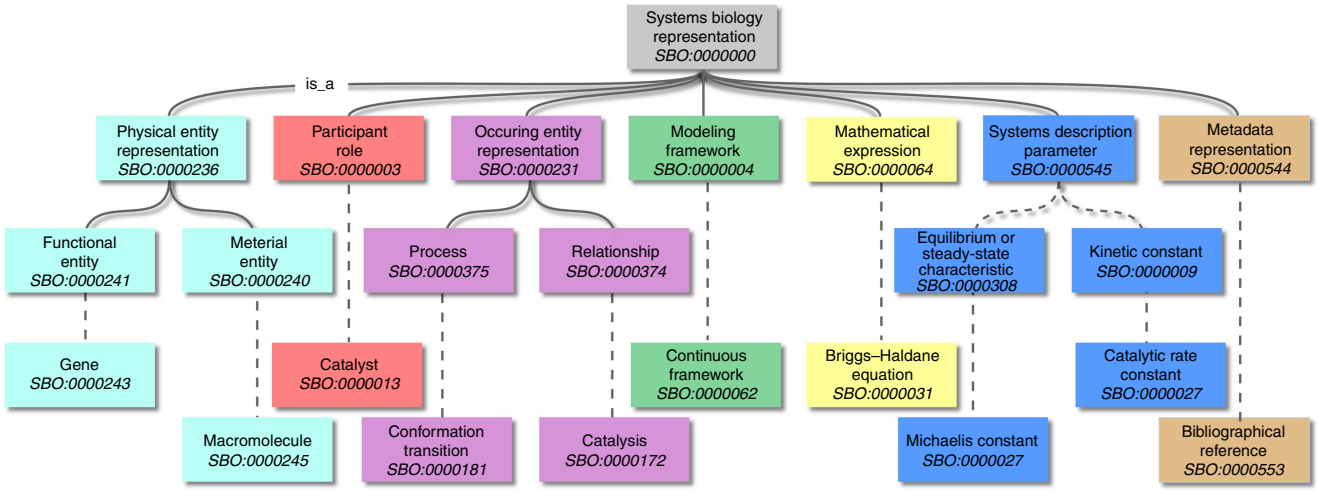
Table I Compliance of the ontologies with the accepted OBO principles[a]

OBO principle	SBO	KiSAO	TEDDY
FP 001 open	Artistic-license	Artistic-license	Artistic-license
FP 002 common format	OBO, OWL	OWL2	OWL
FP 003 URIs	SBO:\d{7}	KISAO:\d{7}	TEDDY:\d{7}
FP 004 versioning	Yes	Yes	No
FP 005 delineated content	Limited overlap at the level of the leaves	Yes	Yes
FP 006 textual definitions	Yes	Yes	Partially
FP 007 relations	Yes	No	No
FP 008 documented	Yes	Partially	Partially
FP 009 users	e.g. SBML, SBGN, NeuroML	Partially	No
FP 010 collaboration	Yes	Yes	Yes
FP 011 locus of authority	BioModels.net SourceForge	BioModels.net SourceForge	BioModels.net SourceForge
FP 012 naming conventions	Yes	Yes	Yes
FP 016 maintenance	BioModels.net	BioModels.net	BioModels.net

Retrieved from http://www.obofoundry.org/wiki/index.php/Category:Accepted on 11 July 2011.
[a]Green: principle fulfilled; yellow: principle partially fulfilled; red: principle not yet fulfilled.

SBO is an open ontology, developed by the community of its users. It is accessible in different formats (OBO format; Day-Richter, 2006; Web Ontology Language; W3C OWL working group, 2009; SBO-XML) under the terms of the artistic license (http://www.opensource.org/licenses/artistic-license-2.0). A number of software tools facilitate the development and exchange of the ontology. The resource is accessible programmatically through Web Services, with a Java library available to aid consumption (Li *et al*, 2010b). SBO, related documentation and associated resources are freely available at http://biomodels.net/sbo/. SBO is also available through the NCBO BioPortal (Noy *et al*, 2009;

Box 1 Structure and content of SBO



SBO terms are presently distributed in seven orthogonal branches described below. See also the graph, where dashed lines indicate that intermediate terms have been omitted.

Physical entity representation: Identifies the *material* or *functional* entity, which is represented by the model's constituent (ontologists call such entities 'continuant,' because they endure over time). Functional entities are those entities that are defined by the function they perform, and include *channel*, *metabolite* and *transporter* entities. The vocabulary for *material* entities identifies the physical type of an entity, and includes terms such as *macromolecule* and *simple chemical*.

Participant role: Identifies the role played by an entity in a modeled process or event, and how it will be affected by it. Examples include roles such as *catalyst*, *substrate*, *competitive inhibitor*. Note that this is different from the meaning of the symbol representing the entity in a mathematical expression, which is described in the systems description parameter vocabulary introduced below.

Modeling framework: Identifies the formal framework into which a given mathematical expression or model component is assumed to be translated. Some examples include *deterministic framework*, *stochastic framework* and *logical framework*. Such contextual information is crucial for interpreting a model description as intended by the author. This branch of SBO is only meant to state the context in which to interpret a mathematical expression, not to express any constraint on the methods to use when instantiating simulations.

Occurring entity representation: Identifies the type of process, event or processual relationships involving physical entities (ontologists call such entities 'occurrent' because they unfold over time). The *process* branch lists types of *biochemical reaction*, such as *cleavage* and *isomerization*. The *relationship* branch depicts types of control that are exerted on biochemical reactions, such as *inhibition* and *stimulation*. When a formula representing such biological events appears in a model, it is frequently difficult to deduce from the formula alone the process that the expression represents; this vocabulary allows the constructs to be annotated in order to make this meaning clear.

Systems description parameter: Defines a parameter used in quantitative descriptions of biological processes. This set of terms includes *forward unimolecular rate constant*, *Hill coefficient*, *Michaelis constant* and others, which can be used to identify the role played by a particular constant or variable in a model. In addition to the subclassing links provided as a relationship between SBO terms, a parameter can be defined as a function of other SBO terms through a mathematical construct.

Mathematical expression: Classifies a mathematical construct used when modeling a biological interaction. In particular, this SBO vocabulary contains a taxonomy of rate equations. Example terms include *mass action kinetics*, *Henri–Michaelis–Menten kinetics* and *Hill equation*. Each term definition contains a mathematical formula, where symbols are defined using three of the vocabularies above (i.e. modeling framework, participant role and systems description parameter). An illustrated example for term Briggs–Haldane rate law SBO:0000031 is shown below.

Metadata representation: Describes the sort of information added to a model description that does not alter the meaning or the behavior of the model. An example for such metadata is a *controlled annotation*.

The branches of SBO are linked to the root by standard *is_a* relationships (Smith *et al*, 2005). Terms within each branch are also linked in this way, which means any instance of a child term is also an instance of its parent term. In the cases where a term includes a mathematical expression, each child term represents a more refined version of the mathematical expressions defined by the parent.

In addition to its stable identifier and term name, an SBO term also contains a definition, synonyms, a list of relationships to child and parent terms, and optionally can also contain a mathematical formula. Free-text comments may be included by the creator of the term for clarification or reference purposes. A log of the history of the term, including creation and modification details, is also available.

[Term]
id: SBO:0000031
name: Briggs–Haldane rate law

Box 1 Continued

def: The Briggs–Haldane rate law is a general rate equation that does not require the restriction of equilibrium of Henri–Michaelis–Menten or irreversible reactions of Van Slyke, but instead makes the hypothesis that the complex enzyme–substrate is in quasi-steady-state. Although of the same form as the Henri–Michaelis–Menten equation, it is semantically different since Km now represents a pseudo-equilibrium constant, and is equal to the ratio between the rate of consumption of the complex (sum of dissociation of substrate and generation of product) and the association rate of the enzyme and the substrate.
comment: Rate law presented by GE Briggs and JBS Haldane (1925): 'A note on the kinetics of enzyme action, *Biochem J*, 19: 338–339.'
is_a: SBO:0000028 ! kinetics of unireactant enzymes
mathml:

```
<math xmlns="http://www.w3.org/1998/Math/MathML">
 <semantics definitionURL="http://biomodels.net/SBO/#SBO:0000062">
  <lambda>
   <bvar><ci definitionURL="http://biomodels.net/SBO/#SBO:0000025">kcat</ci></bvar>
   <bvar><ci definitionURL="http://biomodels.net/SBO/#SBO:0000505">Et</ci></bvar>
   <bvar><ci definitionURL="http://biomodels.net/SBO/#SBO:0000515">S</ci></bvar>
   <bvar><ci definitionURL="http://biomodels.net/SBO/#SBO:0000371">Km</ci></bvar>
   <apply>
    <divide/>
    <apply>
     <times/>
     <ci>kcat</ci>
     <ci>Et</ci>
     <ci>S</ci>
    </apply>
    <apply>
     <plus/>
     <ci>Km</ci>
     <ci>S</ci>
    </apply>
   </apply>
  </lambda>
 </semantics>
</math>
```

ontology 1046, http://purl.bioontology.org/ontology/SBO) and the OBO Foundry.

SBO is developed as a standard ontology, abiding by a set of common development principles, as described by the OBO Foundry (Open Biomedical Ontologies Foundry, http://www.obofoundry.org/wiki/index.php/Category:Accepted). The OBO initiative is an open, community-level collaborative effort to create and apply standardized methodologies in ontology development. Authors of ontologies belonging to this effort are committed to maintain and continually improve their resource, based on community feedback and advancements in their scientific field. SBO itself is an OBO Foundry candidate ontology. The analysis of the compliance level of a candidate ontology with the OBO principles is carried out as part of a formal review, usually by an OBO Foundry coordinator. SBO underwent such a review at the Third Annual OBO Foundry Workshop. The details of the review are publicly available (http://www.ebi.ac.uk/sbo/main/static?page=OBO_status).

Several representation formats in systems biology have already developed formal ties to SBO. Since Level 2 Version 2, SBML elements carry an optional sboTerm attribute, that precisely defines the meaning of encoded model entities (species, compartments, parameters and other elements) and their relationships (variable assignments, reactions, events, etc.), see for instance Figure 2. Information provided by the value of an sboTerm may facilitate distinguishing between, for example, a simple chemical or a macromolecule. Roles played by those entities in processes, such as being an enzyme or an allosteric activator, can also be specified. Furthermore, a model's mathematical formulae may embody implicit assumptions made by the modeler at the time of the model's creation, such as the use of a steady-state approximation rather than a fast equilibrium assumption for enzymatic reactions. Interpretation of SBO terms by software tools enables, for example, checking the consistency of a rate law, and converting reactions from one reference modeling framework to another (e.g. using continuous or discrete variables). Use of SBO terms in SBML is supported by the software libraries libSBML (Bornstein *et al*, 2008) and JSBML (Dräger *et al*, 2011), which provide methods to check for instance whether a term is a subelement of another term, whether a term fits to a certain model component, or to query model elements (for instance, check if myTerm is an 'enzymatic catalyst'). Tools such as semanticSBML (Krause *et al*, 2010) rely, among others, on SBO annotations to search for models or to integrate individual models into a larger one. A growing number of applications have been created to facilitate the addition of SBO terms to model descriptions. Web applications such as Saint (Lister *et al*, 2009) and libraries such as libAnnotationSBML (Swainston and Mendes, 2009) can be used to suggest and add appropriate biological annotations, including SBO terms, to models. Other applications such as SBMLsqueezer (Dräger *et al*, 2008) help identify SBO terms based on existing model components, to further generate appropriate mathematical relationships on top of biochemical maps. SBO terms can be added to experimental data before inclusion in databases, to facilitate their reuse in systems biology projects (Swainston *et al*, 2010). SBO terms also enable the generation of a visual representation from other encoding formats, for instance SBML. The Systems Biology Graphical Notation (SBGN; Le Novère *et al*, 2009) is a set of visual languages to represent

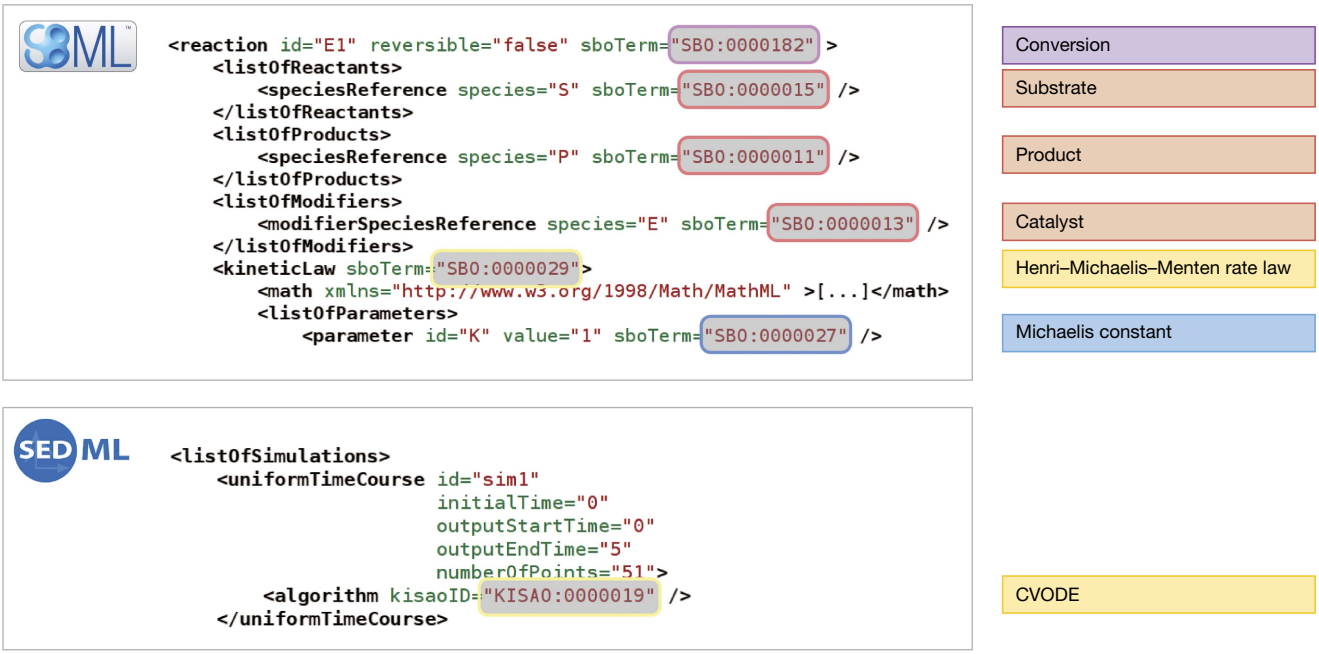


Figure 2 Use of SBO and KiSAO from within SBML and SED-ML. The SBML code on the upper left makes reference to the SBO terms on the upper right. The SED-ML code on the lower left makes reference to the KiSAO term on the lower right.

models and pathways in systems biology. Each symbol from the list of SBGN glyphs corresponds to an SBO term, which provides its precise definition. Reaching out from the realm of systems biology, support of SBO terms via sboTerm attributes is planned in the forthcoming release of NeuroML v2. The CellML initiative also plans to incorporate support for SBO by providing annotation of components with 'MIRIAM' URIs (Le Novère *et al*, 2005).

The use of SBO is not restricted to the development of quantitative models. Using SBO, resources providing quantitative experimental information, such as SABIO Reaction Kinetics (SABIO-RK; Wittig *et al*, 2006), are able to explicitly state the meaning of measured parameters as well as provide information on how they were calculated. In addition, because SBO terms are organized within a relationship network tree, it is possible to infer the relationships between different parameters, and choose the desired level of granularity (depth in the tree). Another example for the application of SBO terms is the combination of structural constraints imposed by SBML (which element contains or refers to which SBO term, as described in the XML schema and the specification document), with the semantic addition of the ontology as described by Lister *et al* (2007). This provides a computationally accessible means of model validation, and ultimately a means of semantic data integration for models (Lister *et al*, 2010). SBO fills a niche not covered by any other ontology. While some existing ontologies have a limited overlapping concept space with SBO, such as the Ontology for Physics in Biology (OPB; Cook *et al*, 2008), none provides features such as the mathematical formulae corresponding to common biochemical rate laws, expressed in ready-to-reuse MathML. OPB is a high-level ontology with a broader scope than SBO. Subbranches of the latter can be cross-referenced at the level of the leaves of the former.

The current coverage of SBO has largely been dictated by the needs of the systems biology community in the last half decade, specifically biochemical modeling. As the field expands so will SBO. Because of the global collaborations that are currently unfolding, in the forthcoming years, the ontology will have to cover the needs of the computational neurosciences, pharmacometrics and physiology. As other computational modeling fields mature, it is anticipated that the scope of SBO will broaden further to cover all modeling in the life sciences.

As the number of terms in SBO increases, there is a growing need to be able to handle scenarios where the content or concept space of SBO impinges upon that of another ontology. In order to maintain orthogonality (one of the primary goals of the OBO Foundry effort), this problem can be handled in SBO through the use of:

(a) MIREOT (Courtot *et al*, 2011), which allows the direct import of terms from an external ontology into a target ontology. This methodology can be used to import single terms, or indeed entire branches, of an external ontology. It allows deferral of the development of some parts of SBO to more appropriately positioned ontology engineers, and is also applicable where the concepts dealt with by the external ontology are thought to be incidental to SBO's main concept space.
(b) Cross-products, where the intersections refer to terms that are essentially a product of terms originating in different ontologies. This method has been used to extend, for example, the Gene Ontology (Mungall *et al*, 2010), and may have some utility for SBO.
(c) Modularization algorithms such as described in Grau *et al* (2007), which would allow to extract part of an ontology while retaining all inferences from the original resource.

Simulation procedures: KiSAO

SBO adds a semantic layer to the formal representation of models in systems biology, resulting in a more complete definition of both the structure and the meaning of computational models. However, formal representations of models do not always provide information about the procedures to follow to analyze and work with the model. A plethora of different results can be generated using a given model (or set of models), depending on the simulation procedure used, the specific simulation algorithms employed and the transformations applied to the variables. Many simulation procedures, and variations thereof, already exist, and more are being regularly introduced. Not all simulation algorithms lead to valid simulation outcomes when run on a specific model. In addition, many algorithms are available only in a limited number of simulation tools, and not all algorithms are publicly available. To enable the execution of a simulation task, even if the original algorithm is not available, it is important to identify both the algorithm intended to be used, as well as analogous algorithms with similar characteristics, that are able to provide comparable results. KiSAO is an ontology developed to address the problem of describing and structuring existing simulation algorithms in an appropriate way. It enables unambiguous references to existing algorithms from simulation experiment descriptions and retrieving information about similar simulation methods. KiSAO furthermore allows the precise identification of the simulation approaches used in each step of the simulation.

KiSAO presents a hierarchy of algorithms, which are linked to their characteristics and parameters (cf Box 2). The hierarchy is based on derivation and specialization: more general algorithms are ancestors of more specific ones, for instance *tau-leaping method* is a descendant of *accelerated stochastic simulation algorithm* and ancestor of *trapezoidal tau-leaping method* and *Poisson tau-leaping method*. Since algorithms are linked to the characteristics they possess, and KiSAO is encoded in OWL, one can reason over the ontology. It is also possible to build algorithm classifications based on any of the characteristics or a combination of several ones. Characteristics currently incorporated into KiSAO include the type of variables used for the simulation (*discrete* or *continuous*), the spatial description (*spatial* or *not spatial*), the system's behavior (*deterministic* or *stochastic*), the type of time steps used by the algorithm (*fixed* or *adaptive*), the type of solution (*approximate* or *exact*) and the type of method (*explicit* or *implicit*). The characteristic-based algorithm classification can be used to provide, for example, possible alternatives to the algorithm covered by a single software package. KiSAO is therefore an ontology to define, with the desired level of abstraction, the algorithms suitable for use within a given simulation setup.

KiSAO is an open ontology, accessible in OWL2 format via the project homepage (http://biomodels.net/kisao/) or through the NCBO BioPortal (ontology 1410, http://purl.bioontology.org/ontology/KiSAO). To facilitate the use of the ontology from within simulation tools and simulation description manipulating software, a free Java library is available (http://biomodels.net/kisao/libkisao.html). The library provides methods to query KiSAO for algorithms, their parameters, characteristics and interrelationships.

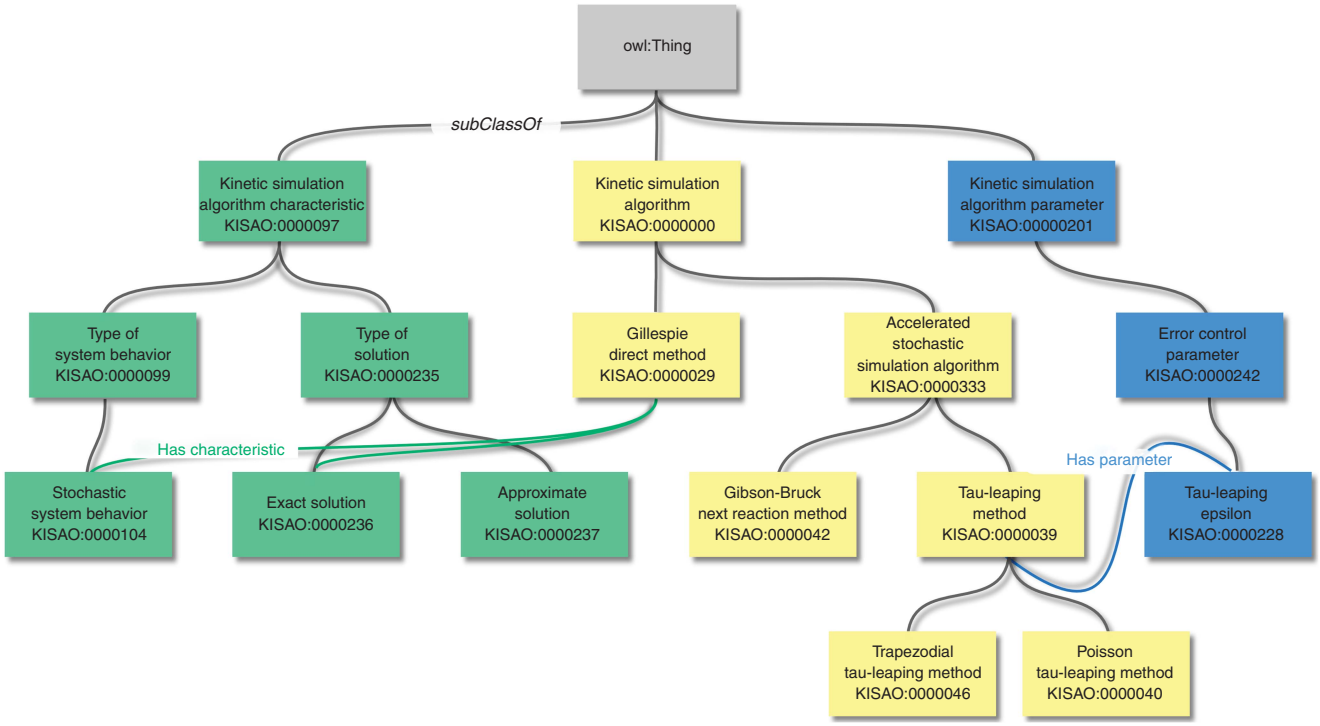
The information about algorithm parameters and their types allows simulation tools to check which parameters need to be specified for the chosen simulation procedure (for instance, *absolute* and *relative tolerances*) or even to perform an update of the user interface containing parameter input fields on-the-fly.

An important use of KiSAO terms is to improve the description of simulation procedures. To date, users must rely on free-text explanations accompanying a model to understand how best to perform a simulation. These explanations often need to be extracted from publications or database entries. Sometimes a script written for a specific simulation environment is provided with a model. The descriptions are specific for a given simulation software package, or rely upon proprietary algorithms, and are therefore rarely reusable in other software systems. The need for a tool-independent, machine-readable description of a simulation experiment has lead to the recent creation of the Simulation Experiment Description Markup Language (SED-ML; Köhn and Le Novère, 2008). SED-ML permits complete description of a simulation experiment by (a) specifying the models to use, (b) specifying the simulation tasks to perform and (c) defining how to report the results. Each algorithm mentioned in an SED-ML file must be identified by a KiSAO term (Figure 2).

The content of KiSAO is not covered by any other ontology at the moment. The Software Ontology (SWO; http://www.ebi.ac.uk/efo/swo) is a subproject of the Experimental Factor Ontology project to describe software used in bioinformatics. It contains an *algorithm* branch, but that does not currently cover modeling and simulation. The Biomedical Resource Ontology (Tenenbaum *et al*, 2011) contains an *algorithm* branch with a few related terms such as *numerical method* and *PDE solver*. However, those terms do not describe the algorithm themselves but the software resources providing access to those algorithms. Other upper ontologies could be used to 'plug in' KiSAO. For example, the SemanticScience Integrated Ontology (Chepelev and Dumontier, 2011) incorporates a term *algorithm*, which is a natural ancestor of *kisao:kinetic simulation algorithm*. EMBRACE Data and Methods ontology (Lamprecht *et al*, 2011) is another upper ontology candidate for KiSAO, which contains a branch *modeling and simulation*. The current emphasis is on structural biology. Plugging KiSAO into a well-crafted upper ontology will facilitate its integration with other OBO ontologies.

KiSAO's current content has been gathered from simulation tools documentation, scientific literature, and key modeling and simulation textbooks. As SED-ML expressiveness increases and it is used within more domains, different types of simulations and analysis will have to be covered. Together with that expanding scope will come representation problems for instance relationships between different types of numerical analyses, possibly very different from kinetic simulation. The description of hybrid algorithms, involving the synchronization of different approaches is also a problem that will become increasingly more important as the tools become more sophisticated.

Box 2 Structure and content of KiSAO



KiSAO consists of three main branches, representing simulation algorithms, their characteristics and parameters. The elements of each algorithm branch are linked to characteristic and parameter branches using *has characteristic* and *has parameter* relationships accordingly.
The *algorithm* branch itself is hierarchically structured using *subClassOf* relationships, which denote that the descendant algorithms were derived from, or specify, more general ancestors (i.e. equivalent to the OBO *is_a*). Every algorithm is annotated with a definition, synonymous names and references to the publication describing it. Some of the algorithms are also annotated with the names of the tools that implement them. In addition to self-contained algorithms, the algorithm branch contains hybrid methods, combining or switching between several algorithms. For example, *LSODA* automatically selects between non-stiff *Adams* and stiff *BDF* algorithms. To represent such interalgorithm dependencies, the complex methods are linked to the algorithms they use by *is hybrid of* and *uses* relationships.

The *characteristic* branch of KiSAO classifies both model and numerical kinetic characteristics. Model characteristics include the *type of variables* used for a simulation—an indication of how the model can be simulated (*discrete* or *continuous*), and information on the *spatial resolution*. Numerical kinetic characteristics include the *system's behavior* (*deterministic* or *stochastic*) as well as the *kind of timesteps* (*fixed* or *adaptive*).

The *parameter* branch describes *error*, *granularity* and *method switching control parameters*, annotated with names, synonyms and descriptions. Information about parameter types is represented using *has type* relationship, for instance *relative tolerance__has type__xsd:double*.

owl:Class: kisao:KISAO_0000039
owl:Annotations:
rdfs:label 'tau-leaping method,'
rdfs:comment 'Approximate acceleration procedure of the Stochastic Simulation Algorithm **[***urn:miriam:biomodels.kisao:KISAO_0000029***]** that divides the time into subintervals and 'leaps' from one to another, firing all the reaction events in each subinterval.',
owl:Annotations:
rdfs:comment 'Gillespie DT. Approximate accelerated stochastic simulation of chemically reacting systems. *The Journal of Chemical Physics*, Vol. 115 (4):1716–1733 (2001). Section V.'
rdfs:seeAlso 'urn:miriam:doi:10.1063/1.1378322,'
owl:Annotations:
oboInOwl:SynonymType 'EXACT'
oboInOwl:Synonym 'tauL,'

isImplementedIn 'ByoDyn,'
isImplementedIn 'Cain,'
isImplementedIn 'SmartCell,'
owl:SubClassOf:
KISAO_0000333 *# 'accelerated stochastic simulation algorithm'*
KISAO_0000245 **some** KISAO_0000237, **#** *'has characteristic' some 'approximate solution'*
KISAO_0000259 **exactly 1** KISAO_0000228, **#** *'has parameter' exactly 1 'tau-leaping epsilon'*

Numerical results: TEDDY

Given a computational model (semantically enriched with SBO terms) and a 'recipe' for producing a simulation experiment (described in part using KiSAO terms), there remains the problem of describing the observed behavior in a systematic and machine-readable manner (Knüpfer *et al*, 2006). The usual approach nowadays involves free-text explanations accompanying a model, e.g.:

> 'Depending on the values of these parameters, at least two types of solutions are possible: the system may converge toward a stable steady state, or the steady state may become unstable, leading to sustained limit-cycle oscillations (Figure 1b and c).' (Elowitz and Leibler, 2000).

While this form of description is concise and pleasant to read, it is not in a form that can be readily interpreted by software tools. Over the last three decades, the success of bioinformatics applications in molecular biology can be attributed mostly to one type of task: comparing sequences. The equivalent task in computational systems biology is comparing dynamical behaviors, tackling questions such as '*How do I find a model describing the protein X and displaying a periodic oscillation?*' '*What behavioral features do all the models have in common?*' '*Which model displays a behavior matching my experimental data?*' Answering these questions requires a means of formally characterizing the qualitative dynamical behaviors of both models and experimental results. Indeed, numerical results of simulation experiments are structurally similar to numerical results of biological experiments. Aligning both is at the core of model parameterization, validation and testing.

TEDDY is an ontology designed to fulfill this need. It comprises four branches: the classification of the concrete temporal behaviors observed in a simulation (the trajectories), the diversifications and relationships between behaviors, the characteristics of specific behaviors and the functional motifs generating particular types of behaviors (Box 3). TEDDY terms should be sufficient to qualify, with variable levels of detail, the critical features of numerical results obtained from simulations as well as those from experimental measurements. Such a qualification could ultimately be extracted from a formal encoding of the results, such as the SAX representation of time series (Lin *et al*, 2007).

Because of the complexity of the relationships between dynamical behaviors, their diversifications and characteristics and their functional motifs, TEDDY is encoded in OWL. TEDDY is available from the project home page (http://biomodels.net/teddy/), with a browsable version provided through NCBO BioPortal (ontology 1407, http://purl.bioontology.org/ontology/TEDDY).

TEDDY only provides the vocabulary for naming the critical dynamical features of models, and relating them within one set of numerical results. In order to comprehensively describe the overall dynamics of a model, including different behaviors with regard to different conditions and the relations between them, an additional language framework is needed. This could in turn be used in conjunction with efforts like the Systems Biology Result Markup Language (Dada *et al*, 2010).

TEDDY is currently a research project, and although much thought was put in its design, its structure is still susceptible to change rapidly. The priority is now to cover the most common dynamical behaviors encountered in biology, and develop procedures to use the ontology in a way to allow reasoning and validation.

Use of ontologies across the modeling and simulation pipeline

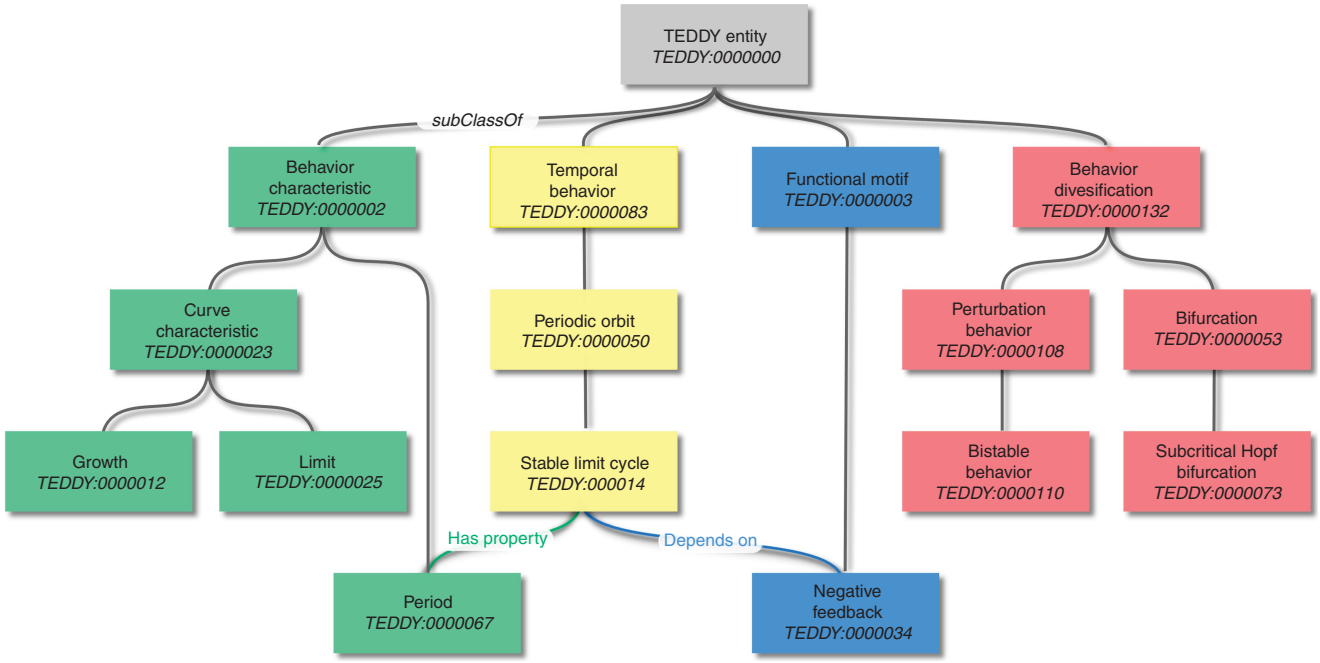
Activities in systems biology are often depicted as a modeling–hypothesis–experiment cycle (Kitano, 2002). Prior biological knowledge forms the basis for designing the model, and in turn the modeling activity generates hypotheses that feed the experimental investigation. Within the main cycle, the modeling and simulation process itself is in fact a cycle (Figure 1). The ontologies described in this article support the multiple steps of this pipeline.

Systematically annotating model components with SBO terms helps not only to document the hypothesis behind the choice of a mathematical representation, but also specify how to interpret it. An example is the 'Michaelis–Menten' equation, which can be an abstracted view of several alternative chemical reaction schemes (Le Novère *et al*, 2007). SBO terms can even be used to propose appropriate mathematical constructs, as shown in the software SBMLsqueezer, and fetch the necessary information from databases such as SABIO-RK. Automatic documentation procedures such as SBML2LATEX (Dräger *et al*, 2009) can directly link controlled vocabulary term identifiers to their unambiguous definitions, which can also be included into a human-readable report on the model structure. Other related ontologies can also be used to enhance the semantics of mathematical description, such as OPB.

The growing complexity of computational models in systems biology makes it more difficult to create models from scratch. In parallel, the increasing number of models available increases the likelihood that a given component has already been published. As such, modelers may decide to reuse portions of existing models as building blocks. Annotation of model components with SBO terms can be used in model search strategies (Schultz *et al*, 2011). Annotation of existing models with TEDDY terms is also potentially an effective way of discovering components of interest by allowing queries such as '*Find a model of MAPK cascade that oscillates*' or '*Find a model of MAPK cascade that can exhibit bistability.*' We anticipate that the same procedure will also make TEDDY extremely useful for synthetic biology, where modularity is seen as a core feature in the construction of novel systems from composable parts. Once appropriate building blocks have been identified, merging them into larger models may be helped by ontologies (Krause *et al*, 2010). SBO can be used to identify model structures that are equivalent although expressed in different formats, and to identify identical model components to act as interfaces between submodels.

In order to run the simulations, modelers need to know the algorithms applicable to simulate the original building blocks, which is the information provided by KiSAO terms. The ontology also supports the retrieval of similar algorithms available in other simulation toolkits. Note that identifying an

Box 3 Structure and content of TEDDY



TEDDY contains four branches, which are linked through a variety of relationships. Within a branch, most of the terms are linked by *subClassOf* relationships.

Temporal behavior describes the way a dynamical system changes with respect to some aspect of the environment (note that a system here can be a variable, a subset of the model's variables or the complete model). Simple examples are *limit cycle* and *fixed point*. More complex examples are *heteroclinic orbit* and *half-stable behavior*. Temporal behaviors can be related by two relationships, *adjacentTo* and *convergeTo*.

Behavior characteristic is a quantitative property that characterizes temporal behaviors. Temporal behaviors can be related to behavior characteristics using *hasProperty*. For instance, a periodic oscillation is characterized by a property *period*, a steady-state by a property *limit*.

Behavior diversification describes the way one or several temporal behaviors are modified or related upon interaction with information external to the system considered. For instance, in a *Hopf bifurcation*, the possible behaviors change by varying a parameter. Behavior diversification can be related to temporal behaviors using the relationships *hasPart*, *hasSubPart*, *hasOnPart* and *hasSuperPart*.

Functional motif describes the structures of a submodel that may generate specific temporal behaviors, such as *negative feedback* or *switch*. Functional motifs are related to temporal behaviors using the relationships *dependsOn* and *realizes*.

owl:Class: TEDDY_0000053

owl:Annotations:
Reference 'http://www.egwald.com/nonlineardynamics/bifurcations.php,'
Definition 'A 'characteristic' describing a qualitative (topological) change in the orbit structure of a system.'
DisplayName 'bifurcation'
owl:SubClassOf:
TEDDY_0000132, **#** *behavior diversification*
TR_0008 **min 1** owl:Thing, **#** *hasSuperPart*
TR_0006 **min 1** owl:Thing **#** *hasSubPart*

algorithm for reuse does not ensure that software claiming to implement the algorithm did so faithfully, without errors or *ad hoc* hypotheses potentially leading to different results in subsequent simulations when compared with the original.

Finally, numerical results, from both experimental measurements and simulations, can be annotated with TEDDY. This information allows verification based, for instance, on temporal logic. Such procedures can be performed during the parameterization of the model, to analyze the results of simulations or to retrieve models based on the potential results of simulation procedures.

Conclusion

Ontologies are quickly becoming an invaluable tool in computational biology. This is largely due to their expressiveness and their capacity for extension and enrichment without disruption to the end user. Ontologies are the perfect media to encode domain knowledge. Because different tools or approaches can share the same ontologies, they become the *de facto* glue between heterogeneous kinds of information, providing for a true integrative biology. We showed how using three different ontologies augments models and increases their

usability by software tools. Semantically improved models will provide more meaningful and reliable information, ultimately resulting in a richer pool of integrated data. However, even the best ontology is only a worthy effort until used. Encouraging a widespread use of SBO, KiSAO and TEDDY, as well as any future similar efforts is and will remain a challenge. With increased adoption, we expect to reach the tipping point. When, due to the amount of annotated models available, the benefits will outweigh the effort required for curation. The existence of coordinated efforts such as COMBINE (http://co.mbine.org/) may also help.

Acknowledgements

We thank the National Institute of General Medical Sciences, European Commission (FP7 SP4 Capacities Preparatory Phase 211601, ELIXIR) and Marie-Curie BioStar for providing resources to carry out this work.

Conflict of interest

The authors declare that they have no conflict of interest.

References

Ananiadou S, Kell DB, Tsujii J-I (2006) Text mining and its potential applications in systems biology. *Trends Biotechnol* **24:** 571–579

Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, Davis AP, Dolinski K, Dwight SS, Eppig JT, Harris MA, Hill DP, Issel-Tarver L, Kasarskis A, Lewis S, Matese JC, Richardson JE, Ringwald M, Rubin GM, Sherlock G (2000) Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat Genet* **25:** 25–29

Ausbrooks R, Buswell S, Carlisle D, Dalmas S, Devitt S, Diaz A, Froumentin M, Hunter R, Ion P, Kohlhase M, Miner R, Poppelier N, Smith B, Soiffer N, Sutor R, Watt SM (2003) Mathematical markup language (MathML) version 2.0. (2nd edn). World Wide Web Consortium, Recommendation REC-MathML2-20031021

Bialek W, Botstein D (2004) Introductory science and mathematics education for 21st-century biologists. *Science* **303:** 788–790

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

Chepelev LL, Dumontier M (2011) Semantic Web integration of Cheminformatics resources with the SADI framework. *J Cheminform* **3:** 16

Cook DL, Mejino JL, Neal ML, Gennari JH (2008) Bridging biological ontologies and biosimulation: the ontology of physics for biology. *AMIA Annu Symp Proc* **2008:** 136–140

Courtot M, Gibson F, Lister AL, Malone J, Schöber D, Brinkman RR, Ruttenberg A (2011) MIREOT: the minimum information to reference an external ontology term. *Appl Ontol* **6:** 23–33

Dada JO, Spasić I, Paton NW, Mendes P (2010) SBRML: a markup language for associating systems biology data with models. *Bioinformatics* **26:** 932–938

Day-Richter (2006) The OBO Flat File Format Specification, version 1.2 http://www.geneontology.org/GO.format.obo-1_2.shtml

Demir E, Cary MP, Paley S, Fukuda K, Lemer C, Vastrik I, Wu G, D'Eustachio P, Schaefer C, Luciano J, Schacherer F, Martinez-Flores I, Hu Z, Jimenez-Jacinto V, Joshi-Tope G, Kandasamy K, Lopez-Fuentes AC, Mi H, Pichler E, Rodchenkov I *et al* (2010) BioPAX – a community standard for pathway data sharing. *Nat Biotechnol* **28:** 935–942

Dräger A, Hassis N, Supper J, Schröder, Zell A (2008) SBMLsqueezer: a CellDesigner plug-in to generate kinetic rate equations for biochemical networks. *BMC Syst Biol* **2:** 39

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, Rodriguez N, Dumousseau M, Dörr A, Wrzodek C, Le Novère N, Zell A, Hucka M (2011) JSBML: a flexible Java library for working with SBML. *Bioinformatics* **27:** 2167–2168

Elowitz MB, Leibler S (2000) A synthetic oscillatory network of transcriptional regulators. *Nature* **403:** 335–338

Gleeson P, Crook S, Cannon RC, Hines ML, Billings GO, Farinella M, Morse TM, Davison AP, Ray S, Bhalla US, Barnes SR, Dimitrova YD, Silver RA (2010) NeuroML: a language for describing data driven models of neurons and networks with a high degree of biological detail. *PLoS Comput Biol* **17:** e1000815

Grau BC, Horrocks I, Kazakov Y, Sattler U (2007) Just the right amount: extracting modules from ontologies. In *Proceedings 16th Intl World Wide Web Conf*, Banff, Canada

Hucka M, Finney A, Sauro HM, Bolouri H, Doyle JC, Kitano H, Arkin AP, Bornstein BJ, Bray D, Cornish-Bowden A, Cuellar AA, Dronov S, Gilles ED, Ginkel M, Gor V, Goryanin II, Hedley WJ, Hodgman TC, Hofmeyr JH, Hunter PJ *et al* (2003) The Systems Biology Markup Language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics* **19:** 524–531

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

Kitano H (2002) Systems biology: a brief overview. *Science* **295:** 1662–1664

Knüpfer C, Beckstein C, Dittrich P (2006) Towards a semantic description of bio-models: meaning facets—a case study. In *Proc 2nd Intl Symp Semantic Mining Biomedicine*, Ananiadou S, Fluck J (eds). CEUR-WS, Aachen: RWTH University pp 97–100

Köhn D, Le Novère N (2008) SED-ML – an XML format for the implementation of the MIASE guidelines. Proc 6th conf Comput Meth Syst Biol (2008), Heiner M, Uhrmacher AM (eds). *Lect Notes Bioinfo* **5307:** 176–190

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

Lamprecht A-L, Naujokat S, Margaria T, Steffen B (2011) Semantics-based composition of EMBOSS services. *J Biomed Semantics* **2** (Suppl 1)**:** S5

Le Novère N, Finney A, Hucka M, Bhalla US, Campagne F, Collado-Vides J, Crampin EJ, Halstead M, Klipp E, Mendes P, Nielsen P, Sauro H, Shapiro B, Snoep JL, Spence HD, Wanner BL (2005) Minimum information requested in the annotation of biochemical models (MIRIAM). *Nat Biotechnot* **23:** 1509–1515

Le Novère N, Bornstein B, Broicher A, Courtot M, Donizelli M, Dharuri H, Li L, Sauro H, Schilstra M, Shapiro B, Snoep JL, Hucka M (2006) BioModels Database: a free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. *Nucleic Acids Res* **34:** D689–D691

Le Novère N, Courtot M, Laibe C (2007) Adding semantics in kinetics models of biochemical pathways. *Proc 2nd Intl Symp Exp Std Cond Enz*, Charact pp. 137–153. Available at http://www.beilstein-institut.de/index.php?id=196

Le Novère N (2008) Principled annotation of quantitative models in Systems Biology. *Genomes to Systems*, http://www.ebi.ac.uk/~lenov/LECTURES/G2S-LeNovere.pdf

Le Novère N, Hucka M, Mi H, Moodie S, Schreiber F, Sorokin A, Demir E, Wegner K, Aladjem MI, Wimalaratne SM, Bergman FT, Gauges R, Ghazal P, Kawaji H, Li L, Matsuoka Y, Villéger A, Boyd SE, Calzone L, Courtot M *et al* (2009) The systems biology graphical notation. *Nat Biotechnol* **27:** 735–741

Li C, Donizelli M, Rodriguez N, Dharuri H, Endler L, Chelliah V, Li L, He E, Henry A, Stefan MI, Snoep JL, Hucka M, Le Novère N, Laibe C (2010a) BioModels Database: an enhanced, curated and annotated resource for published quantitative kinetic models. *BMC Syst Biol* **4:** 92

Li C, Courtot M, Laibe C, Le Novère N (2010b) BioModels.net Web Services, a free and integrated toolkit for computational modelling software. *Brief Bioinfo* **11:** 270–277

Lin J, Keogh E, Wei L, Lonardi S (2007) Experiencing SAX: a novel symbolic representation of time series. *Data Min Knowl Discov* **15:** 107–144

Lister AL, Pocock M, Wipat A (2007) Integration of constraints documented in SBML, SBO, and the SBML Manual facilitates validation of biological models. *J Integr Bioinfo* **4:** 1–12

Lister AL, Pocock M, Taschuk M, Wipat A (2009) Saint: a lightweight integration environment for model annotation. *Bioinformatics* **25:** 3026–3027

Lister AL, Lord P, Pocock M, Wipat A (2010) Annotation of SBML models through rule-based semantic integration. *J Biol Sem* **1** (Suppl 1)**:** S3

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

May RM (2004) Uses and abuses of mathematics in biology. *Science* **303:** 790–793

Mungall DL, Bada M, Berardini TZ, Deegan J, Ireland A, Harris MA, Hill DP, Lomax J (2010) Cross-product extensions of the Gene Ontology. *J Biomed Info* **44:** 80–86

Noy NF, Shah NH, Whetzel PL, Dai B, Dorf M, Griffith N, Jonquet C, Rubin DL, Storey MA, Chute CG, Musen MA (2009) BioPortal: ontologies and integrated data resources at the click of a mouse. *Nucleic Acids Res* **37:** W170–W173

Rubin DL, Shah NH, Noy NF (2008) Biomedical ontologies: a functional perspective. *Brief Bioinfo* **9:** 75–90

Rosse C, Mejino JVL (2003) A reference ontology for biomedical informatics: the Foundational Model of Anatomy. *J Biomed Inform* **36:** 478–500

Schultz M, Krause F, Le Novère N, Klipp E, Liebermeister W (2011) Retrieval, alignment and clustering of computational models based on semantic annotations. *Mol Syst Biol* **7:** 512

Smith B (2003) Ontology. In *Blackwell Guide to the Philosophy of Computing and Information*, Floridi L (ed). Oxford: Blackwell, pp 155–166

Smith B, Ceusters W, Klagges B, Köhler J, Kumar A, Lomax J, Mungall C, Neuhaus F, Rector AL, Rosse C (2005) Relations in biomedical ontologies. *Genome Biol* **6:** R46

Smith B, Ashburner M, Rosse C, Bard J, Bug W, Ceusters W, Goldberg LJ, Eilbeck K, Ireland A, Mungall CJ, Leontis N, Rocca-Serra P, Ruttenberg A, Sansone S-A, Scheuermann RH, Shah N, Whetzel PL, Lewis S, The OBI Consortium (2007) The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nat Biotechnol* **25:** 1251–1255

Swainston N, Mendes P (2009) libAnnotationSBML: a library for exploiting SBML annotations. *Bioinformatics* **25:** 2292–2293

Swainston N, Golebiewski M, Messiha HL, Malys N, Kania R, Kengne S, Krebs O, Mir S, Sauer-Danzwith H, Smallbone K, Weidemann A, Wittig U, Kell DB, Mendes P, Müller W, Paton NW, Rojas I (2010) Enzyme kinetics informatics: from instrument to browser. *FEBS J* **277:** 3769–3779

Taylor CF, Field D, Sansone S-A, Aerts J, Apweiler R, Ashburner M, Ball CA, Binz P-A, Bogue M, Brazma A, Brinkman R, Clark AM, Deutsch EW, Fiehn O, Fostel J, Ghazal P, Gibson F, Gray T, Grimes G, Hardy NW *et al* (2008) Promoting coherent minimum reporting requirements for biological and biomedical investigations: the MIBBI project. *Nat Biotechnol* **26:** 889–896

Tenenbaum JD, Whetzel PL, Anderson K, Borromeo CD, Dinov ID, Gabriel D, Kirschner B, Mirel B, Morris T, Noy N, Nyulas C, Rubenson D, Saxman PR, Singh H, Whelan N, Wright Z, Athey BD, Becich MJ, Ginsburg GS, Musen MA *et al* (2011) The Biomedical Resource Ontology (BRO) to enable resource discovery in clinical and translational research. *J Biomed Infor* **44:** 137–145

W3C OWL working group (2009) OWL 2 Web Ontology Language Document Overview. http://www.w3.org/TR/owl2-overview/

Wittig U, Golebiewski M, Kania R, Krebs O, Mir S, Weidemann A, Anstein S, Saric J, Rojas I (2006) SABIO-RK: integration and curation of reaction kinetics data. *Lect Notes Comput Sci* **4075:** 94–103

6

Norme graphique pour la biologie des systèmes

Le Novère N, Hucka M, Mi H, Moodie S, Shreiber F, Sorokin A, Demir E, Wegner K, Aladjem M, Wimalaratne S, Bergman F T, Gauges R, Ghazal P, Kawaji H, Li L, Matsuoka Y, Villéger A, Boyd S E, Calzone L, Courtot M, Dogrusoz U, Freeman T, Funahashi A, Ghosh S, Jouraku A, Kim S, Kolpakov F, Luna A, Sahle S, Schmidt E, Watterson S, Goryanin I, Kell D B, Sander C, Sauro H, Snoep J L, Kohn K, Kitano H. The Systems Biology Graphical Notation. *Nature Biotechnology* (2009), 27 : 735-741

Résumé :

Circuits électriques et UML sont deux exemples de langages visuels normalisés qui ont aidé au progrès de leur domaine respectif en encourageant la régularité, en supprimant l'ambiguité et en permettant un support logiciel pour la communication d'information complexe. Ironiquement, bien qu'ayant un des rapports graphique/texte les plus importants, la biologie n'a toujours pas de notation graphique standard. Le déluge récent d'information biologique rend pressant le besoin de s'atteler à ce problème. À cette fin, nous présentons la *Systems Biology Graphical Notation* (SBGN), un langage visuel développé par une communauté de biochimistes, modélisateurs et informaticiens. SBGN est formés de trois langages complémentaires : *process descriptions*[1], *entity relationships* et *activity flows*. Ensemble, ces langages permettent aux scientifiques de représenter les réseaux d'interactions biochimiques d'une façon normalisée et non-ambigüe. Nous pensons que SBGN va favoriser les représentations, visualisations, stockages, échanges et réutilisations de manière efficace et précise, ce pour toute forme d'information biologique, des régulations de gènes au métabolisme, aux voies de signalisation cellulaire.

[1]le nom des langages n'a été formalisé qu'après la publication principale. Dans le résumé nous utilisons les noms officiels

PERSPECTIVE

The Systems Biology Graphical Notation

Nicolas Le Novère[1], Michael Hucka[2], Huaiyu Mi[3], Stuart Moodie[4], Falk Schreiber[5,6], Anatoly Sorokin[7], Emek Demir[8], Katja Wegner[9], Mirit I Aladjem[10], Sarala M Wimalaratne[11], Frank T Bergman[12], Ralph Gauges[13], Peter Ghazal[4,14], Hideya Kawaji[15], Lu Li[1], Yukiko Matsuoka[16], Alice Villéger[17,18], Sarah E Boyd[19], Laurence Calzone[20], Melanie Courtot[21], Ugur Dogrusoz[22], Tom C Freeman[14,23], Akira Funahashi[24], Samik Ghosh[16], Akiya Jouraku[24], Sohyoung Kim[10], Fedor Kolpakov[25,26], Augustin Luna[10], Sven Sahle[13], Esther Schmidt[1], Steven Watterson[4,22], Guanming Wu[27], Igor Goryanin[4], Douglas B Kell[18,28], Chris Sander[8], Herbert Sauro[12], Jacky L Snoep[29], Kurt Kohn[10] & Hiroaki Kitano[16,30,31]

Circuit diagrams and Unified Modeling Language diagrams are just two examples of standard visual languages that help accelerate work by promoting regularity, removing ambiguity and enabling software tool support for communication of complex information. Ironically, despite having one of the highest ratios of graphical to textual information, biology still lacks standard graphical notations. The recent deluge of biological knowledge makes addressing this deficit a pressing concern. Toward this goal, we present the Systems Biology Graphical Notation (SBGN), a visual language developed by a community of biochemists, modelers and computer scientists. SBGN consists of three complementary languages: process diagram, entity relationship diagram and activity flow diagram. Together they enable scientists to represent networks of biochemical interactions in a standard, unambiguous way. We believe that SBGN will foster efficient and accurate representation, visualization, storage, exchange and reuse of information on all kinds of biological knowledge, from gene regulation, to metabolism, to cellular signaling.

"*Un bon croquis vaut mieux qu'un long discours*" ("A good sketch is better than a long speech"), said Napoleon Bonaparte. This claim is nowhere as true as for technical illustrations. Diagrams naturally engage innate cognitive faculties[1] that humans have possessed since before the time of our cave-drawing ancestors. Little wonder that we find ourselves turning to them in every field of endeavor. Just as with written human languages, communication involving diagrams requires that authors and readers agree on symbols, the rules for arranging them and the interpretation of the results. The establishment and widespread use of standard notations have permitted many fields to thrive. One can hardly imagine today's electronics industry, with its powerful, visually oriented design and automation tools, without having first established standard notations for circuit diagrams. Such was not the case in biology[2]. Despite the visual nature of much of the information exchange, the field was permeated with *ad hoc* graphical notations having little in common between different researchers, publications, textbooks and software tools. No standard visual language existed for describing biochemical interaction networks, inter- and intracellular signaling gene regulation—concepts at the core of much of today's research in molecular, systems and synthetic biology. The closest to a standard is the notation long used in many metabolic and signaling pathway maps, but in reality, even that lacks uniformity between sources and suffers from undesirable ambiguities (**Fig. 1**). Moreover, the existing tentative representations, however well crafted, were ambiguous, and only suitable for specific needs, such as representing metabolic networks or signaling pathways or gene regulation.

The molecular biology era, and more recently the rise of genomics and other high-throughput technologies, have brought a staggering increase in data to be interpreted. It also favored the routine use of software to help formulate hypotheses, design experiments and interpret results. As a group of biochemists, modelers and computer scientists working in systems biology, we believe establishing standard graphical notations is an important step toward more efficient and accurate transmission of biological knowledge among our different communities. Toward this goal, we initiated the SBGN project in 2005, with the aim of developing and standardizing a systematic and unambiguous graphical notation for applications in molecular and systems biology.

Historical antecedents

Graphical representation of biochemical and cellular processes has been used in biochemical textbooks as far back as sixty years ago[3], reaching an apex in the wall charts hand drawn by Nicholson[4] and Michal[5]. Those graphs describe the processes that transform a set of inputs into a set of outputs, in effect being process, or state transition, diagrams. This style was emulated in the first database systems that depicted metabolic networks, including EMP[6], EcoCyc[7] and KEGG[8]. More notations have been 'defined' by virtue of their implementation in specialized software tools such as pathway and network designers (e.g., NetBuilder[9], Patika[10], JDesigner[11], CellDesigner[12]). Those

A list of affiliations appears at the end of the paper.

Published online 7 August 2009; doi:10.1038/nbt1558

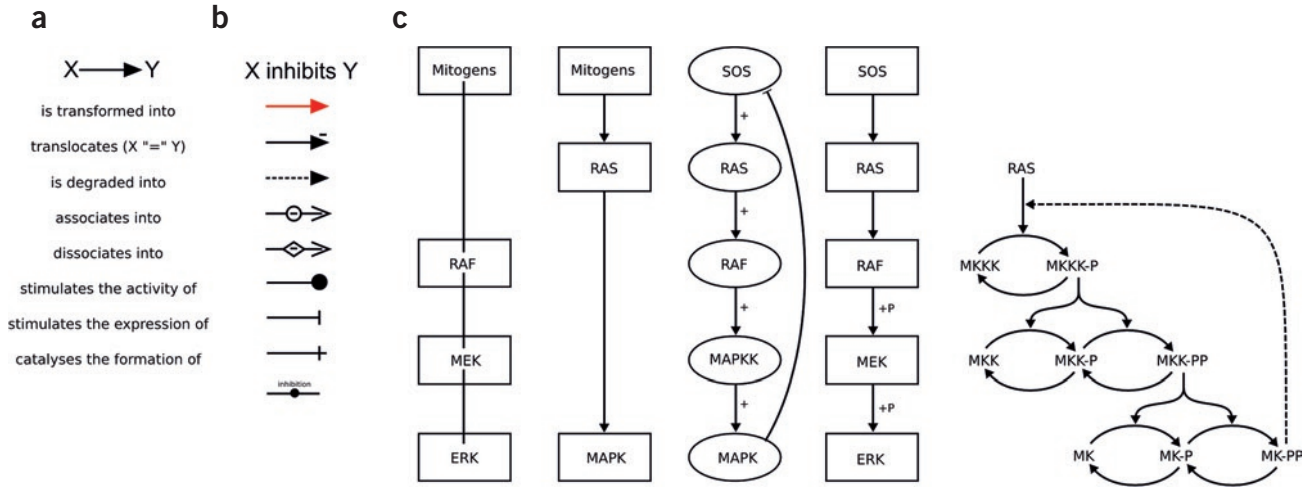


Figure 1 Inconsistency and ambiguity of current nonstandardized notations. (**a**) Eight different meanings associated with the same symbol in a chart describing the role of cyclin in cell regulations (http://www.abcam.com/ps/pdf/nuclearsignal/cell_cycle.pdf). (**b**) Nine different symbols found in the literature to represent the same meaning. (**c**) Five different representations of the MAP kinase cascade found in the scientific literature, depicting progressive levels of biological and biochemical knowledge. From left to right: relations[30], directionality of influence[31], directionality of effect[32], biochemical effect[33], chemical reactions[34]. In the last diagram, different instances of an identical arrowhead style represent catalysis, production and inhibition.

graphical notations were not standardized, and their understanding relied mainly on relating examples with one's preexisting knowledge of biochemical processes. Although the classical graphs adequately conveyed information about biochemistry, other types of diagrams were needed to represent signaling pathways, and incomplete or indirect information, as coming from molecular biology or genomics. Those conventions effectively mimicked the empirical notations used by biologists, describing either the relationships between elements[13,14] or the flow of activity or influence[15–17]. Lists of standard glyphs (**Box 1**) to represent identified concepts were then provided. The efforts to create rigidly defined schema were pioneered by Kurt Kohn with his Molecular Interaction Maps (MIM), which defined not only a set of symbols but also a syntax to describe interactions and relationships of molecules[18,19]. The MIM notation influenced other proposals[14]. Several proposals followed to describe process diagrams, not only with standard symbols but also defined grammars[20–23].

The SBGN project

Despite the popularity of some of the efforts mentioned above, none of the notations has acquired the status of a community standard. This can be attributed partly to the fact that the efforts only went as far as to propose notations, or implement them in software. Several of us have been involved in the development of the Systems Biology Markup Language (SBML)[24], from which we learned that establishing a standard is extremely difficult without an explicit, concerted, effort to engage a community and build a consensus among participants. We organized the SBGN project with this lesson in mind.

For SBGN to be successful, it must satisfy a majority of technical and practical needs and be embraced by a diverse community of biologists, biochemists, bioinformaticians, geneticists, theoreticians and software engineers. Early in the project's history, we established the following overarching principles to help steer SBGN toward those aims, ranked by rough hierarchical order of precedence.

The notation should
- be free of intellectual property restrictions to allow free use by the community;
- be syntactically and semantically consistent and unambiguous;
- support representation of diverse common biological objects, their properties and their interactions;
- keep the number of symbols and syntax to a minimum to help comprehension and learning by humans;
- be visually consistent and concise, using discriminable symbols;
- support modularity to help cope with diagram size and complexity;
- support the automated generation of diagrams by software starting from mathematical models.

Many of the design principles above resonate with research on visual languages[25,26] and studies aimed at understanding end-user needs in pathway visualization[27], although we derived them from our collective hands-on experiences with developing notations and software. In addition to these principles, we also sought to avoid many problems (**Table 1**) that affect some existing notations.

SBGN aims to specify the connectivity of the graphs and the types of the nodes and edges, but not the precise layout of the graphs. The semantics of an SBGN diagram does not depend on the relative position of the symbols. Furthermore, it does not depend on colors, patterns, shades, shapes and thickness of edges (**Fig. 2**). Similarly, the labels of symbols are not regulated and are only required to be unique within a map.

Finally, it was clear at the outset that it would be impossible to design a perfect and complete language from the beginning. Apart from the prescience this would require, it also would likely require a vast language that most newcomers would shun as being too complex. Thus, the SBGN community decided to stratify language development into levels. A level in SBGN represents a usable set of functionalities that the user community agrees is sufficient for a reasonable set of tasks and goals. Capabilities and features that cannot be agreed upon and are judged insufficiently critical to require inclusion in a given level are postponed to a higher level. In this way, SBGN development is envisioned to proceed in stages, with each higher SBGN level adding richness compared to the levels below it, while maintaining compatibility whenever possible. Furthermor only the actual usage of SBGN languages will tell us how well they work for the diverse communities involved, and this experience will certainly shape the evolution of the notation.

The three languages of SBGN

Molecular entities possess many properties that affect their interactions with other entities. Attempting to represent all the possible reactions and interactions in the same diagram is often futile, usually resulting in an incomprehensible jumble. The different styles of notations described above were attempts to control this complexity by presenting only what was needed in a specific context, or what was available through specific views of the system[14]. Each view focuses on only a portion of the semantics of the overall system, trading off diagram comprehensibility against completeness of biological knowledge.

SBGN follows this strategy and defines three orthogonal and complementary types of diagrams that can be seen as three alternative projections of the underlying more complex biological information. The process diagram draws its inspiration from process-style notations, borrowing ideas from the work of CellDesigner[28] and EPE[22]. By contrast, the entity relationship diagram is based to a large extent on Kohn's MIM notation[18,19]. The SBGN activity flow diagram depicts only the cascade of activity, thus making the notation similar to the reduced representations often used in the current literature to describe signaling pathways and gene regulatory networks. In **Figure 2**, we illustrate the three views applied to a very simple example. The characteristics of the SBGN languages are summarized in **Table 2**.

The idea of having three diagram types naturally begs the question of whether they could be merged into one, at least in paper form. The answer is no, for at least two reasons. First, a single diagram type would bring us back to the problem of dealing with unreasonable numbers of interactions as described above. Second, each SBGN language reflects fundamental differences in the underlying formal description of the phenomena. The meanings are so different that merging diagram types would compromise their representational robustness.

Having multiple visual languages is not uncommon in engineering (consider, for example, block diagrams and circuit diagrams in electronics, UML class, state sequence and deployment diagrams in software engineering), and this supports the idea that having three sublanguages in SBGN will be manageable in practice. In SBGN, the sharing of symbols representing identical concepts further reduces the differences between the three languages to differences in syntax and semantics. We believe that this, combined with careful design, will mitigate some of the difficulties of learning SBGN. However, it is to be noted that the clean orthogonality of the languages makes their overlap very limited, mostly to modulatory arcs, and node decorations.

Box 1 Glossary

SBGN diagrams are a specific set of graphs and thus make use of concepts from graph theory. The following list defines the terms used most often. We are aware of the unavoidable circularity of such definitions.

- **Arc**. A directed edge, that is, an edge that is not symmetrical in shape.
- **Edge**. A line joining two nodes.
- **Glyph**. A symbol that conveys information nonverbally.
- **Graph**. A set of nodes connected with edges.
- **Node**. A point that terminates a line or curve or comprises the intersection of two or more lines or curves.

SBGN process diagram

A process diagram represents all the molecular processes and interactions taking place between biochemical entities, and their results. This type of diagram depicts how entities transition from one form to another as a result of different influences; thus, it portrays the temporal qualities of molecular events occurring in biochemical reactions. In this way, the approach underlying process diagrams is the same as in the familiar textbook drawings of metabolic pathways. The main drawback of process diagrams is that a given entity must appear multiple times in the same diagram if it exists under several states; therefore, the notation is sensitive to the combinatorial explosion of possible entities and reactions, as is often the case in signaling pathways.

The SBGN process diagram level 1 specification defines six major classes of glyphs: entity pool nodes, process nodes, container nodes, reference nodes, connecting arcs and logical operators (**Supplementary Note 1**). In **Figure 3a**, we show a complete example of an SBGN process diagram. The number of symbols in level 1 of the SBGN process diagram notation has been purposefully limited so that they could be easily memorized. The notation may be enriched (perhaps using subclasses of symbols) in higher levels of SBGN.

Table 3 lists software projects that are already developing support for SBGN process diagram level 1 (see also **Supplementary Note 2**). Some of these rely on manual design of the pathways, whereas others, such as Arcadia, automatically generate SBGN PD from SBML models that have been annotated with terms from the Systems Biology Ontology[29]. The encoding of SBGN diagrams using computer-readable formats, a crucial step toward exchange and reuse of SBGN

Table 1 Features of *ad hoc* graphical notations, and the problems they create

Feature	Problem(s)
Different line thicknesses distinguish different types of processes or elements Dotted or dashed line styles distinguish different types of processes or elements	1. Rescaling a diagram can make line thicknesses and styles impossible to discern 2. Photocopying or faxing a diagram can cause differences in line thicknesses and styles to disappear 3. Differences in line thickness and style are difficult to make consistent in diagrams drawn by hand
Different colors distinguish different types of processes or elements	1. Photocopying or faxing a diagram will cause color differences to be indistinguishable 2. Color characteristics are difficult to achieve and keep consistent when drawing diagrams by hand
Identical line terminators (e.g., a single arrow) indicate different effects or processes depending on context	1. Greater ambiguity is introduced into a diagram 2. Interpreting a diagram requires more thought on the part of the reader 3. Automated verification of diagrams is more difficult due to lack of distinction between different processes or elements
Ad hoc symbols introduced at will by author	Interpreting a diagram requires the reader to search for additional information explaining the meaning of the symbols

diagrams, is currently supported in different formats such as SBML, GML and GraphML by different tools, and a general XML-based exchange format for SBGN is currently under discussion.

SBGN entity relationship diagram

The SBGN notation for entity relationships puts the emphasis on the influences that entities have upon each other's transformations rather than the transformations themselves. One can imagine that each of the relationships represents a specific conclusion of a scientific experiment or article. Their addition on a map represents the knowledge we have of the effects the entities have upon each other. Contrary to the process diagrams, where the different processes affect each other, the relationships are independent, and this independence is the key to avoiding the combinatorial explosion inherent to process diagrams. Unlike in process diagrams, a given entity may appear only once. Readers can better grasp at first sight all the possible influences and interactions affecting an entity, without having to explore the whole diagram to discover the different states an entity may be in, or to trace all the edges to find the relevant process nodes.

The relationship symbols in entity relationship diagrams support the representation of interactions and state variable assignments, thus allowing the notation to describe certain processes that cannot be expressed in process diagrams, such as allosteric modulation. In process diagrams, one can represent the formation of a ligand-receptor complex, but it is not possible to state that the complex is more active than the receptor alone without additional markup; entity relationships support this by allowing the interaction with the ligand to modulate the assignment of the variable representing the activity. The trade-off is that the temporal course is difficult to follow in entity relationships, because the sequence of events is not explicitly described (**Fig. 2a**,**b**).

The specification of SBGN entity relationship diagram level 1 defines three major classes of glyphs: entity nodes, statements and influences (logical operators are entity nodes). We summarize the symbols and the rules for their assembly (**Supplementary Note 3**). In **Fig. 3b**, we show a complete example of an SBGN entity relationship diagram.

SBGN activity flow diagram

A strategy often used for coping with biochemical network complexity or with incomplete or indirect knowledge is to selectively ignore the biochemical details of processes, instead representing the influences between entities directly. SBGN's activity flow diagrams permit modulatory arcs to directly link different activities, rather than entities and processes or relationships as described previously. Instead of displaying the details of biochemical reactions with process nodes and connecting arcs, the activity flow diagrams show only influences such as 'stimulation' and 'inhibition' between the activities displayed by the molecular entities (**Fig. 2c**). For example, a signal 'stimulates' the activity of a receptor, and this activity in turn 'stimulates' the activity of an intracellular transducing protein (note that activity flow retains the sequential chains of influences). Because most signaling pathway diagrams in the current literature are essentially activity flow diagrams, we expect many biologists will find this type of diagram familiar.

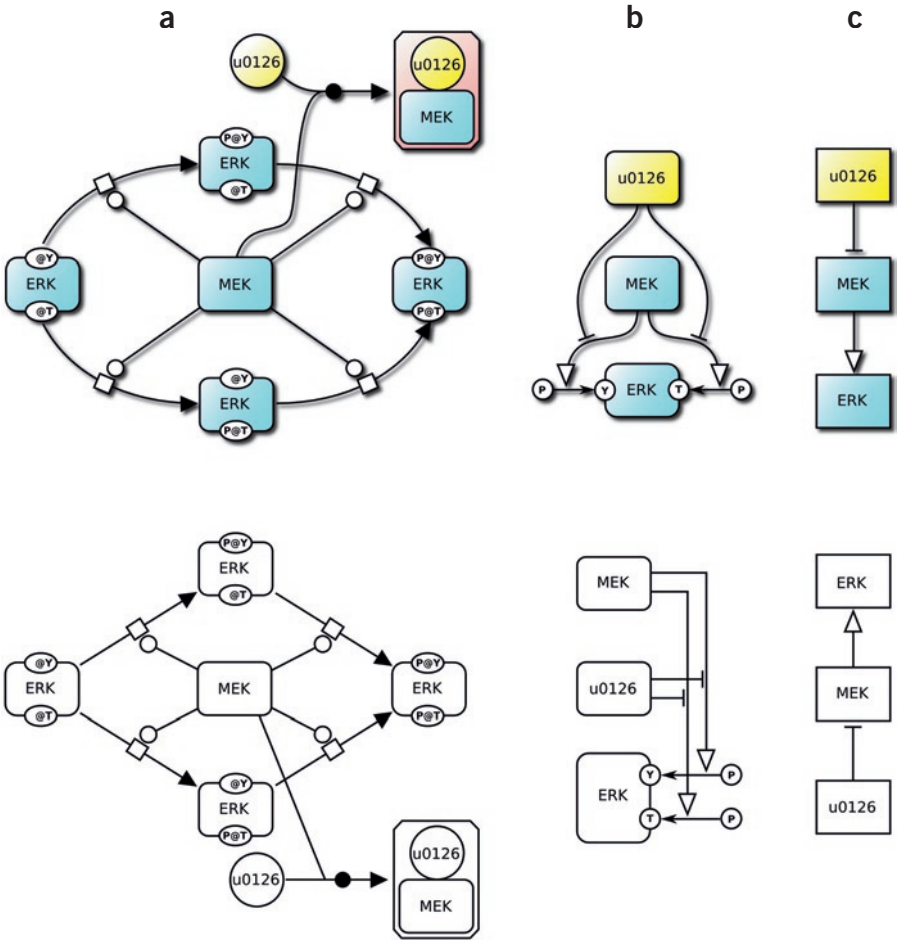


Figure 2 Simple example of protein phosphorylation catalyzed by an enzyme and modulated by an inhibitor. The semantics of an SBGN diagram does not depend on the relative position of the symbols, or on colors, patterns, shades, shapes and thickness of edges. Therefore, the upper and lower diagrams are identical as far as SBGN is concerned, and have to be interpreted exactly the same way. (**a**) Process diagrams, explicitly displaying the four forms of ERK, phosphorylated and nonphosphorylated on the tyrosine and the threonine, as well as the processes of phosphorylation by MEK and the inhibition of MEK by complexation with u0126. Note that the inhibition in this diagram emerges from the sequestration of MEK and is not explicitly represented. The phosphorylation sites are represented by variables, which in this example are labeled simply as 'Y' and 'T' (but in general could be anything desired by the diagram author), shown adorning the main symbols for ERK. (**b**) Entity relationship diagrams, showing ERK and the assignment of its phosphorylations (at the tyrosine and threonine residues), as well as the relationships between those and MEK and u0126. Note that ERK appears only once in this diagram; the different possible states are not explicitly depicted. (**c**) Activity flow diagrams depicting the activation of ERK by MEK and the inhibition of MEK by u0126. In this notation, only the relevant activities of u0126, MEK and ERK are represented, as well as abstract representations of the influences of activities upon each other, whereas the biochemical details are omitted.

Table 2 Comparison between the three languages of SBGN

	Process diagram	Entity relationship diagram	Activity flow diagram
Purpose	Represent processes that convert physical entities into other entities, change their states or change their location	Represent the interactions between entities and the rules that control them	Represent the influence of biological activities on each other
Building block	Different states of physical entities are represented separately	Physical entities are represented only once	Different activities of physical entities are represented separately
Ambiguity	Unambiguous transcription into biochemical events	Unambiguous transcription into biochemical events	Ambiguous interpretation in biochemical terms
Level of description	Mechanistic descriptions of processes	Mechanistic description of relationships	Conceptual description of influences
Temporality	Representation of sequential events	Absence of sequentiality between events	Representation of sequential influences
Pitfalls	Sensitive to combinatorial explosion of states and processes	Creation, destruction and translocation are not easily represented	Not suitable to represent association, dissociation, multistate entities
Advantages	The best for representing temporal/mechanistic aspects of processes such as metabolism	The best for representing signaling involving multistate entities	The best for functional genomics and signaling with simple activities

By ignoring processes and entity states, the number of nodes in an activity flow diagram is greatly reduced compared to an equivalent process diagram (**Fig. 2a,c**). Activity flow diagrams are also especially convenient for representing the effects of perturbations, whether genetic or environmental, because the complete mechanisms of the perturbations may not be known, or are irrelevant to the goals of a given study. The drawback is that activity flow diagrams may contain a high level of ambiguity. For instance, the biochemical basis of a positive or negative influence in a given system is left undefined. For this reason, this type of SBGN diagram should not exist alone; it should be associated, when possible, with detailed entity relationship and process diagrams, and used only for viewing purposes. We expect it will often be possible to generate activity flow diagrams mechanically from process diagrams and entity relationships, and have already performed preliminary work in that direction.

The SBGN activity flow diagram level 1 specification defines four major classes of glyphs: activity nodes, container nodes, modulating arcs and logical operators (**Supplementary Note 4**). **Figure 3c** shows a complete example of an SBGN activity flow diagram.

Participation and future prospects

The SBGN website (http://sbgn.org/) is a portal for all things related to SBGN. Interested persons can get involved in SBGN discussions by joining the SBGN discussion list (sbgn-discuss@sbgn.org). Face-to-face meetings of the SBGN community, generally held as satellite workshops of larger conferences, are announced on the website as well as the mailing list.

Standardizing a notation for depicting networks of biochemical interactions has so far remained an elusive goal, despite numerous but isolated efforts in that direction. Only with such a standardized notation will biologists, modelers and computer scientists be able to exchange accurate descriptions of complex systems—a task that continues to grow more demanding as our collective knowledge expands. SBGN blends many influences from past efforts, and also introduces many new ideas designed to overcome limitations of other notations.

Using a community-based approach involving many interested groups and individuals (including some who have been involved in previous efforts), we have developed and released the first version of the three languages of the SBGN, the process diagram, the entity relationship diagram and the activity flow diagram.

Future levels of the three languages should address major challenges currently faced by the systems biology community, as the field matures and diversifies. To cite but a few examples, the representation of spatial structures and spatial events, of composed and modular models, and of dynamic creation or destruction of compartments remains unchartered territory.

Table 3 List of software systems known to provide support, or to be in the process of developing support, for SBGN

Name	Organization	Link
Arcadia	Manchester Centre for Integrative Systems Biology, Manchester, UK	http://arcadiapathways.sourceforge.net/
Athena	University of Washington, Seattle, WA, USA	http://www.codeplex.com/athena/
BioModels Database	European Bioinformatics Institute, Cambridge, UK	http://www.ebi.ac.uk/biomodels/
BioUML	Institute of Systems Biology, Novosibirsk, Russia	http://www.biouml.org/
ByoDyn	Institut Municipal d'Investigació Mèdica, Barcelona, Spain	http://byodyn.imim.es/
CellDesigner	The Systems Biology Institute, Tokyo	http://www.celldesigner.org/
Dunnart	Monash University, Melbourne, Australia	http://www.csse.monash.edu.au/~mwybrow/dunnart/
Edinburgh Pathway Editor	Edinburgh Centre for Bioinformatics, Edinburgh, UK	http://www.pathwayeditor.org/
JWS Online	Stellenbosch University, Stellenbosch, South Africa	http://jjj.biochem.sun.ac.za/
NetBuilder	STRI, University of Hertfordshire, Hatfield, UK	http://strc.herts.ac.uk/bio/maria/Apostrophe/
PANTHER	Artificial Intelligence Center, SRI international, Menlo Park, CA, USA	http://www.pantherdb.org/pathway/
Reactome	European Bioinformatics Institute, Cambridge, UK	http://www.reactome.org/
Vanted	IPK Gatersleben, Gatersleben, Germany	http://vanted.ipk-gatersleben.de/
VISIOweb	Bilkent University, Ankara, Turkey	http://www.bilkent.edu.tr/~bcbi/pvs.html

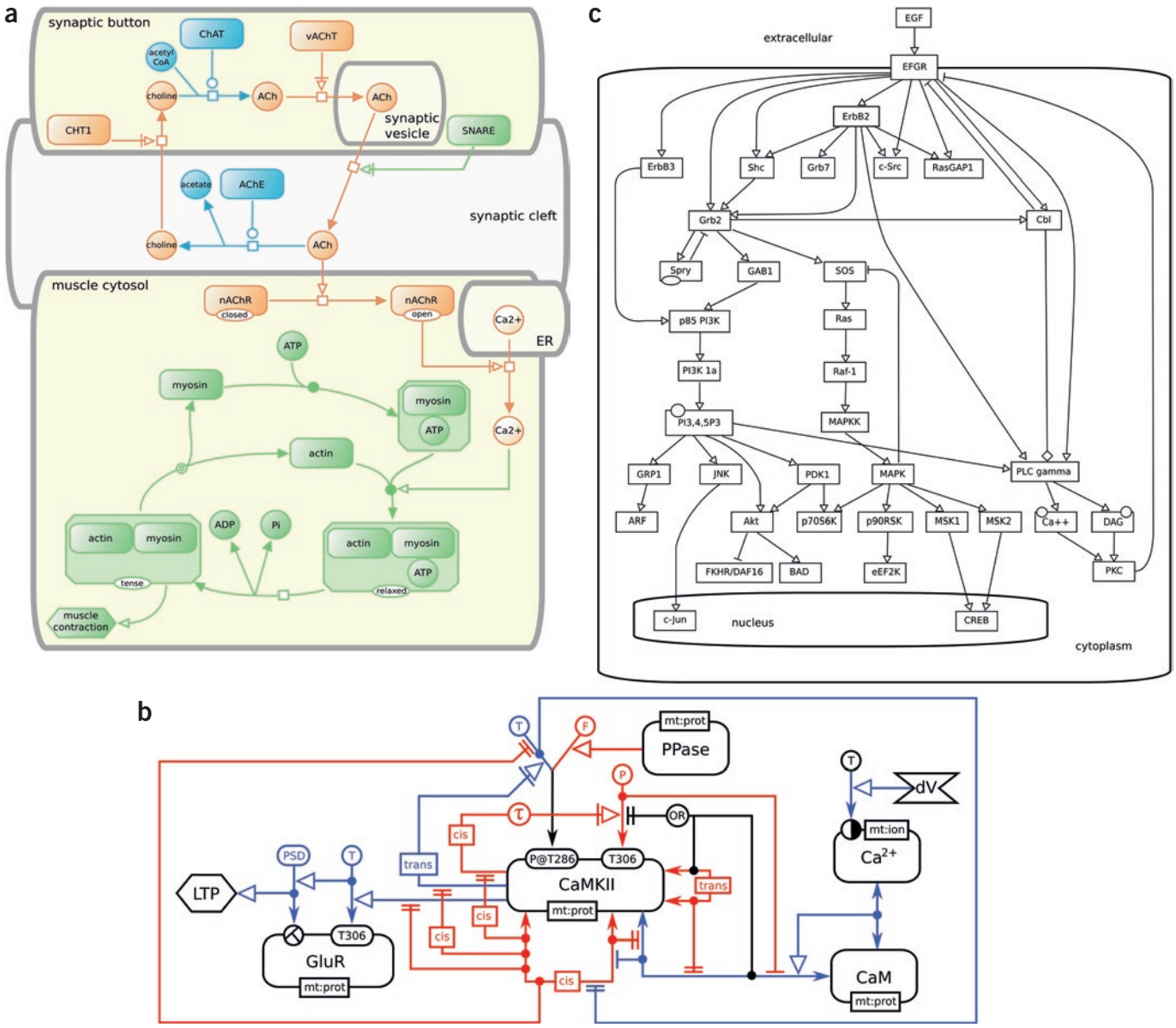


Figure 3 Example of complete SBGN diagrams. (**a**) Process diagram representing the synthesis of the neurotransmitter acetylcholine in the synaptic button of a nerve terminal, its release in the synaptic cleft, degradation in the synaptic cleft, the post-synaptic stimulation of its receptors and the subsequent effect on muscle contraction. Colors are used to enhance the biological semantics, blue representing catalytic reactions, orange for transport between compartments (including unrepresented ions, through channels) and green for the function of contractile proteins. However, it is important to note that those colors are not part of SBGN process diagram notation, and must not change the interpretation of the graph. (**b**) SBGN entity relationship diagram representing the transduction, by calcium/calmodulin kinase II, of the effect of voltage-induced increase of intracellular calcium onto the long-term potentiation (LTP) of the neuronal synapses, triggered by a translocation of glutamate receptors. The diagram describes the various relationships between the phosphorylations of the kinase monomers and their conformation. Colors highlight the direction of the relationships relative to the phenotype; blue relationships enhance LTP whereas red ones preclude this enhancement. (**c**) SBGN activity flow diagram representing the cascade of signals triggered by the epidermal growth factor, and going from the plasma membrane to the nucleus. The diagram is derived from reference 30.

Note: Supplementary information is available on the Nature Biotechnology website.

ACKNOWLEDGMENTS

The development of SBGN was mainly supported by a grant from the New Energy and Industrial Technology Development Organization of the Japanese government. SBGN workshops also benefited from funding by the following organizations: the UK Biotechnology and Biological Sciences Research Council, the National Institute of Advanced Industrial Science and Technology of Japan, the Okinawa Institute of Science and Technology), the European Media Laboratory, Heidelberg, Germany and the Beckman Institute at the California Institute of Technology, Pasadena, California, USA. Attendance at the meetings by Japanese authors was supported by the Japan Science and Technology Agency and by the genome network project of the Japanese Ministry of Education, Culture, Sports, Science, and Technology. I.G., S.M. and A.S. acknowledge support by the British Engineering and Physical Sciences Research Council. F.T.B. acknowledges support by the National Institutes of Health (NIH; grant 1R01GM081070-01). The contributions of M.I.A., S.K., A.L. and K.K. were supported by the Intramural Research Program of the NIH, Center for Cancer Research, National Cancer Institute (NIH).

AUTHOR CONTRIBUTIONS

N.L.N., M.H., H.M., S.M., F.S. and A.S. contributed equally to the redaction of SBGN specifications.

Published online at http://www.nature.com/naturebiotechnology/.

1. Larkin, J.H. & Simon, H.A. Why a diagram is (sometimes) worth ten thousands words. *Cogn. Sci.* **11**, 65–100 (1987).
2. Lazebnik, Y. Can a biologist fix a radio?—Or,what I learned while studying apoptosis. *Cancer Cell* **2**, 179–182 (2002).
3. Gortner, R.A. *Outlines of Biochemistry*. (Wiley, New York, 1949).
4. Dagley, S. & Nicholson, D.E. *An Introduction to Metabolic Pathways* (Wiley, New York, 1970).
5. Michal, G. Biochemical Pathways (wall chart). (Boehringer Mannheim, Mannheim, Germany, 1984)
6. Sel'kov, E.E., Goryanin, I.I., Kaimatchnikov, N.P., Shevelev, E.L. & Yunus, I.A. Factographic data bank on enzymes and metabolic pathways. *Studia Biophysica* **129**, 155–164 (1989).
7. Karp, P.D. & Paley, S.M. Representations of metabolic knowledge: pathways. *Proc. Int. Conf. Intell. Syst. Mol. Biol.* **2**, 203–211 (1994).
8. Goto, S. *et al.* Organizing and computing metabolic pathway data in terms of binary relations. *Pac. Symp. Biocomput.* **PSB97**, 175–186 (1997).
9. Brown, C.T. *et al.* New computational approaches for analysis of cis-regulatory networks. *Dev. Biol.* **246**, 86–102 (2002).
10. Demir, E. *et al.* PATIKA: an integrated visual environment for collaborative construction and analysis of cellular pathways. *Bioinformatics* **18**, 996–1003 (2002).
11. Sauro, H.M. *et al.* Next generation simulation tools: the Systems Biology Workbench and BioSPICE integration. *OMICS* **7**, 355–372 (2003).
12. Funahashi, A., Morohashi, M., Kitano, H. & Tanimura, N. CellDesigner: a process diagram editor for gene-regulatory and biochemical networks. *BIOSILICO* **1**, 159–162 (2003).
13. Kohn, K.W. Functional capabilities of molecular network components controlling the mammalian G1/S cell cycle phase transition. *Oncogene* **16**, 1065–1075 (1998).
14. Kitano, H. A graphical notation for biochemical networks. *BIOSILICO* **1**, 169–176 (2003).
15. Pirson, I. *et al.* The visual display of regulatory information and networks. *Trends Cell Biol.* **10**, 404–408 (2000).
16. Cook, D.L., Farley, J.F. & Tapscott, S.J. A basis for a visual language for describing, archiving and analyzing functional models of complex biological systems. *Genome Biol.* **2**, research0012.1–0012.10 (2001).
17. Longabaugh, W.J.R., Davidson, E.H. & Bolouri, H. Visualization, documentation, analysis, and communication of large-scale gene regulatory networks. *Biochim. Biophys. Acta* **1789**, 363–374 (2009).
18. Kohn, K.W. Molecular interaction map of the mammalian cell cycle control and DNA repair systems. *Mol. Biol. Cell* **10**, 2703–2734 (1999).
19. Kohn, K.W., Aladjem, M.I., Weinstein, J.N. & Pommier, Y. Molecular interaction maps of bioregulatory networks: a general rubric for systems biology. *Mol. Biol. Cell* **17**, 1–13 (2006).
20. Demir, E. *et al.* An ontology for collaborative construction and analysis of cellular pathways. *Bioinformatics* **20**, 349–356 (2004).
21. Kitano, H., Funahashi, A., Matsuoka, Y. & Oda, K. Using process diagrams for the graphical representation of biological networks. *Nat. Biotechnol.* **23**, 961–966 (2005).
22. Moodie, S.L., Sorokin, A.A., Goryanin, I.I. & Ghazal, P. A graphical notation to describe the logical interactions of biological pathways. *J. Integr. Bioinform.* [Au: Correct publication is only one page?] **3**, 36–46 (2006).
23. Raza, S. A logic-based diagram of signalling pathways central to macrophage activation. *BMC Syst. Biol.* **2**, 36 (2008).
24. Hucka, M. *et al.* The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics* **19**, 524–531 (2003).
25. Britton, C., Jones, S., Kutar, M., Loomes, M. & Robinson, B. Evaluating the intelligibility of diagrammatic languages used in the specification of software. in *Theory and Application of Diagrams: Diagrams 2000* (eds. Anderson, M., Cheng, P. & Haarslev, V.) 376–391 (Springer, New York, 2000).
26. Narayanan, N.H. & Hübscher, R. Visual language theory: towards a human-computer interaction perspective. in *Visual Language Theory* (eds. Marriott, K. & Meyer, B.) 85–127 (Springer, New York, 1998).
27. Saraiya, P., North, C. & Duca, K. Visualizing biological pathways: requirements analysis, systems evaluation and research agenda. *Inform. Visual* **4**, 191–205 (2005).
28. Funahashi, A. *et al.* CellDesigner 3.5: a versatile modeling tool for biochemical networks. *Proc. IEEE* **96**, 1254–1265 (2008).
29. Le Novère, N., Courtot, M. & Laibe, C. Adding semantics in kinetics models of biochemical pathways. in *Proc. 2nd Intl. Symp. Exp. Stand. Cond. Enzyme Characterizations*, Rüdesheim, Germany, March 19–23, 2006 (Beilstein Institute, Frankfurt, 2007). <http://www.beilstein-institut.de/index.php?id=196>
30. Anonymous. MAP kinases. Gene set bank. *Riken BioResource Center DNA Bank* <http://www.brc.riken.go.jp/lab/dna/en/GENESETBANK/index.html> (August 19, 2008).
31. Riechelmann, H. Cellular and molecular mechanisms in environmental and occupational inhalation toxicology. *GMS Cur. Topics. Otorhinolaryngol.—Head Neck Surg.* **3**, Doc02 (2004).
32. Schlessinger, J. Epidermal growth factor receptor pathway. *Sci. Signal.* (connections map in the database of cell signaling, as seen May 29, 2009). <http://stke.sciencemag.org/cgi/cm/stkecm;CMP_14987>
33. Anonymous. MAPK signaling pathway, *Homo sapiens. KEGG Pathway hsa04010* <http://www.genome.jp/kegg/pathway/hsa/hsa04010.html> (July 15, 2009).
34. Kholodenko, B. Negative feedback and ultrasensitivity can bring about oscillations in the mitogen-activated protein kinase cascades. *Eur. J. Biochem.* **267**, 1583–1588 (2000).

[1]EMBL European Bioinformatics Institute, Hinxton, UK. [2]Engineering and Applied Science, California Institute of Technology, Pasadena, California, USA. [3]SRI International, Menlo Park, California, USA. [4]Centre for Systems Biology at Edinburgh, University of Edinburgh, Edinburgh, UK. [5]Leibniz Institute of Plant Genetics and Crop Plant Research, Gatersleben, Germany. [6]Institute of Computer Science, University of Halle, Halle, Germany. [7]School of Informatics, University of Edinburgh, Edinburgh, UK. [8]Memorial Sloan Kettering Cancer Center - Computational Biology Center, New York, NY, USA. [9]Science and Technology Research Institute, University of Hertfordshire, Hatfield, UK. [10]National Cancer Institute, Bethesda, Maryland, USA. [11]Auckland Bioengineering Institute, University of Auckland, Auckland, New Zealand. [12]Department of Bioengineering, University of Washington, Seattle, Washington, USA. [13]BIOQUANT, University of Heidelberg, Heidelberg, Germany. [14]Division of Pathway Medicine, University of Edinburgh Medical School, Edinburgh, UK. [15]Riken OMICS Science Center, Yokohama City, Kanagawa, Japan. [16]The Systems Biology Institute, Tokyo, Japan. [17]School of Computer Science, University of Manchester, Manchester, UK. [18]Manchester Interdisciplinary Biocentre, Manchester, UK. [19]Clayton School of Information Technology, Faculty of Information Technology, Monash University, Melbourne, Victoria, Australia. [20]U900 INSERM, Paris Mines Tech, Institut Curie, Paris, France. [21]Terry Fox Laboratory, British Columbia Cancer Research Center, Vancouver, British Columbia, Canada. [22]Bilkent Center for Bioinformatics, Bilkent University, Ankara, Turkey. [23]The Roslin Institute, University of Edinburgh, Midlothian, UK. [24]Department of Biosciences and Informatics, Keio University, Hiyoshi, Kouhoku-ku, Yokohama, Japan. [25]Institute of Systems Biology, Novosibirsk, Russia. [26]Design Technological Institute of Digital Techniques SB RAS, Novosibirsk, Russia. [27]Ontario Institute for Cancer Research, Toronto, Ontario, Canada. [28]School of Chemistry, University of Manchester, Manchester, UK. [29]Department of Biochemistry, Stellenbosch University, Matieland, South Africa. [30]Sony Computer Science Laboratories, Tokyo, Japan. [31]Okinawa Institute of Science and Technology, Okinawa, Japan. Correspondence should be addressed to N.L.N. (lenov@ebi.ac.uk).

Supplementary Note 1:
Symbols Used SBGN Process Diagram Level 1

The SBGN Process Diagram specification defines a comprehensive set of symbols with precise semantics, together with detailed syntactic rules defining their use. It also describes how such graphical information is to be interpreted. The essence of a process diagram is *change*. It shows how different entities in a system transition from one form to another. Glyphs (symbols) are the graphical units that represent concepts in SBGN. Each one is uniquely identified by a term from the Systems Biology Ontology (SBO). There are two types of glyphs in SBGN: nodes (which are subdivided into entity pool nodes, container nodes, process nodes, and logical operator nodes), and arcs (edges) that characterize the quantitative effect of a substance on a process or vice versa.

Entity pool nodes *(EPNs)* represent ensembles of entities, such as molecules, that are considered indistinguishable from each other in the context of a given graph. Level 1 of the SBGN Process Diagram defines six distinct glyphs for the following concepts: *unspecified entity*, *simple chemical*, *macromolecule*, *nucleic acid feature*, *perturbing agent* (such as light, temperature, etc.) and *source and sink*. The *EPNs* associated with molecular entities can be duplicated and stacked to represent multimers of identical elements. An additional construct, the *complex*, can be also used as an *EPN*. The semantics of *EPNs* can be modified by *auxiliary units*, which represent a particular state, the fact that the *EPN* has been cloned in the maps, or some additional information that may be encoded using controlled vocabularies. Finally, *tags* can be used to identify an *EPN* used in two or more physically different maps, thereby allowing the modular decomposition of diagrams.

Process nodes *(PNs)* describe the way in which *EPNs* are transformed into other *EPN*s. SBGN Process Diagram Level 1 defines five *PNs*: *process* (used to represent most of the transformations between *EPNs*), *omitted process* (when several transitions are known to exist but not represented), *uncertain process* (when the transition may or may not exist), *association*, and *dissociation* (representing the

Nature Biotechnology: doi:10.1038/nbt.1558

whereabouts of non-covalent complexes). In addition, a particular type of PN is the *phenotype*, which can be modulated but does not consume or produce anything. More types of transition may be defined in the future in higher levels of the SBGN Process Diagram notation.

Connecting arcs link *EPNs* and *PNs*, and indicate how entities influence processes. In addition to *consumption* and *production* arcs, which indicate the effect on the flux of matter through *PNs*, the SBGN Process Diagram specification also provides arcs for representing different possible modifications of a process, such as *modulation*, *stimulation*, *catalysis*, *inhibition* and *trigger* (or absolute activation). Finally, some arcs link nodes of the same type, such as the *equivalence* arc, linking *EPN* and *tag*, and the *logic arc*, linking two *logical operators* or an *EPN* to a *logical operator.*

Logical operators provide the means of indicating boolean combinations of influences from *EPNs* onto *PNs*. The three possibilities are conjunction (*and*), disjunction (*or*), and negation (*not*).

Compartments and ***submaps*** are containers that permit to gather together EPNs and PNs, either by spatial proximity or as map "modules". *Submaps* are "folded" in the main map, represented by a symbol. The unfolded *submaps* can be retrieved, for instance, in other windows of a software-based system, or on other pages if the diagrams are printed on paper.

Nature Biotechnology: doi:10.1038/nbt.1558

Rules for building an SBGN process diagram

The SBGN Process Diagram specification prescribes a number of rules in an effort to help eliminate ambiguity in an SBGN diagram. These rules must be complied with in order for the diagram to be a valid Level 1 SBGN Process Diagram. It is important to realize that SBGN does not dictate how to represent something, but rather how to interpret the representation. There is generally more than one way to represent a concept (for instance, hemoglobin can be represented as a *macromolecule*, a *complex* of four macromolecules, or a *multimer* with cardinality 4). However, everyone should interpret an SBGN Process Diagram the same way. Here we list some of the most important rules. Note that these are not the only rules defined by the SBGN Process Diagram Level 1 specification—users should consult the official specification documents at http://sbgn.org/ for a complete list of rules.

- An *entity pool node* belongs to only one *compartment*. If no compartment is drawn, it is assumed to belong to a "default" compartment.
- *Compartments* cannot be nested, and they represent disjoint spatial containers. *Compartments* may overlap visually, but such overlap does not imply any kind of physical containment; i.e., a *compartment* is never "part" of another.
- The layout or organization of a *compartment* does not imply anything about its topology.
- A *complex* may contain subunits that belong to different *compartments* (however, the complex itself belongs to only one).
- The layout or organization of the *EPNs* in a *complex* does not imply any information about topology.
- *Complexes* can be nested, making a given *complex*'s topology explicit.

Nature Biotechnology: doi:10.1038/nbt.1558

- A *complex* should consist of different *EPNs*. If two or more elements of the *complex* are identical then they should be replaced by a *multimer*.

- All substrates of a *Process node* should be different. If several copies of the same *EPN* are involved in the process, the cardinality label of the *consumption* arc should be used. The same rule applies to all products.

- Once the cardinality label is added to one arc, all other arcs connected to a *PN* must display a cardinality label.

- A *PN* should correspond to only one process or series of connected processes. If the same set of *EPNs* are consumed and produced by alternative processes, they should be connected by different *PNs*.

- A *PN* with no modulations has an underlying “basal rate” which describes the rate at which it converts inputs to outputs.

- When *modulations*, *stimulations* and *inhibitions* connect to the same *PN*, their effect on the basal rate of the process is combined. If their effects are independent of alternatives, different *PN* or *logical operators* must be used.

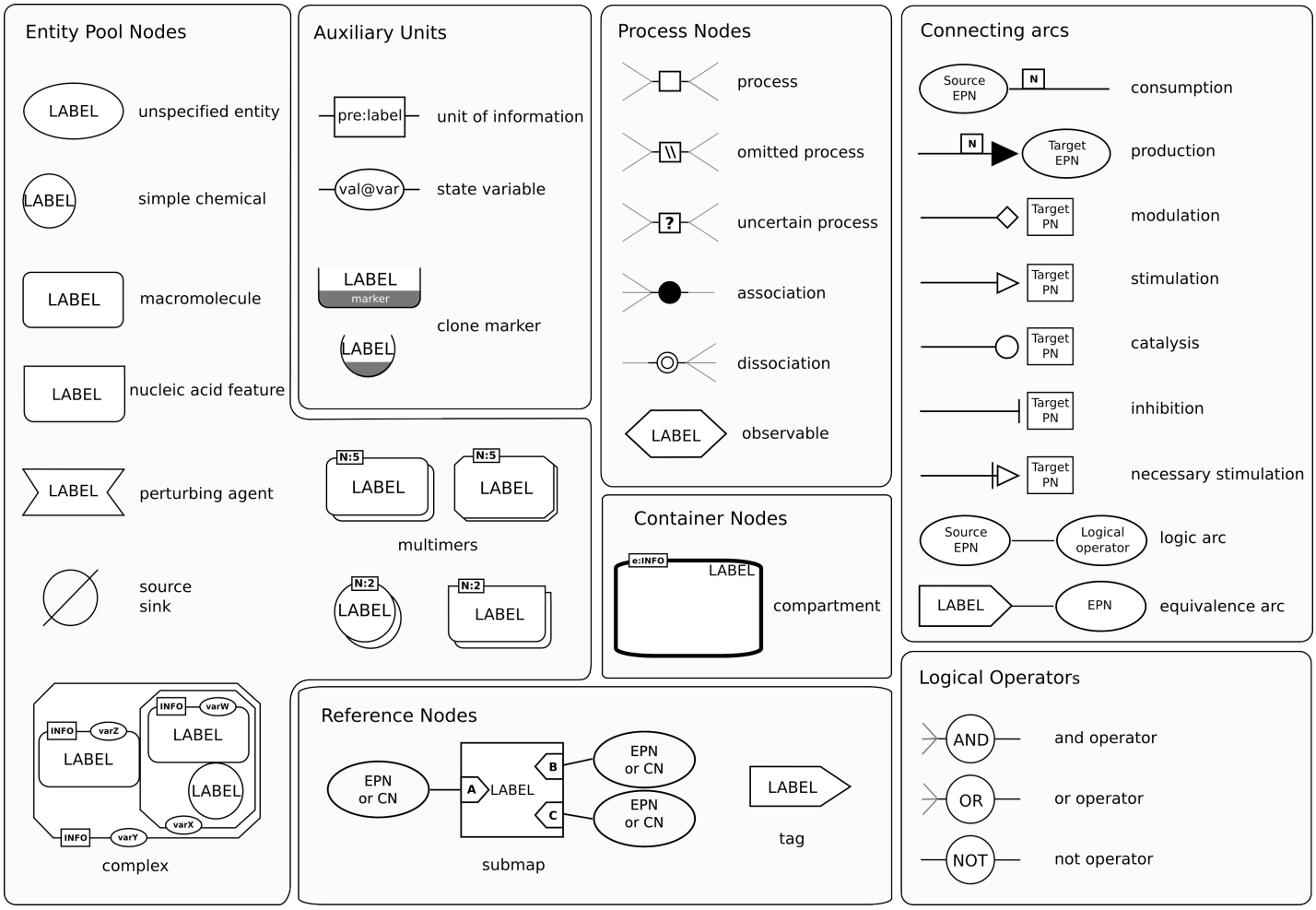
- Modulators that do not interact with each other in the manner above should be drawn as modulating different process nodes. Their effect is therefore additive.

- At most one *necessary activation* can be connected to a process node. If several *necessary activations* affect a process, their combination must be explicitly expressed using *logical operators*.

- At most one *catalysis* arc can connect to a process node. If several enzymes catalyze the “same” biochemical reaction, each catalytic process should be represented by a different *process node*.

Nature Biotechnology: doi:10.1038/nbt.1558

Figure S1: List of all glyphs specified by SBGN Process Diagram Level 1



Nature Biotechnology: doi:10.1038/nbt.1558

Supplementary Note 3:
Symbols Used in SBGN Entity Relationship Diagram Level 1

The SBGN Entity Relationship Diagram specification defines a set of symbols with precise semantics, together with syntactic rules that control their assembly. It also describes how such graphical information is to be interpreted. The essence of an entity relationship diagram is to depict the influences of entities upon the behavior of others. The entities are things that exist, either on their own or when statements become true. For instance, an entity can exist, different entities can interact, or a value can be assigned to an entity's property. “Influences” in this context therefore can be understood as logical consequences of this existence. Contrary to the Process Diagram notation, where the different processes affect each other, the relationships here are independent. One can imagine that each of the relationships represents a specific conclusion of a scientific experiment or article. Their addition to a drawing represents the knowledge gained about the effects of the entities upon each other. The independence of relationships is the key to avoiding the potential for combinatorial explosion inherent in the Process Diagram notation.

Entity nodes *(ENs)* represent elements of truth, i.e., things that exist. In ontology parlance, they are “continuants”. Entity nodes are the sources of influences. The SBGN Entity Relationship Diagram Level 1 specification provides three different types of *ENs*: the *interactors* (e*ntity* and *outcome*), the *logical operators* (*and*, *or*, *not* and *delay*) and the *perturbing agent.*

The semantics of SBGN entity relationship diagrams is carried by *Relationships*. Relationships are rules that decide the existence of entity nodes, based on the existence of others. In ontology parlance, they are “occurants”. Level 1 of the SBGN Entity Relationship notation provides two types of relationships, the *statements* and the *influences*.

Statements can be true or false. *Statements* are targets of *influences*. They are not true themselves, but can carry truth element (the *EN outcome*). SBGN Entity

Nature Biotechnology: doi:10.1038/nbt.1558

Relationships Level 1 provides four types of *statements*: *assignment*, *interaction*, *non-interaction* and *phenotype*.

Influences represent the effects of an entity on other relationships. The symbols attached to their extremities make precise their semantics. SBGN entity relationship *influences* can be viewed as logical rules linking *ENs* and other rules. The Entity Relationship Diagram Level 1 specification provides seven *influences*: *modulation*, *stimulation*, *inhibition*, *necessary stimulation*, *absolute inhibition*, *absolute stimulation*, and *logic arc*.

Nature Biotechnology: doi:10.1038/nbt.1558

Rules for building an SBGN entity relationship diagram

The SBGN Entity Relationship Diagram notation defines a number of rules to help eliminate ambiguity in a map. These rules must be followed in order for the diagram to be a valid SBGN Entity Relationship Diagram. Here we list some of the most important rules, but they are not the only ones defined by SBGN—users should consult the official specification at http://sbgn.org/ for a complete list of rules.

- Only one relationship can originate from an outcome, whether it is *influence* or *interaction*. The relationships are seen as independent rules; separate consequences of an assignment or an interaction have to originate from different outcomes, that is assertion of truth of this assignment or interaction.
- There cannot be both an *absolute stimulation* and an *absolute inhibition* targeting the same statement.
- In the case of a non-binary interaction, the “cis” or “trans” *unit of information* must be carried by the circle representing the n-ary *interaction*, and not the arc connecting this circle and a given interactor.
- If an *influence* targeting an *interaction* carries a “cis” or “trans” unit of information, at least one of the *interactors* must be the same *entity* as the origin of the *influence.*
- If more than one instance of an *entity* is involved in an *interaction* or a *non-interaction*, a *unit of information cardinality* must be associated with each entity involved in the statement.
- A *cis* or *trans unit of information* can be carried only by a relationship involving a single *entity*.

Nature Biotechnology: doi:10.1038/nbt.1558

Rules for understanding an SBGN entity relationship diagram

It is important to realize that the SBGN Entity Relationship Diagram specification does not dictate how to represent something, but rather how to interpret the representation. There is generally more than one way to represent a concept. However, everyone should interpret an SBGN entity relationship diagram the same way. SBGN entity relationships can be interpreted as logical rules describing the consequences of the existence of entities.

- An *interaction* linking the *interactors* A and B means: “A interacts with B”. An *outcome* on an *interaction* represents the cases when the statement is true, that is when the interaction effectively exists. If the interaction is a physical interaction between molecules, the *outcome* represents the complex resulting from the interaction. It is used as follows: “when (or if) A interacts with B then ...”
- An *assignment* linking a state variable value v to a *state-variable* V of an *entity* E means: “v is assigned to V of E” or “V of E takes the value v”. An *outcome* on an *assignment* represents the cases when the statement is true, that is when the variable effectively displays the value. It is used as follows: “when (or if) V of E takes the value v then ...”
- A *phenotype* P means: “P exists (can be observed)”.
- A *modulation* linking an *entity node* E and a *relationship* R means: “if E exists then R is either reinforced or weakened”.
- A *stimulation* linking an *entity node* E and a *relationship* R means: “if E exists then R is reinforced” or “if E then the probability of R is increased”.
- An *absolute stimulation* linking an *entity node* E and a *relationship* R means: “if E exists then R always takes place”.
- A *necessary stimulation* linking an *entity node* E and a *relationship* R means: “R only takes place if E exists”.
- An *inhibition* linking an *entity node* E and a *relationship* R means: “if E exists then R is weakened” or “if E then the probability of R is lowered”.
- An *absolute inhibition* linking an *entity node* E and a *relationship* R means: “if

Nature Biotechnology: doi:10.1038/nbt.1558

E exists then R never takes place".

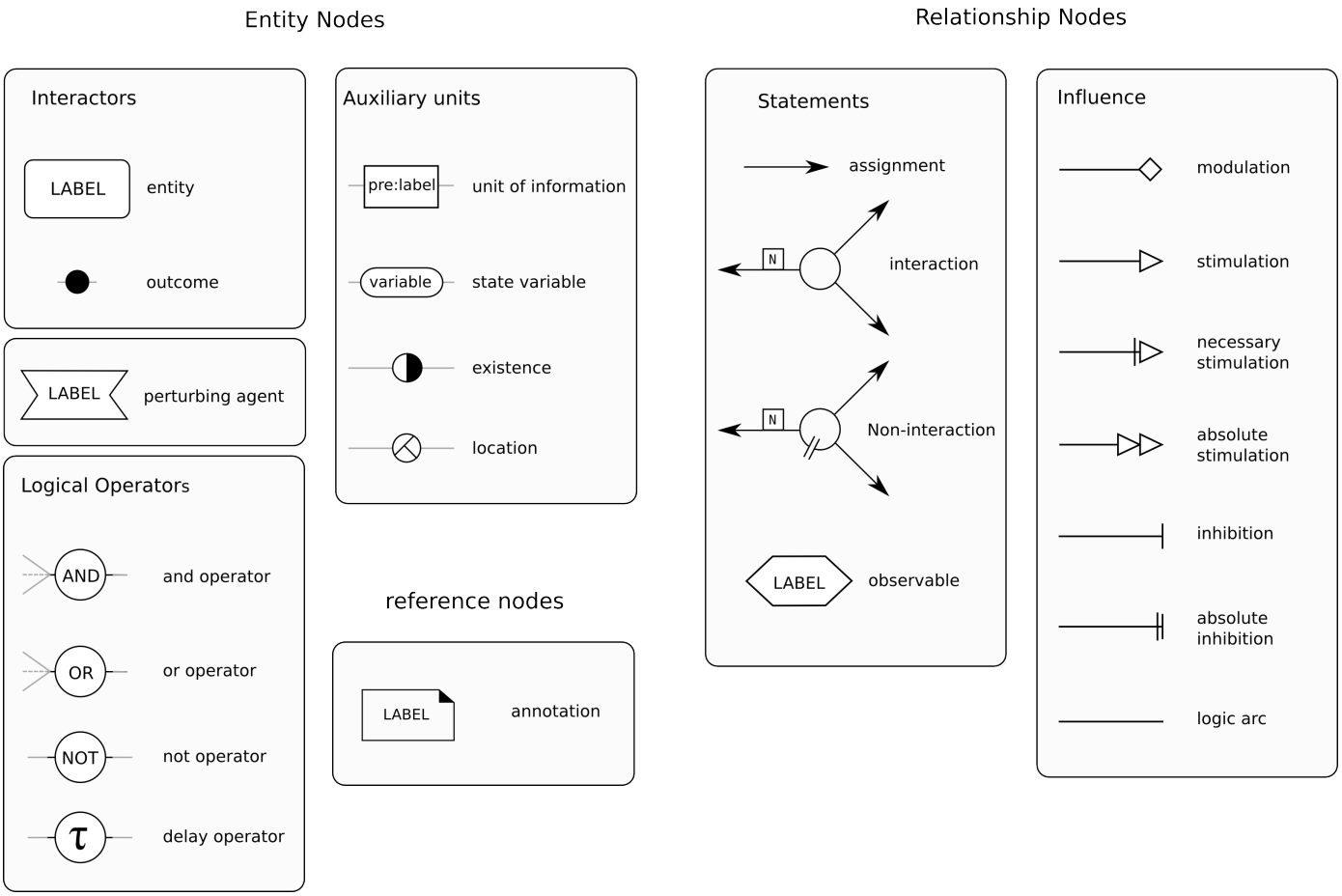
- An *and* linking several *logic arcs* originating from *entity nodes* E_i and an *influence* F means: "if for each i, E_i exists, then F".
- An *or* linking several *logic arcs* originating from *entity nodes* E_i and an *influence* F means: "if for any i, E_i exists, then F".
- A *not* linking a *logic arc* originating from an *entity node* E and an *influence* F means: "if E does not exist, then F".
- A *delay* linking a *logic arc* originating from an *entity node* E and an *influence* F means: "if E exists then F takes place, but not immediately".

The use of "cis" and "trans" units of information on a combination of relationships brings power and versatility to entity relationship diagrams in SBGN. However, the resulting semantics may be difficult to grasp. Here are the basic rules that permit understanding the diagrams:

- The *unit of information* "cis" or "trans" carried by an *interaction* refers to the *interactors* targeted by the *interaction*.
- The *unit of information* "cis" or "trans" carried by an *influence* targeting a state variable *assignment* refers to the origin of the *influence* and to the *entity* carrying the target of the *assignment*.
- The *unit of information* "cis" or "trans" carried by an *influence* targeting another *influence* refers to the origin of the carrying *influence* and to the origin of the targeted *influence*.
- The unit of information "cis" or "trans" carried by an influence targeting an *interaction* refers to the origin of the *influence* and all the relevant *interactors* targeted by the *interaction*.

Nature Biotechnology: doi:10.1038/nbt.1558

Figure S3: List of all glyphs specified by SBGN Entity Relationship Diagram Level 1



Nature Biotechnology: doi:10.1038/nbt.1558

Supplementary Note 4:
Symbols Used in SBGN Activity Flow Diagram Level 1

The SBGN Activity Flow Diagram specification defines a set of symbols and detailed syntactic rules for their use, to allow users to create pathway diagrams—for example, diagrams of signaling pathways—resembling those often found in research publications. An activity flow diagram is designed to show how activities are propagated from one entity to another. Glyphs (symbols) are the graphical units that represent concepts in SBGN. Each one is uniquely identified by a term from the Systems Biology Ontology (SBO). There are two types of glyphs in SBGN: nodes (which are subdivided into activity nodes, unit of information node, container nodes, and logical operator nodes), and arcs (edges) that characterize the relationships between nodes.

Activity nodes *(ANs)* represent activities produced by an entity from an entity pool. Level 1 of the SBGN Activity Flow Diagram specification defines three distinct glyphs, one for each of the following concepts: *biological activity, perturbation, and phenotype*. The notation uses one glyph to represent activities from all kinds of biological entities; collectively, they are called “biological activity”. The nature of the molecule that the activity comes from (e.g., simple chemical or macromolecule) can be encoded in the *units of information*. A biological activity can come from one biological entity, a part of an entity, or a combination of them; thus, biological activity is not equivalent to a biological entity per se. Each activity node can only be represented once within a given compartment.

Modulation arcs *(MAs)* describe the way in which one *AN* affects (or influences) others. Level 1 of the SBGN specification defines four *MAs*: *positive influence* (activation)*, negative influence* (inhibition)*, unknown influence,* and *necessary stimulation*.

Unit of information provides a way to illustrate the nature of the entity producing a given activity. In the SBGN activity flow notation, the *unit of information* is a

Nature Biotechnology: doi:10.1038/nbt.1558

unique type of node, in that its relationship to another node does not require an arc. Level 1 of the specification defines distinct glyphs for representing five different types of entities: *macromolecule, simple chemical, genetic, unspecified,* and *complex*.

Logical operators provide a means of indicating Boolean combinations of influences from *ANs* onto other *ANs*. The four possibilities are conjunction (*and*), disjunction (*or*), negation (*not*), and *delay*.

Compartments are containers that provide particular information about the location of ANs and MAs.

Rules for building an SBGN activity flow diagram

The SBGN Activity Flow notation defines a number of rules to help eliminate ambiguity in a diagram. These rules must be complied with in order for the diagram to be a valid Level 1 SBGN Activity Flow Diagram. It is important to realize that SBGN's activity flow notation does not impose how to represent something, but rather how to interpret the representation. There is generally more than one way to represent a concept; for instance, an EGF receptor can be represented by one *biological activity* node as EGF receptor activity, or by two *biological activity* nodes, one as EGF binding activity, the other as EGF receptor kinase activity. However, everyone should interpret an SBGN activity flow diagram the same way. Here we list some of the most important rules. Note that they are not the only rules defined by SBGN Activity Flow Level 1; users should consult the official specification at http://sbgn.org/ for a complete list of rules.

- An *activity node* can only appear once in a given *compartment*. If no compartment is drawn, the node is assumed to belong to a "default" compartment.

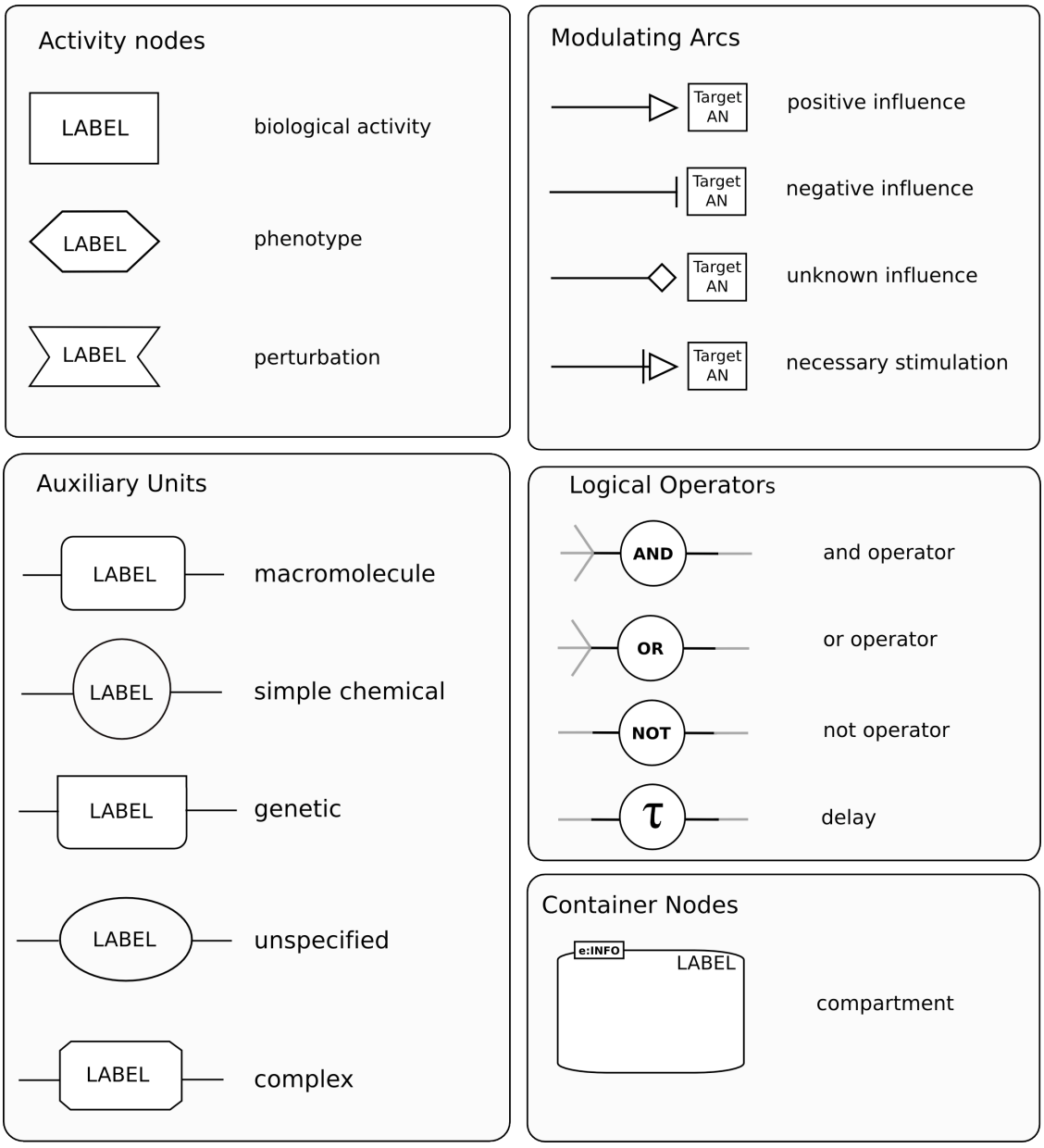
- *Compartments* cannot be nested, and they represent disjoint spatial containers. *Compartments* may overlap visually, but such overlap does not imply any kind of physical containment; i.e., a *compartment* is never "part" of another.

- The layout or organization of a *compartment* does not imply anything about its topology.

- The use of *unit of information* is not required, but should be used when it becomes necessary for proper interpretation of a diagram. In addition, *the unit of information* can be used without a label to simply indicate the nature of the entity producing the activity, or with a label to provide more information.

- *Perturbation* is used only as an origin to a modulation arc or a logic operator.

Figure S4: List of all glyphs specified by SBGN Activity Flow Diagram Level 1



Nature Biotechnology: doi:10.1038/nbt.1558

7 Conclusions et perspectives

Le développement d'un « écosystème » logiciel construit autour d'un ensemble de normes pour l'échange des modèles a eu un impact important sur la biologie des systèmes. Cet ensemble ne couvre pas encore la totalité du cycle de vie des modèles. De plus, il est pour l'instant très focalisé sur un type de modèle particulier. Cependant le travail continu, et des solutions sont en train d'émerger pour permettre la couverture nécessaire aux modèles de cellules, d'organes et d'organismes entiers.

7.1 Impact des normes présentées dans ce manuscrit

Les normes ayant été créées en collaborations avec les développeurs des outils principaux du domaines ainsi que les modélisateurs eux-mêmes, leur adoption a été rapide. Les citations des articles ne sont évidemment pas une preuve d'utilisation, tout au plus une indication d'intérêt (au pire une indication de notoriété). On peut cependant noter que les papiers SBML, MIRIAM et SBGN ont atteint 1829, 337 et 285 citations (Google Scholar 4 octobre 2013), des nombres assez respectables dans le domaine de la modélisation en biologie. Cette adoption a été assez rapide. En 2007, un sondage est paru sur l'utilisation des normes en biologie des systèmes (Klipp *et al.*, 2007). 80 % des réponses indiquaient que les normes étaient nécessaires, 60 % connaissaient SBML et 40 % MIRIAM. Si l'on excepte les logiciels généraux comme MatLab, Excel our R, les logiciels les plus utilisés étaient libSBML, CellDesigner et BioModels Database, reflétant l'importance prise par SBML.

Les pratiques des modélisateurs ont changés, avec l'aide des éditeurs de journaux scientifiques. Plus de 300 journaux recommandent la déposition des modèles en SBML et/ou dans BioModels Database. Cette recommandation est suivie de plus en plus fréquemment, résultant en un nombre croissant de modèles disponibles pour la communauté. Ces modèles sont de meilleure qualité. Lorsque BioModels Database a été créée, nous étions forcé à ré-écrire 100 % des modèles. Leur syntaxe était généralement fausse, et lorsque nous pouvions les utiliser, les résultats obtenus par simulation ne correspondaient pas aux publications. La situation à changé dramatiquement. Il est désormais très fréquent que les models soumis à BioModels Database soit non seulement corrects au niveau du format, mais fournissent également les bons résultats directement. Cerise sur le gâteau, les modèles sont également de plus en plus fréquemment annotés, en particulier avec des liens vers d'autres sources d'informations. L'amélioration de la

qualité des modèles mis à disposition de la communauté a entraîné leur réutilisation croissante, un des objectifs initiaux de tout nos efforts. Il y a désormais des « phylogénies » de modèles, avec plusieurs générations de modèles modifiés déposés dans les bases de données.

De nouveaux types d'activité de recherche ont également été rendus possibles, ou plus aisés, grâce à l'existence de ces normes. À partir de la description de réseaux biochimiques, on peut désormais produire automatiquement des modèles mathématiques (Büchel *et al.*, 2013). On peut importer des données en provenance de ressources variées pour construire des modèles cinétiques complets (Li *et al.*, 2010c). On peut également comparer et estimer de manière quantitative la « distance » entre deux modèles (Schulz *et al.*, 2011) et de ce fait rechercher les modèles pertinents. Une fois la collection de modèles achevée, il est possible de la joindre en un modèle global (Schulz *et al.*, 2006). Finalement, les formats standards et les identifiants partagés permettent d'intégrer modèles informatiques et données expérimentales

7.2 Que nous manque-t'il ?

Au cours des neuf années passées, nous avons complété SBML de façon à couvrir la description des modèles informatiques et leurs simulations dans le cadre de la biologie des systèmes (Figure 7.1). Il est clair que SED-ML ne couvre pour l'instant qu'une partie des types de simulations et analyses conduites dans le domaine. Cependant le langage se développe rapidement, et la balle est désormais dans le camp des utilisateurs et des éditeurs.

	Génération modèle	Description modèle	Simulations et analyses	Résultats numériques
Directives		MIRIAM	MIASE	
Formats		SBML, SBGN	SED ML	NuML
Terminologies		SBO	KISAO	TEDDY

FIGURE 7.1 – Matrice classant les normes en modélisation des systèmes biologiques.

Après que la première version de TEDDY ait été produite, il est devenu évident qu'une de ses utilisations pourrait être d'annoter des résultats de simulation. Le groupe de Pedro Mendes à Manchester développait alors un format, le *Systems Biology Result Markup Language* (SBRML, Dada *et al.*, 2010), et nous avons commencé à travailler sur un projet de banque de données de simulations, qui n'a finalement pas été financé. Quand les discussions sur SED-ML ont abordé la question des estimations de paramètres, qui nécessitent la comparaison de résultats de simulation avec des mesures expérimentales, SBRML est devenu un candidat naturel. SBRML inclut le modèle et les résultats, et est limité aux modèles encodés en SBML, ce qui empêchait son utilisation tel-quel (puisque SED-ML peut être utilisé avec n'importe quel format de modèle si il est basé sur XML). Nous avons donc extrait la partie de SBRML qui encodait les résultats, qui est devenu le *Numerical Markup Language* (NuML)[1]). Ce langage est proposé comme norme à utiliser en conjonction avec SBML ou SED-ML, dès que l'on a besoin de partager des tableaux de nombres. Il faut noter que d'autres communautés ont développé des formats similaires, par exemple SignalML[2] (Durka & Ircha, 2004) et BioSignalML (Brooks *et al.*, 2011). En revanche il n'y a pas à ma connaissance de directives décrivant les informations minimales à échanger avec des résultats numériques.

Une partie du cycle de vie des modèles n'est pas couverte du tout par les normes d'échange existantes ; il s'agit de la génération des modèles. Initialement les modèles utilisés en biologie des systèmes étaient principalement écrit *de novo*, à partir de connaissances extraites de la littérature scientifique par le modélisateur. Ce n'est plus toujours le cas. Nous avons déjà mentionné des modèles générés automatiquement à partir de cartes (Li *et al.*, 2010c; Büchel *et al.*, 2013) ou par fusion de modèles existants (Schulz *et al.*, 2006, 2011). Il est également possible de construire des modèles directement à partir de résultats expérimentaux (Axelsson *et al.*, 2011). Enfin, une importante part du travail de modélisation dans le développement de médicaments consiste à décider quel modèle correspond le mieux aux données obtenues chez les patients (Bonate, 2011). Il est donc important de pouvoir échanger des informations à propos de ce processus de construction, qui pourront être ensuite utilisées par exemple pour décider de la valeur d'un modèle dans un contexte donné. Ici tout est à faire, bien que des efforts existent pour certaines des approches, par exemple les procédures de *machine learning*, comme le *Predictive Model Markup Langage* (PMML)[3].

Notre travail au cours de la décennie passée s'est principalement concentré sur les modèles de type cinétique chimique, car ce sont les plus utilisés en biologie des systèmes (Hübner *et al.*, 2011). Les paquets de SBML niveau 3 permettent d'étendre le langage, et cela a été fait avec succès par exemple pour les modèles qualitatifs (Chaouiya *et al.*, 2014) ou bien les modèles de flux contraints[4]. Cependant, une telle extension d'un format fondamentalement basé sur la description de processus transformant le contenu de réservoirs (*pools*) en contenus d'autres

[1] `http://code.google.com/p/numl/`

[2] `http://bci.fuw.edu.pl/wiki/SignalML`

[3] `http://www.dmg.org/`

[4] `http://sbml.org/Documents/Specifications/SBML_Level_3/Packages/Flux_Balance_Constraints_(flux)`

réservoirs rencontre vite ses limites. Il est difficile d'imaginer que SBML puisse couvrir tous les types de modélisation en biologie. De plus, différentes communautés de modélisateurs ont commencé à développer leurs propres formats (Figure 7.2). En sus du format NeuroML (Goddard *et al.*, 2001; Gleeson *et al.*, 2010), développé par la communauté des modélisateurs en neuroscience, L'*International Neuroinformatics Coordinating Facility* (INCF) a soutenu le développement du format NineML, plus adapté aux réseaux de neurones. Les membres du project physiome ont développé depuis plusieurs années le langage CellML (Lloyd *et al.*, 2004), auquel s'est rajouté plus récemment FieldML (Christie *et al.*, 2009). Le consortium *Drug Disease Model Repository* (DDMoRe)[5], formé de groupes de recherche académiques et d'industries pharmaceutiques, développe une infrastructure pour la modélisation pharmacométrique, incluant le format PharmML.

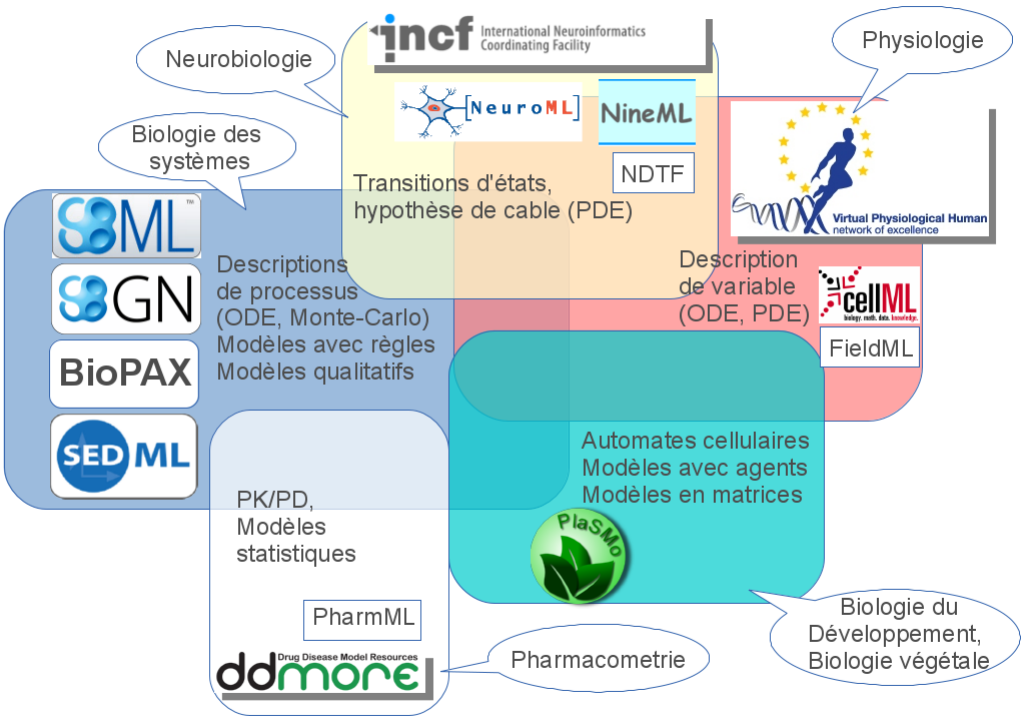


FIGURE 7.2 – Complémentarité et recouvrement des types de modélisation et des normes associées dans les différents domaines de modélisation du vivant. PlasMo est un projet financé par le BBSRC (http ://www.plasmo.ed.ac.uk/), focalisé sur les modèles de croissance végétale. les autres projets sont décris dans le texte.

Il est important de coordonner ces différents développements. En 2010, en collaboration

[5] `http://ddmore.eu`

avec Mike Hucka, représentant SBML, j'ai créé le *Computational Modeling in Biology Network* (COMBINE)[6], un forum qui tente d'amener tous les développeurs de normes pour la modélisation en biologie à la même table. Les activités de COMBINE sont variées, incluant l'organisation de réunions de travail, mais aussi la maintenance d'une infrastructure enregistrant et documentant les normes, ainsi que le développement de normes communes. SBML, SED-ML et SBGN ont été des normes COMBINE dès le départ. CellML a été acceptée plus récemment. Du fait de mon activité en neuroscience j'ai été en contact fréquent avec les développeurs de NeuroML et NineML. NeuroML est une norme candidate, qui remplit la plupart des critères[7] à l'exception d'une documentation technique détaillée et stable. FieldML et NineML (comme NuML) sont toujours plus à l'état de projets que de normes stables supportées par des communautés organisées. Jusqu'à mon départ de l'EBI en Octobre 2012 je coordonnais également le développement de PharmML. PharmML utilise SBML pour les « modèles structuraux » (la partie non statistique des modes pharmacométriques), et l'archive COMBINE (voir plus bas) comme conteneur. Il est donc probable que le format rejoindra COMBINE à terme. Cependant, du fait de son financement mixte publique/privé, il est probable que PharmML ne deviendra une norme associée qu'à la fin du projet DDMoRe.

Un des buts de COMBINE est d'améliorer l'intégration entre les normes membres. Comme nous l'avons vu plus haut, une expérience de modélisation nécessite non seulement la description du modèle (correspondant aux équations) — qui peut être décrit à l'aide de plusieurs fichiers —, mais aussi ce que l'on doit faire avec. Certains modèles complexes requièrent également des fichiers annexes comme des géométries, ou d'autres mesures expérimentales, données de patients etc. Un des premiers outils que nous avons développé est l'archive COMBINE [8]. Il s'agit d'un fichier ZIP[9] contenant toute l'information nécessaire pour reproduire une expérience de modélisation. Il contient les modèles nécessaires, la description des simulations, les fichiers de données, plus un fichier de metadonnées relatif à l'archive. L'espoir est que la possibilité de télécharger un fichier unique facilitera le stockage et les échanges de modèles. Un des chantiers à venir est d'améliorer les liens entre formats utilisés par l'archive. Par exemple il faut pouvoir identifier un élément d'un modèle encodé en SBML et sa représentation graphique dans un fichier SBGN-ML.

Un autre effort récemment lancé est la *MAthematical Modeling Ontology* (MAMO)[10]. Il existe de nombreuses ontologies utilisées dans des activités de modélisation. J'ai décris SBO plus haut. On peut également mentioner la *Computational Neuroscience Ontology* (CNO)[11] ainsi que la *Discrete-event Modeling Ontology* (DeMO)[12]. Ces trois efforts sont focalisés sur un type, ou un petit groupe de types, de modélisation donné. MAMO se veut plus générale et être

[6] `http://co.mbine.org`
[7] `http://co.mbine.org/Documents/criteria`
[8] `http://co.mbine.org/documents/archive`
[9] `https://en.wikipedia.org/wiki/Zip_(file_format)`
[10] `http://sourceforge.net/projects/mamo-ontology/`
[11] `http://www.incf.org/programs/modeling/cno`
[12] `http://www.cs.uga.edu/~jam/jsim/DeMO/`

une *upper ontology* pour la modélisation en biologie. Elle aidera à l'intégration sémantique des modèles utilisant différents formalismes.

7.3 Vision

Il y a un an paraissait dans le journal Cell le premier modèle d'une cellule complète (Karr *et al.*, 2012). Certes il manquait des petites parties, comme la physique des membranes, ainsi que l'aspect spatial des réactions biochimiques. Certes il s'agissait de la cellule la plus simple connue, *Mycoplasma genitalium*, ne comportant que 517 gènes. Mais ce fut néanmoins une rupture qualitative. Désormais, on peut modéliser des organismes complets, avec une granularité biochimique. Le modèle de *M genitalium* est un modèle composite, formé de plus de 20 modules dont la simulation est effectuée à l'aide de différentes approches (états stationnaires, équations différentielles et processus stochastiques). Si le groupe de Markus Covert a pu écrire le modèle à la main en MatLab, cette approche ne peut résister au changement d'échelle lorsque l'on modélisera des cellules eucaryotes. Le nombre de gène n'augmentera « que » d'un facteur 5 à 50. Mais il ne faut pas oublier les régulations épigénétiques, les épissages alternatifs, les ARN non codants, les régulations post-traductionnelles etc. De plus, si l'on peut espérer avoir un modèle unique de *M genitalium*, ce ne sera pas le cas pour une cellule humaine par exemple. Un modèle comprendra toute une constellation de modules, certains couvrant les mêmes processus cellulaires mais avec des granularités ou des approches différentes. Cette multiplication des modules sera encore plus flagrante quand l'on passera au niveau du tissu, comme c'est le cas au sein du projet *Virtual Liver Network* (Holzhütter *et al.*, 2012)[13]. La biologie rejoindra en quelque sorte la météorologie, où des ensembles de modèles de complexité variée couvrent des échelles de temps allant de l'heure au millénaire et d'espace allant de la centaine de mètres à la centaine de kilomètres.

De tels projets en biologie ne pourront voir le jour sans l'utilisation de normes informatiques complémentaires, compatibles, comportant une couche sémantique riche et normalisée elle aussi. Les modèles devront être ancrés dans les données expérimentales, de sorte que ces données deviendront partie intégrale des modèles. Ces normes devront être ouvertes. Plusieurs milliards d'euros ont été perdus dans les délais de construction de l'Airbus A380. Ces délais étaient dus à une incompatibilité entre les versions 4 et 5 du logiciel propriétaire CATIA de Dassault Système, utilisés par différentes composantes d'Airbus. Dans le cadre des sciences de la vie, où des milliers de laboratoires travaillent sur les mêmes modèles (au sens biologique du terme), à l'aide de centaines de logiciels, on peut se demander combien d'énergie, de temps et d'argent a été perdu à cause de problèmes similaires. Si on veut que la modélisation biologique réalise l'impact espéré sur la santé humaine, la nutrition et l'environnement, des formats propriétaires ne peuvent être la réponse. Non plus que des normes commerciales, développées par des organismes dont le but est de gagner de l'argent via la vente des spécifications et la

[13] `http://www.virtual-liver.de`

certification des outils et entreprises.

Open Access

Open Data

Open Source

Open Standards

Bibliographie

Axelsson E, Sandmann T, Horn T, Boutros M, Huber W, Fischer B (2011). Extracting quantitative genetic interaction phenotypes from matrix combinatorial rnai. *BMC bioinformatics* 12 : 342

Berners-Lee T, Fielding R, Masinter L (2005). Uniform resource identifier (URI) : Generic syntax. Available via the World Wide Web at `http://www.ietf.org/rfc/rfc3986.txt`

Bonate PL (2011). *Pharmacokinetic-Pharmacodynamic Modeling and Simulation*. Springer

Bornheimer SJ, Maurya MR, Farquhar MG, Subramaniam S (2004). Computational modeling reveals how interplay between components of a gtpase-cycle module regulates signal transduction. *Proc Natl Acad Sci USA* 101 : 15899–15904

Bornstein BJ, Keating SM, Jouraku A, Hucka M (2008). Libsbml : an api library for sbml. *Bioinformatics* 24 : 880–881

Bray T, Paoli J, Sperberg-McQueen CM, Maler E, Yergeau F (1997). Extensible markup language (xml). *World Wide Web Journal* 2 : 27–66

Brooks DJ, Hunter PJ, Smaill BH, Titchener MR (2011). Biosignalml–a meta-model for biosignals. *Conf Proc IEEE Eng Med Biol Soc* 2011 : 5670–5673

Büchel F, Rodriguez N, Swainston N, Wrzodek C, Czauderna T, Keller R, Mittag F, Schubert M, Glont M, Golebiewski M, van Iersel M, Keating S, Rall M, Wybrow M, Hermjakob H, Hucka M, Kell DB, Müller W, Mendes P, Zell A, Chaouiya C, Saez-Rodriguez J, Schreiber F, Laibe C, Dräger A, Le Novère N (2013). Large-scale generation of computational models from biochemical pathway map. *BMC Syst Biol* in revision. URL `http://arxiv.org/abs/1307.7005`

Chance B, Garfinkel D, Higgins J, B H (1969). Metabolic control mechanisms. 5. a solution for the equations representing interaction between glycolysis and respiration in ascites tumor cells. *J Biol Chem* 235 : 2426–2439

Chaouiya C, Berenguier D, Keating SM, Naldi A, van Iersel MP, Rodriguez N, Dräger A, Büchel F, Cokelaer T, Kowal B, Wicks B, Gonçalves E, Dorier J, Page M, Monteiro PT, von Kamp A, Xenarios I, de Jong H, Hucka M, Klamt S, Thieffry D, Le Novère N, Saez-Rodriguez J, Helikar T (2014). Sbml qualitative models : a model representation format and infrastructure to foster interactions between qualitative modelling formalisms and tools. *submitted* URL `http://arxiv.org/abs/1309.1910`

Christie GR, Nielsen PM, Blackett SA, Bradley CP, Hunter PJ (2009). Fieldml : concepts and implementation. *Philosophical Transactions of the Royal Society A : Mathematical, Physical and Engineering Sciences* 367 : 1869–1884

Courtot M, Juty N, Knüpfer C, Waltemath ZA D, Dräger A, Dumontier M, Finney A, Golebiewski M, Hastings J, Hoops S, Keating S, Kell DB, Kerrien S, Lawson J, Lister A, Lu J, Machne R, Mendes P, Pocock M, Rodriguez N, Villeger A, Wilkinson DJ, Wimalaratne S, Laibe C, Hucka M, Le Novère N (2011). Controlled vocabularies and semantics in systems biology. *Mol Syst Biol* 7 : 543

Dada JO, Spasić I, Paton NW, Mendes P (2010). Sbrml : a markup language for associating systems biology data with models. *Bioinformatics* 26 : 932–938

Danos V, Laneve C (2004). Formal molecular biology. *Theor Comput Sci* 325 : 69–110

Demir E, Cary MP, Paley S, Fukuda K, Lemer C, Vastrik I, Wu G, D'Eustachio P, Schaefer C, Luciano J, Schacherer F, Martinez-Flores I, Hu Z, Jimenez-Jacinto V, Joshi-Tope G, Kandasamy K, Lopez-Fuentes AC, Mi H, Pichler E, Rodchenkov I, Splendiani A, Tkachev S, Zucker J, Gopinath G, Rajasimha H, Ramakrishnan R, Shah I, Syed M, Anwar N, Babur O, Blinov M, Brauner E, Corwin D, Donaldson S, Gibbons F, Goldberg R, Hornbeck P, Luna A, Murray-Rust P, Neumann E, Ruebenacker O, Reubenacker O, Samwald M, van Iersel M, Wimalaratne S, Allen K, Braun B, Whirl-Carrillo M, Cheung KH, Dahlquist K, Finney A, Gillespie M, Glass E, Gong L, Haw R, Honig M, Hubaut O, Kane D, Krupa S, Kutmon M, Leonard J, Marks D, Merberg D, Petri V, Pico A, Ravenscroft D, Ren L, Shah N, Sunshine M, Tang R, Whaley R, Letovksy S, Buetow KH, Rzhetsky A, Schachter V, Sobral BS, Dogrusoz U, McWeeney S, Aladjem M, Birney E, Collado-Vides J, Goto S, Hucka M, Le Novère N, Maltsev N, Pandey A, Thomas P, Wingender E, Karp PD, Sander C, Bader GD (2010). The biopax community standard for pathway data sharing. *Nat Biotechnol* 28 : 935–42

Dräger A, Rodriguez N, Dumousseau M, Dörr A, Wrzodek C, Le Novère N, Zell A, Hucka M (2011). Jsbml : a flexible java library for working with sbml. *Bioinformatics* 27 : 2167–2168

Durka PJ, Ircha D (2004). Signalml : metaformat for description of biomedical time series. *Bioinformatics* 76 : 253–259

Edelstein SJ, Schaad O, Henry E, Bertrand D, Changeux JP (1996). A kinetic mechanism for nicotinic acetylcholine receptors based on multiple allosteric transitions. *Biol Cybern* 75 : 361–379

Funahashi A, Matsuoka Y, Jouraku A, Morohashi M, Kikuchi N, Kitano H (2008). Celldesigner 3.5 : a versatile modeling tool for biochemical networks. *Proceedings of the IEEE* 96 : 1254–1265

Gleeson P, Crook S, Cannon RC, Hines ML, Billings GO, Farinella M, Morse TM, Davison AP, Ray S, Bhalla US, Barnes SR, Dimitrova YD, Silver R (2010). Neuroml : a language for describing data driven models of neurons and networks with a high degree of biological detail. *PLoS computational biology* 6 : e1000815

Goddard NH, Hucka M, Howell F, Cornelis H, Shankar K, Beeman D (2001). Towards neuroml : model description methods for collaborative modelling in neuroscience. *Philosophical Transactions of the Royal Society of London Series B : Biological Sciences* 356 : 1209–1228

Gonzalez AG, Naldi A, Sanchez L, Thieffry D, Chaouiya C (2006). Ginsim : a software suite for the qualitative modelling, simulation and analysis of regulatory networks. *Biosystems* 84 : 91–100

Holzhütter HG, Drasdo D, Preusser T, Lippert J, Henney AM (2012). The virtual liver : a multidisciplinary, multilevel challenge for systems biology. *Wiley Interdisciplinary Reviews : Systems Biology and Medicine* 4 : 221–235

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

Huang CY, Ferrell JEJ (1996). Ultrasensitivity in the mitogen-activated protein kinase cascade. *Proc Natl Acad Sci USA* 93 : 10078–10083

Hübner K, Sahle S, Kummer U (2011). Applications and trends in systems biology in biochemistry. *FEBS Journal* 278 : 2767–2857

Hucka M (2004). Ruminations on creating a biomodels.net. Technical report, California Institute of Technology. URL `http://lenoverelab.org/documents/biomodels-14May2004-Hucka.pdf`

Hucka M, Finney A, Sauro HM, Bolouri H, Doyle JC, Kitano H, Arkin AP, Bornstein BJ, Bray D, Cornish-Bowden A, Cuellar AA, Dronov S, Gilles ED, Ginkel M, Gor V, Goryanin II, Hedley WJ, Hodgman TC, Hofmeyr JH, Hunter PJ, Juty NS, Kasberger JL, Kremling A,

Kummer U, Le Novère N, Loew LM, Lucio D, Mendes P, Minch E, Mjolsness ED, Nakayama Y, Nelson MR, Nielsen PF, Sakurada T, Schaff JC, Shapiro BE, Shimizu TS, Spence HD, Stelling J, Takahashi K, Tomita M, Wagner J, Wang J (2003). The systems biology markup language (sbml) : a medium for representation and exchange of biochemical network models. *Bioinformatics* 19 : 524–531

Ideker T, Galitski T, Hood L (2001). A new approach to decoding life : systems biology. *Annual review of genomics and human genetics* 2 : 343–372

Juty N, Le Novère N, Laibe C (2012). Identifiers.org and miriam registry : community resources to provide persistent identification. *Nucleic Acids Res* 40 : D580–D586

Karr JR, Sanghvi JC, Macklin DN, Gutschow MV, Jacobs JM, Bolival B, Assad-Garcia N, Glass JI, Covert MW (2012). A whole-cell computational model predicts phenotype from genotype. *Cell* 150 : 389–401

Kauffman SA (1969). Metabolic stability and epigenesis in randomly constructed genetic nets. *J Theor Biol* 22 : 437–467

Keating SM, Bornstein BJ, Finney A, Hucka M (2006). Sbmltoolbox : an sbml toolbox for matlab users. *Bioinformatics* 22 : 1275–1277

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

Kitano H (2002). Systems biology : a brief overview. *Science* 295 : 1662–1664

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

Knüpfer C, Beckstein C, Dittrich P, Novère NL (2013). Structure, function, and behaviour of computational models in systems biology. *BMC Syst Biol* 7 : 43

Köhn D, Le Novère N (2008). Sed-ml - an xml format for the implementation of the miase guidelines. In Heiner M, Uhrmacher A (eds.) *Proceedings of the 6th conference on Computational Methods in Systems Biology*, volume 5307 of *Lecture Notes in Bioinformatics*, pp. 176–190

Laibe C, Le Novère N (2007). Miriam resources : tools to generate and resolve robust cross-references in systems biology. *BMC Syst Biol* 1 : 58

Le Novère N, Bornstein B, Broicher A, Courtot M, Donizelli M, Dharuri H, Li L, Sauro H, Schilstra M, Shapiro B, Snoep JL, Hucka M (2006). Biomodels database : a free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. *Nucleic Acids Res* 34 : D689–D91

Le Novère N, Hucka M, Mi H, Moodie S, Schreiber F, Sorokin A, Demir E, Wegner K, Aladjem MI, Wimalaratne SM, Bergman FT, Gauges R, Ghazal P, Kawaji H, Li L, Matsuoka Y, Villeger A, Boyd SE, Calzone L, Courtot M, Dogrusoz U, Freeman TC, Funahashi A, Ghosh S, Jouraku A, Kim S, Kolpakov F, Luna A, Sahle S, Schmidt E, Watterson S, Wu G, Goryanin I, Kell DB, Sander C, Sauro H, Snoep JL, Kohn K, Kitano H (2009). The systems biology graphical notation. *Nat Biotechnol* 27 : 735–41

Le Novère N, Shimizu TS (2001). Stochsim : modelling of stochastic biomolecular processes. *Bioinformatics* 17 : 575–576

Le Novère N, Finney A (2005). A simple scheme for annotating sbml with references to controlled vocabularies and database entries. Technical report, European Bioinformatics Institute. URL `http://lenoverelab.org/documents/MIRIAM-annotations_08December2005-Le_Novere-Finney.pdf`

Li C, Courtot M, Laibe C, Le Novère N (2010a). Biomodels.net web services, a free and integrated toolkit for computational modelling software. *Brief Bioinfo* 11 : 270–277

Li C, Donizelli M, Rodriguez N, Dharuri H, Endler L, Chelliah V, Li L, He E, Henry A, Stefan MI, Snoep JL, Hucka M, Le Novère N, Laibe C (2010b). Biomodels database : An enhanced, curated and annotated resource for published quantitative kinetic models. *BMC Syst Biol* 4 : 92

Li P, Dada J, Jameson D, Spasic I, Swainston N, Carroll K, Dunn W, Khan F, Malys N, Messiha H, Simeonidis E, Weichart D, Winder C, Wishart J, Broomhead DS, Goble CA, Gaskell SJ, Kell DB, Westerhoff HV, Mendes P, Paton N (2010c). Systematic integration of experimental data and models in systems biology. *BMC bioinfo* 11 : 582

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

Meyer T, Stryer L (1988). Molecular model for receptor-stimulated calcium spikin. *Proc Natl Acad Sci USA* 85 : 5051–5055

Munn K, Smith B (eds.) (2008). *Applied Ontology. An introduction*. Heusenstamm : Ontos Verlag

Schulz M, Krause F, Le Novere N, Klipp E, Liebermeister W (2011). Retrieval, alignment, and clustering of computational models based on semantic annotations. *Mol Syst Biol* 7

Schulz M, Uhlendorf J, Klipp E, Liebermeister W (2006). Sbmlmerge, a system for combining biochemical network models. *Genome Info Series* 17 : 62

Shapiro BE, Hucka M, Finney A, Doyle J (2004). Mathsbml : a package for manipulating sbml-based biological models. *Bioinformatics* 20 : 2829–2831

Turing AM (1952). The chemical basis of morphogenesis. *Phil Trans Royal Soc, Series B, Bil Sci* 237 : 37–72

Ueda HR, Hagiwara M, Kitano H (2001). Robust oscillations within the interlocked feedback model of drosophila circadian rhythm. *J Theor Biol* 210 : 401–406

van Iersel MP, Villéger AC, Czauderna T, Boyd SE, Bergmann FT, Luna A, Demir E, Sorokin A, Dogrusoz U, Matsuoka Y, Funahashi A, Aladjem MI, Mi H, Moodie SL, Kitano H, Le Novère N, Schreiber F (2012). Software support for sbgn maps : Sbgn-ml and libsbgn. *Bioinformatics* 28 : 2016–2021

von Bertalanffy L (1928). *Modern Theories of Development : An Introduction to Theoretical Biology (Kritische Theorie der Formbildung)*. Oxford University Press

Waltemath D, Adams R, Beard DA, Bergmann FT, Bhalla US, Britten R, Chelliah V, Cooling MT, Cooper J, Crampin EJ, Garny A, Hoops S, Hucka M, Hunter P, Klipp E, Laibe C, Miller AK, Moraru I, Nickerson D, Nielsen P, Nikolski M, Sahle S, Sauro HM, Schmidt H, Snoep JL, Tolle D, Wolkenhauer O, Le Novère N (2011a). Minimum information about a simulation experiment (miase). *PLoS Comput Biol* 7 : e1001122

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

Zhukova A, Adams R, Laibe C, Le Novère N (2012). Libkisao : a java library for querying kisao. *BMC Notes* 5 : 220

A

Curriculum Vitæ

Né le 12 March 1969 à Strasbourg, France
✉ : Babraham Institute, Cambridge, CB22 3AT, Royaume-Uni
☎ : +44-1223-496-433
@ : n.lenovere@gmail.com
WWW : http://lenoverelab.org/perso/lenov

A.1 Qualifications et titres

2010 Directeur de recherche CNRS ;

2001 Chargé de recherche CNRS ;

1998 Doctorat de pharmacologie moléculaire et cellulaire *Université Paris VI* ;

1993 Magistere de Biology-Biochimie *École Normale Supérieure, Paris* ;

1993 DEA de pharmacologie moléculaire et cellulaire *Université Paris VI* ;

1991 Maitrise de biologie cellulaire *Université Paris VI* ;

1991 Licence de biologie cellulaire et physiologie *Université Paris VI* ;

1988 Baccalaureat scientifique *Prytané Militaire, La Flèche*.

A.2 Expérience de recherche

depuis 2012 Chef de groupe senior, Babraham Institute, Cambridge, Royaume-Uni ;

2003-2012 Chef de groupe, EMBL-EBI, Wellcome-Trust Genome Campus, Cambridge, Royaume-Uni ;

2001-2003 Chargé de recherche, équipe de Jean-Pierre Changeux, Institut Pasteur, Paris, France ;

1999-2001 Chercheur post-doctoral, équipe de Dennis Bray, Université de Cambridge, Royaume-Uni ;

1992-1999 Thésard et chercheur post-doctoral, équipe de Jean-Pierre Changeux, Institut Pasteur, Paris ;

1991 Stage de maitrise, équipe de Michel Fardeau, INSERM, Paris.

A.3 Honneurs

2003 Prix JM Le Goff (Académie des Sciences)

1999-2001 Financement post-doctoral EMBO

1998-1999 Allocation *Roux* de l'Institut Pasteur

1994-1997 Allocation de thèse du ministère de la recherche

A.4 Financements sur dossiers[1]

2013-2014 BBSRC Linking data with Identifiers.org - GBP 120K

2013-2016 EU Commission AgedBrainSYSBIO Large-scale collaborative project - Eur 470K

2012-2015 BBSRC BioModels Database - GBP 534K

2012-2014 EU commission Open PHACTS – Eur 64K

2011-2016 EU commission Drug Diseases Model Resources - Eur 1800K

2010-2014 BBSRC How do cells shape and interpret PIP3 signals ? - GBP 380K

2010-2012 NIH Continued support and development of SBML. USD 335K

2010-2014 EU commission SynSys Large-scale collaborative project - Eur 850K

2010 BBSRC First COMBINE meeting : Bridging model structure, semantics and representation - GBP 10.1K

2008-2009 EU commission ELIXIR technical feasibility study - GBP 100K

2008-2011 BBSRC BioModels Database : a unified resource for Systems Biology models - GBP 530K

2007-2010 NIH Continued support and development of SBML - USD 265K

2007-2011 BBSRC Interfacing standards and ontologies in Systems Biology - GBP 45K

2006-2007 BBSRC Software infrastructure to support the standard of model curation and annotation MIRIAM - GBP 80K

[1] Les quantités représentent les parts de mon groupe, non les financements complets

2005-2008 NEDO Development of The Systems Biology Graphical Notation, a standard graphical notation for diagrams of computational models in biology - JPY 14M

2005-2007 NIH Continued support and development of SBML - USD 173K

2004-1009 EU commission The European Membrane Protein Consortium - EUR 376K

2003-2005 ESF The Meso-telencephalic Dopamine Consortium (DopaNet) - EUR 80K

B

Animation de la recherche

B.1 Encadrement d'étudiant

Encadrement doctoral Benedetta Frida Baldi (2009-present), Christine Hoyer (2009-present), Michele Mattioni (2008-2012), Lu Li (2007-2010), Melanie Stefan (2006-2009), Dominic Tolle (2005-2009), Zhi-Yan Han (1999-2002)

Visiteurs doctoraux Youwei Zhang (2011), Nathan Skeene (2010), Ron Henkel (2009), Koray Dagan Kaya (2008), Christian Knüpfer (2007), Dagmar Köhn (2007)

Stages John Gowers (2013, co-supervision Nicolas Rodriguez), Grigalius Taukanska (2013), Marija Jankovic (2013, co-supervision Vladimir Kiselev), Yangyang Zhao (2012, co-supervision Camille Laibe), Ishan Ajmera (2011, co-supervision Vijilaskhimi Chelliah), Michael Schubert (2011, supervision Vijilaskhimi Chelliah), Anna Zhukova (2011), Gael Jalowicki (2011, supervision by Nicolas Rodriguez), Jean-Baptiste Petit (2010, co-supervision Nicolas Rodriguez), Karim Tazibt (2010, supervision Camille Laibe), Marine Dumousseau (2009), Duncan Berenguier (2008), Ranjita Dutta-Roy (2008), Kedar Nath Natarajan (2008), Antonia Mayer (2007), Anika Oellrich (2007), Enuo He (2006), Arnaud Henry (2006), Camille Laibe (2006), Renaud Schiappa (2006), Lu Li (2005), Alexander Broicher (2004), Marie-Ange Djite (2004).

Jurys de thèse Lukas Endler (2012), Allyson Lister (2011), Andrew Miller (2011), Dagmar Waltemath (2011), Aurelien Rizk (2011), Carito Guloziowski (2010), Sarala Wimalaratne (2009), Benjamin Hall (2007), Pim Van Nierop (2007), Stéphanie Weiss (2006), Anne-Sophie Villegier (2005), Geoffroy Golfier (2004)

B.2 Enseignement

Organisateur Cours EBI *In silico systems biology* (2010, 2011, 2012, 2013) ; E-MeP *Advanced Training Workshop in Bioinformatics of Membrane Proteins* (2008) ; Okinawa *Computational Neuroscience Course* (2006) ;

Intervenant Cours doctoral de l'Instituto Gulbenkian de Ciência (2012) ; EBI-Wellcome Trust *Summer School in Bioinformatics* (2010, 2011, 2012) ; SystemsX.ch/SIB *Summer School* (2011) ; CRG *Summer course on systems biology* (2009) ; formation EBI *Networks and Pathways* (2009) ;

Équipe iGEM de Cambridge (2007, 2009) ; APO-SYS *data management workshop* (2008) ; SysBioMed *Winter School* (2007) ; EMBL *international PhD program* (2005, 2007) ; Symbionic course *Methods, data handling and standards in neuronal systems biology* (2005) ; FEBS *1st Advanced Course on Systems Biology* (2005) ; ESF course *Modelling Metabolic and Signal Transduction Networks* (2005) ; DEA *Neurobiologie et pharmacologies* (École Normale Supérieure) (2002).

B.3 Évaluation scientifique

Comités de lecture Bioinformatics, BMC Bioinformatics, BMC Genomics, BMC Neuroscience, BMC Systems Biology, British Journal of Pharmacology, Databases, EMBO Journal, European Journal of Neuroscience, FEBS Journal, FEBS Letters, Gene, Genome Biology, IEEE/ACM Transactions on Computational Biology and Bioinformatics, IET Systems Biology, IUBMB Life, Journal of Computational Neuroscience, Journal of Neurobiology, Journal of Neuroscience, Journal of Neurochemistry, Journal of Theoretical Biology, Medicinal Chemistry, Molecular Biology and Evolution, Molecular Biosystems, Molecular Systems Biology, Nature Reviews Molecular Cell Biology, Neural Networks, Physics Letters A, PLoS Computational Biology, Proceedings of the National Academy of Science USA, Receptors and Channels, Royal Society Interface focus, Wiley Interdisciplinary Reviews : Systems Biology and Medicine

Comités éditoriaux PeerJ (2012-), BMC Systems Biology (2006-, associate editor 2008-)

Comités de financement FRM Bioinformatics (2013), Forschungszentrum Juelich GmbH (2013), BMBF eBio (2011), BBSRC TRDF (2010) and STRDF (2011), ANR/BBSRC Systems Biology (2007), ANR SYSCOMM (2008, 2009), BBSRC expert pool (2009)

Comités scientifique de conférences ICSB (2006, 2008, 2010, 2014),BioSysBio (2008, 2009), CIBB 2009, CMSB (2005, 2008), ECCB 2008, ICBO (2012, 2013), SBML forums and hackathons, SBGN forums and hackathons.

Comités de standardisation IUPHAR nomenclature sub-committee for nicotinic receptors (co-chair), Systems Biology Markup Language (2006-2008, 2011-), Systems Biology Graphical Notation (2008-2012)

Sociétés scientifiques the International Society for Systems Biology (executive board 2007-), International Society for Computational Biology

Comités de pilotage ISBE (2012-), AgedBrainSYSBIO (2013-), The European Membrane Protein Consortium (E-MeP, 2004-2007)

Conseils scientifiques BioPAX (NIH), ENFIN (EU), CARMEN (EPSRC), Plant Models Portal (BBSRC), Centre de recherche de l'Institut Curie, Virtual Liver Network (BMBF), Pole biosanté Rabelais, Labex EpiGenMed

Comités, d'experts AERES (2 unités), EU Systems Biology for Medical Applications (2007), INSERM Tabac : comprendre la dépendance pour agir (2003).

B.4 Présentations

B.4.1 Invitations à présenter dans des conférences internationales

35. 15 November 2012 - WT/EBI Open Source Software for Systems, Pathways, Interactions and Networks, Hinxton, UK. "BioModels Database - Sharing and re-using computational models of biological processes"

34. 11 September 2012 - INCF Neuroinformatics conference 2012, Munich, DE. "Large-scale generation of mathematical models from biological pathways".

33. 12 March 2012 - Convergence in Computational Neuroscience 2012, Edinburgh, UK. "The COMBINE Initiative".

32. 30 January 2012 - SIB days, Biel, CH. "MIRIAM Registry and Identifiers.org. Robust and versatile identification in life sciences".

31. 05 May 2011 - Data Management for systems biology and the life sciences, Heidelberg, DE. "Models For All - Standards for Describing the Whole Life-Cycle of Modeling in Life Sciences".

30. 12 October 2010 - Eleventh International Conference on Systems Biology, Edinburgh, UK. "Modelling the Response of Allosteric Calcium Sensors Involved in Synaptic Plasticity".

29. 30 August 2010 - third INCF Neuroinformatics congress, Kobe, JP. " Describing the whole life-cycle of modelling in neuroscience".

28. 19 May 2010 - Sixth Annual Symposium of the Cambridge Computational Biology Institute, Cambridge, UK. "Ligand depletion in vivo modulates the dynamic range and cooperativity of signal transduction".

27. 22 March 2010 - International Symposium on Integrative Bioinformatics 2010, Cambridge, UK. "Knowledge representation and ontologies in Systems Biology".

26. 03 March 2010 - Therapeutic Applications of Computational Biology and Chemistry (TACBAC) 2010, Hinxton, UK. "Knowledge representation in systems biology"

25. 11 February 2010 - WT/CSHL omputational Cell Biology 2010, Hinxton, UK. "Toward a consistent set of interoperable standards to represent models and simulations".

24. 22 July 2009 - Data integration in the life sciences (DILS2009), Manchester, UK. “Data Integration and Semantic Enrichment of Systems Biology Models”.

23. 30 June 2009 - 17th Annual International Conference on Intelligent Systems for Molecular Biology (ISMB2009), Stockholm, SE. “BioModels Database, a database of curated and annotated quantitative models with Web Services and analysis tools”.

22. 27 June 2009 - 10th BioPathwways meeting, Stockholm, SE. “The Systems Biology Graphical Notation”.

21. 23 October 2008 - 10th International EMBL PhD Student Symposium, Heidelberg, DE. “Remember or forget : allosteric changeover switches and molecular memory”.

20. 13 October 2008 - 6th Conference on Computational Methods in Systems Biology (CMSB), Warnemunde, DE. “Multiscale modeling of synaptic signalling”.

19. 04 October 2008 - 5th international meeting on computational intelligence for bioinformatics and biostatistics (CIBB), Vietri Sul Mare, IT. “The Systems Biology Graphical Notation”.

18. 24 August 2008 - 9th International Conference on Systems Biology (ICSB), Goteborg, SE. “The Minimum Information About a Simulation Experiment”.

17. 18 March 2008 - Genomes to Systems 2008, Manchester, UK. “Principled annotation of quantitative models in Systems Biology”.

16. 13 October 2007 - WT/CSHL meeting on *Functional Genomics and Systems Biology*, Hinxton, UK. “Computational Models of Synaptic Plasticity”.

15. 05 October 2007 - Satellite symposium of the 8th International Conference on Systems Biology (ICSB), *Frontiers in Application of Systems Modeling and Simulation*, Long Beach, USA. “Linking biophysics and chemical kinetics to improve realism of models : calmodulin and synaptic plasticity”.

14. 23 July 2007 - 15th Annual International Conference on Intelligent Systems for Molecular Biology (ISMB), Vienna, AU. “BioModels Database, a curated resource of annotated published models”.

13. 12 July 2007 - Journées Ouvertes Biologie, Informatique et Mathématiques (JOBIM) 2007, Marseille, FR. “Kinetic models [in [Systems] Neurobiology] What, why and how”.

12. 12 January 2007 - BioSysBio 2007, Manchester, UK. “BioModels.net : Semantics for model sharing”.

11. 11 October 2006 - 7th International Conference on Systems Biology (ICSB), Yokohama, JP. “MIRIAM and Biomodels DB : Curation and Exchange of Quantitative Models”.

10. 16 June 2006 - EU workshop *Databasing the brain*, Oslo, NO. “BioModels Database, curation and exchange of quantitative models”.

9. 21 March 2006 - 2nd International Symposium on Experimental Standard Conditions of Enzyme Characterizations, Ruedesheim/Rhein, DE. “Adding semantics to kinetic models of biochemical pathways”.

8. 18 November 2005 - 1st International BioPAX Symposium *Biological pathways : communication and analysis*, Tokyo, JP. “Curation and exchange of kinetic models of biochemical pathways”.

7. 12 September 2005 - 4th Workshop on Computation of Biochemical Pathways and Genetic Networks, Heidelberg, DE. “Particle-based Stochastic Simulations”.

6. 13 September 2005 - 4th Workshop on Computation of Biochemical Pathways and Genetic Networks, Heidelberg, DE. “BioModels.net, Tools and Resources to Support Computational Systems Biology”.

5. 17 March 2005 - 1st FEBS Advanced Course on Systems Biology *From Molecules and Modeling To Cells*, Gosau, AT. “Computational Systems Neurobiology”.

4. 02 November 2002 - Satellite symposium of the 32nd annual meeting of the Society for Neuroscience *Neuroscience Database : Web Capabilities for. Data Sharing*, Orlando, USA. “Toward a systems biology of the neuron - The DopaNet consortium”.

3. 10 April 2002 - 6ème congrès annuel de la Société Française de Pharmacologie, Rennes. “Nicotinic receptors : From structures to behaviours”.

2. 14 June 2001 - XXIème Séminaire de la Société Française de Biologie Théorique. *Les Niveaux d’Organisation en Biologie : Enjeux et Perspectives*, Paris. “Modélisation de l’appareil chimiotactique bactérien”.

1. 12 March 2001 - Colloque INSERM *Modélisation mathématique et simulation : une démarche innovante en recherche thérapeutique*, Paris. “Stochastic simulation of biochemical pathways”.

B.4.2 Autres présentations

2013 - Babraham Institute Lab talks, Babraham, UK. “Large-scale harnessing of pathway knowledge to provide pump-priming models for all”

2012 - NCBO webinars, Stanford, USA (invited by Trish Whetzel). “MIRIAM Registry and Identifiers.org. Robust and versatile identification in life sciences”.

2012 - Imperial College, London, UK (invited by Michael Sternberg). “This or that ? Allosteric Calcium Sensors and Synaptic Plasticity.”

2012 - Instituto Gulbenkian de Ciência, Lisbon, PT (invited by Claudine Chaouya). “Decoding calcium signals involved in synaptic plasticity”.

2011 - EMBL-EBI Industry Program, Hinxton, UK. “What is Systems Biology ? What are the challenges ahead ?”

2011 - EMBL-EBI, Hinxton, UK - “Models For All - Standards for Describing the Whole Life-Cycle of Modeling in Life Sciences”.

2011 - EMBL-Hamburg, Hamburg, DE (invited by Matthias Wilmanns) - “Decoding calcium signals involved in synaptic plasticity”.

2010 - Institut Génétique Biologie Moléculaire Cellulaire, Ilkirch, FR (invited by Olivier Pourquie) - “Sharing enriched computational models, a cornerstone for Integrative Biology”.

2010 - Merk/Serono, Geneva, CH - “Sharing enriched computational models, a cornerstone for Integrative Biology”.

2010 - SBML tenth anniversary, Edinburgh, UK (Invited by Michael Hucka) - “Sharing enriched computational models, a cornerstone for Integrative Biology (Desperate attempt to justify my existence)”.

2010 - INRIA, Bordeaux, FR (Invited by Claude Kirchner) - “Sharing enriched computational models, a cornerstone for Integrative Biology”.

2010 - Institut Pasteur, Paris, FR (invited by Tony Pugsley) - “Sharing enriched computational models, a cornerstone for Integrative Biology”.

2010 - Babraham Institute, Babraham, UK (invited by Len Stephens) “Modelling the behaviour of allosteric calcium sensors involved in synaptic plasticity - Cooperativity, sensitivity and all that”.

2010 - Institut Pasteur, Paris, FR (invited by Uwe Maskos) “Modélisation du comportement des senseurs allosteriques du calcium impliqués dans la plasticité synaptique - Cooperativité, sensibilité et tout ça”.

2010 - Institut Génétique Biologie Moléculaire Cellulaire, Ilkirch, FR (invited by Olivier Pourquie) “Modelling the behaviour of allosteric calcium sensors involved in synaptic plasticity - Cooperativity, sensitivity and all that”.

2010 - University of Rostock, Rostock, DE (invited by Dagmar Waltemath) “Toward a consistent set of interoperable standards to represent models and simulations”.

2009 - EBI Industry program meeting on Toxicogenomics, Hinxton, UK (invited by Dominic Clark) “Towards a public repository for pharmacodynamic/pharmacokinetic models”.

2009 - National Centre for Biological Sciences, Bangalore, IN (invited by Upinder Bhalla) “SBML, where it’s been, where it’s going” followed by “The problem of combinatorial explosion”,

2009 - Institut des Neurosciences de Bordeaux (INB), Bordeaux, FR (invited by Christophe Mulle) “Modélisation du comportement des protéines allostériques impliquées dans la plasticité synaptique”.

2009 - Laboratoire Bordelais de Recherche en Informatique (LaBRI), Bordeaux, FR (invited by Guy Melançon) “The Systems Biology Graphical Notation”.

2009 - INCF Task Force for a Standard Language in Neural Network Modeling, Tokyo, JP. “Lessons from developing the Systems Biology Markup Language”.

2009 - NeuroML workshop, London, UK. “Quick overview of the Systems Biology Markup Language”.

2008 - Zentrum Für Molekulare Biologie, Heidelberg, DE (invited by Victor Sourjik). “Remember or forget : allosteric changeover switches and molecular memory”.

2008 - Hartree Centre workshop : Towards the virtual cell , Abingdon, UK. “Simulations in Systems Neurobiology”.

2008 - OCISB launch meeting, Oxford, UK. “Integrators, loops and switches. The mechanic of learning”.

2007 - AIP Séquençage et CRB *Du séquençage à la biologie intégrative*, Paris, FR. “Ressources Europeennes pour la biologie des systemes”.

2007 - EBI Alumni meeting, Hinxton, UK. “Integrators, loops and switches. The mechanic of learning”.

2007 - Joint meeting EBI-Cambridge University, Hinxton, UK. “Computational Models of Synaptic Plasticity”.

2007 - Institut Curie, Paris, FR. “Modeling synaptic plasticity at molecular and sub-cellular levels”

2007 - *Storage and Annotation of Reaction Kinetics Data*, Heidelberg, DE. Controlled annotation of kinetics models,

2007 - *Systems Biology and Psychiatry - the intracellular dopamine signaling network and schizophrenia*, Munich, DE. “Modeling the intracellular dopamine signaling network : DARPP32 - a robust signal integrator”.

2006 - EBI philosophical society, Hinxton, UK. “Signalling : The hourglass, the bomb and the ruler”.

2006 - Systems Biology Research Centre, Newcastle, UK. “BioModels.net : Semantics for model sharing”.

2006 - BioScope-IT Annual Meeting, Ghent, BE. “Standards and Resources in Systems Biology : collaborative scale-up toward virtual life”.

2006 - Department of Biochemistry, University of Oxford, UK (invited by Mark Sansom). “Mesoscopic modelling of receptive membranes”.

2006 - *When Neuroinformatics meets Systems Biology*, Edinburgh, UK (invited by David Willshaw). “Importance of standards and model exchange in computational systems neurobiology”.

2005 - Société de Biologie, Paris, (invited by Jean Rossier). “Modélisation de l’intégration des signaux dans le neurone (Computational Systems Neurobiology). La biologie après le séquençage des génomes”.

2005 - seminaire VICANNE *Aspects stochastiques de la modélisation des réseaux de régulations*, Sophia-Antipolis. “Apports de la simulation stochastique des particules en « Systems Biology ». Exemple de la transduction des signaux transmembranaires”.

2004 - Programme Épigenomique - atelier de réflexion, Evry, FR. “Modélisation systémique de la transduction du signal dans les neurones. Changements d’échelles et niveaux d’analyse”.

2004 - Institut d’Histoire et de Philosophie des Sciences (IHPST), Paris (invited by Jean-Claude Dupont). “Aspects systémiques de la communication neuronale”.

2004 - Institut du Fer à Moulin, Paris (invited by Jean-Antoine Girault). “Toward a Systems Biology of the Neuron”.

2004 - N+N NSF/EPSRC Workshop in High Performance Computing for Biomolecules and Materials, Washington, USA. “Toward a Computational Systems Neurobiology”.

2002 - École Polytechnique Fédérale de Lausanne, Lausanne, Suisse (invited by Horst Vogel). “Molecular modelings shed lights on nicotinic receptor structure and fonction”.

2002 - European Bioinformatics Institute, Hinxton, Royaume-Uni (invited by Janet Thornton). “Bioinformatics of Ligand-Gated Ion Channels”.

2002 - École Polytechnique Fédérale de Lausanne, Lausanne, Suisse (invited by Henri Markram). “Time of realism has come for the modelling of post-synaptic densities”.

2002 - Commissariat à l’Énergie Atomique, Saclay (invited by André Menez). “nAChRs : From primary to quaternary structures”.

1999 - Collège de France, Paris (invited by Jean-Pierre Changeux). “Bases moléculaires et comportementales de la dépendance à la nicotine”.

1998 - Séminaire Algorithmique et Biologie, Institut Pasteur, Paris. “La prédiction de structure secondaire des protéines : théorie et pratique”.

1998 - European course on Protein folding and structure prediction, Torino, Italy. “Protein secondary structure prediction. Principles and methods”.

1996 - 4$^{\text{ème}}$ colloque de l’école doctorale neurobiologie et comportement, Paris. “Vers l’identification d’un composé responsable de la dépendance à la nicotine”.

B.4.3 Cours

2012 - EBI-Wellcome Trust Summer School in Bioinformatics, Hinxton, UK “What is Systems Biology ? Where does it come from ? What are the challenges ahead ?”

2012 - WT/EBI course In Silico Systems Biology, Cambridge, UK “Modeling chemical kinetics”

2012 - WT/EBI course In Silico Systems Biology, Cambridge, UK “Models For All. Standards for describing the whole life-cycle of modeling in the life sciences”

2012 - WT/EBI course In Silico Systems Biology, Cambridge, UK “What is Systems Biology ? Where does it come from ? What are the challenges ahead ?”

2012 - Instituto Gulbenkian de Ciência PhD student course, Lisbon, PT “Models For All - Standards for Describing the Whole Life-Cycle of Modeling in Life Sciences”

2012 - Instituto Gulbenkian de Ciência PhD student course, Lisbon, PT “Modelling chemical kinetics”

2012 - Instituto Gulbenkian de Ciência PhD student course, Lisbon, PT “What is Systems Biology ? What are the challenges ahead ?”

2011 - SystemsX.ch/SIB Summer School 2011, Kandersteg, CH “Models For All - Standards for Describing the Whole Life-Cycle of Modeling in Life Sciences”

2011 - SystemsX.ch/SIB Summer School 2011, Kandersteg, CH “Modelling chemical kinetics”

2011 - EBI-Wellcome Trust Summer School in Bioinformatics, Hinxton, UK “Systems Biology : Where it comes from, what it is, and what it does”.

2011 EBI - FEBS : In Silico Systems Biology : Network Reconstruction, Analysis and Network-based Modelling, Cambridge, UK “Modelling chemical kinetics, Hands-on training”.

2010 - EBI-Wellcome Trust Summer School in Bioinformatics, Hinxton, UK “Systems Biology : Where it comes from, what it is, and what it does”.

2010 - EMBO Practical Course *In silico systems biology : network reconstruction, analysis and network based modellin*, Cambridge, UK “Modelling chemical kinetics”.

2010 - EMBO Practical Course *In silico systems biology : network reconstruction, analysis and network based modelling*, Cambridge, UK “Systems Biology”.

2009 - International Genetically Engineered Machine competition (iGEM), Cambridge, UK “Modelling chemical kinetics”.

2009 - EBI course *Interactions and Pathways*, Hinxton, UK “Systems Biology”.

2008 - EBI Open Day, Hinxton, UK “The truth about systems biology”.

2007 - International Genetically Engineered Machine competition (iGEM), Cambridge, UK “Computational models in systems biology”.

2007 - Winter School on Systems Biology for Medical Applications, Puerto de la Cruz, Tenerife, ES. “Systems Biology of neuronal signalling”.

2006 - Okinawa Computational Neuroscience Course 2006, OIST, JP. “Mesoscopic simulations of receptive lattices”.

2006 - Okinawa Computational Neuroscience Course 2006, OIST, JP. “Integration of Dopamine and Glutamate signals by DARPP-32”.

2006 - Okinawa Computational Neuroscience Course 2006, OIST, JP. “Bioinformatics resources and standards for modeling neuronal signalling”.

2005 - Symbionic course on *Methods, data handling and standards in neuronal systems biology*, Trieste, IT. “Computational Systems Neurobiology”.

2005 - Symbionic course on *Methods, data handling and standards in neuronal systems biology*, Trieste, IT. “Curation, exchange annotation of kinetic models : The BioModels.net initiative”.

2005 - Symbionic course on *Methods, data handling and standards in neuronal systems biology*, Trieste, IT. “Controlled Vocabularies in Systems Biology”.

2004 - ESF course *Modelling Metabolic and Signal Transduction Networks*, Oxford, UK. “Computational Systems Neurobiology”.

2004 - EBI course for EMBL PhD students, Hinxton, UK “Computational Systems Biology”.

2002 - DEA de Neurobiologie et Pharmacologie, École Normale Supérieure de Paris. “Structure des récepteurs-canaux de la famille à « cys-loop » ”.

C Liste de publications

La plupart des tirés à part peuvent être téléchargés à `http://lenoverelab.org/perso/lenov/`. Les nombres de citations sont fournis par Google Scholar en octobre 2013 (total=11418, h-index=45).

C.1 Publications dans des journaux à comité de lecture

87. CLAUDINE CHAOUIYA, DUNCAN BERENGUIER, SARAH M KEATING, AURELIEN NALDI, MARTIJN P. VAN IERSEL, NICOLAS RODRIGUEZ, ANDREAS DRÄGER, FINJA BÜCHEL, THOMAS COKELAER, BRYAN KOWAL, BENJAMIN WICKS, EMANUEL GONÇALVES, JULIEN DORIER, MICHEL PAGE, PEDRO T. MONTEIRO, AXEL VON KAMP, IOANNIS XENARIOS, HIDDE DE JONG, MICHAEL HUCKA, STEFFEN KLAMT, DENIS THIEFFRY, NICOLAS LE NOVÈRE, JULIO SAEZ-RODRIGUEZ, TOMÁŠ HELIKAR. SBML Qualitative Models : a model representation format and infrastructure to foster interactions between qualitative modelling formalisms and tools. *Submitted*. [cit=2]

86. FINJA BÜCHEL, NICOLAS RODRIGUEZ, NEIL SWAINSTON, CLEMENS WRZODEK, TOBIAS CZAUDERNA, ROLAND KELLER, FLORIAN MITTAG, MICHAEL SCHUBERT, MIHAI GLONT, MARTIN GOLEBIEWSKI, MARTIJN VAN IERSEL, SARAH KEATING, MATTHIAS RALL, MICHAEL WYBROW, HENNING HERMJAKOB, MICHAEL HUCKA, DOUGLAS B. KELL, WOLFGANG MÜLLER, PEDRO MENDES, ANDREAS ZELL, CLAUDINE CHAOUIYA, JULIO SAEZ-RODRIGUEZ, FALK SCHREIBER, CAMILLE LAIBE, ANDREAS DRÄGER, NICOLAS LE NOVÈRE Large-scale generation of computational models from biochemical pathway maps. *BMC Systems Biology*, accepted. [cit=1]

85. AJMERA I., SWAT M., LAIBE C., LE NOVÈRE N., CHELLIAH V. The impact of mathematical modeling on the understanding of diabetes and related complications. (2013) *CPT : Pharmacometrics & Systems Pharmacology*, 2 : e54.

84. KELLER R., DÖRR A., TABIRA A., FUNAHASHI A., ZILLER M.J., ADAMS R., RODRIGUEZ N., LE NOVÈRE N., HIROI N., PLANATSCHER H., ZELL A., DRÄGER A. The systems biology simulation core algorithm. (2013) *BMC Systems Biology*, 7 : 55. [cit=1]

83. MATTIONI M., LE NOVÈRE N. Integration of biochemical and electrical signaling - multiscale model of the medium spiny neuron of the striatum. (2013) *PLoS ONE*, 8(7) : e66811

82. STEFAN M.I., LE NOVÈRE N. Cooperative binding. (2013) *PLoS Computational Biology*, 9(6) : e1003106.

81. KNÜEPFER C, BECKSTEIN C, DITTRICH P, LE NOVÈRE N. Structure, function, and behaviour of computational models in systems biology. (2013) *BMC Systems Biology* 7(1) :43.

80. EDELSTEIN S.J., LE NOVÈRE N. Cooperativity of allosteric receptors. (2013) *Journal of Molecular Biology*, 425(9) : 1424-1432. [cit=3]

79. THIELE I., SWAINSTON N., FLEMING R.M.T., HOPPE A., SAHOO1 S., AURICH M.K., HARALDSDOTTIR H., MO M.L., ROLFSSON O., STOBBE M.D., THORLEIFSSON S.G., AGREN R., BÖLLING C., BORDEL S., CHAVALI A.K., DOBSON P., DUNN W.B., ENDLER L., GORYANIN I., GUDMUNDSSON S., HALA1 D., HUCKA M., HULL D., JAMESON D., JAMSHIDI N., JONSSON J.J., JUTY N., KEATING S., NOOKAEW I., LE NOVÈRE N., MALYS N., MAZEIN A., PAPIN J.A, PRICE N.D., SELKOV E., SIGURDSSON M.I., SIMEONIDIS E., SONNENSCHEIN N., SMALLBONE K., SOROKIN A., VAN BEEK H., WEICHART D., NIELSEN J.B., WESTERHOFF H.V, KELL D.B., MENDES P., PALSSON B.Ø. A community-driven global reconstruction of human metabolism. (2013) *Nature Biotechnology*, 31 : 419–42531 : 419–425. [cit=31]

78. VON EICHBORN J., DUNKEL M., GOHLKE B., PREISSNER S., HOFFMANN M., BAUER J., ARMSTRONG J.D., SCHAEFER M., ANDRADE M., LE NOVÈRE N., CRONING M., GRANT S.G.N., VAN NIEROP P., SMIT A.B., PREISSNER R. SynSysNet : Integration of experimental data on synaptic protein-protein interactions with drug-target relations. (2013) *Nucleic Acids Research*, 41(D1) :D834-D840. [cit=2]

77. ZHUKOVA A., ADAMS R., LAIBE C., LE NOVÈRE N. LibKiSAO : a Java Library for Querying KiSAO. (2012) *BMC Research Notes*, 5 : 520.

76. L. LI, M.I. STEFAN, N. LE NOVÈRE. Calcium input frequency, duration and amplitude differential modulate the relative activation of calcineurin and CaMKII. (2012) *PLoS ONE*, 7(9) : e43810. [cit=3]

75. BÜCHEL F., WRZODEK C., MITTAG F., DRÄGER A, EICHNER J., RODRIGUEZ N., LE NOVÈRE N., ZELL A. Qualitative translation of relations from BioPAX to SBML qual. (2012) *Bioinformatics*, 28(20) : 2648-2653. [cit=4]

74. ADAMS R R, TSORMAN N, STRATFORD K, AKMAN O, GILMORE S, JUTY N, LE NOVÈRE N, AND MILLAR A. The Input Signal Step Function (ISSF) : A Standard Method to Encode

Input Signals in SBML Models with Software Support, Applied to Circadian Clock Models. (2012) *Journal of Biological Rhythms*, 27 : 328-332. [cit=1]

73. M. DUMOUSSEAU, N. RODRIGUEZ, N. LE NOVÈRE. MELTING, a flexible platform to compute the melting temperature of nucleic acid. (2012) *BMC Bioinformatics*, 13 :101. [cit=1]

72. MATTIONI M., COHEN U., LE NOVÈRE N. Neuronvisio : a Graphical User Interface with 3D capabilities for NEURON. (2012) *Frontiers in Neuroinformatics*, 6 :20. [cit=2]

71. VAN IERSEL M.P., VILLÉGER A.C., CZAUDERNA T., BOYD S.E., BERGMANN F.T., LUNA A., DEMIR E., SOROKIN A., DOGRUSOZ U., MATSUOKA Y., FUNAHASHI A., ALADJEM M.I., MI H., MOODIE S.L., KITANO H., LE NOVÈRE N., SCHREIBER F. Software support for SBGN maps : SBGN-ML and LibSBGN. (2012) *Bioinformatics*, 28 : 2016-2021. [cit=10]

70. STEFAN M.I., MARSHALL D.P., LE NOVÈRE N. Structural analysis and stochastic modelling suggest a mechanism for calmodulin trapping by CaMKII. (2012) *PLoS ONE*, 7(1) : e29406. [cit=2]

69. JUTY N., LE NOVÈRE N., LAIBE C. Identifiers.org and MIRIAM Registry : community resources to provide persistent identification. (2012) *Nucleic Acids Research*, 40 : D580-D586. [cit=26]

68. D. WALTEMATH, R. ADAMS, F.T. BERGMANN, M. HUCKA, F. KOLPAKOV, A.K. MILLER, I.I. MORARU, D. NICKERSON, S. SAHLE, J.L. SNOEP, N. LE NOVÈRE N. Reproducible computational biology experiments with SED-ML - The Simulation Experiment Description Markup Language. (2011) *BMC Systems Biology*, 5 :198. [cit=19]

67. M. COURTOT, JUTY N., KNÜPFER C., WALTEMATH D., DRÄGER A., FINNEY A., GOLEBIEWSKI M., HASTINGS J., HOOPS S., KEATING S., KELL D.B., KERRIEN S., LAWSON J., LISTER A., LU J., MACHNE R., MENDES P., POCOCK M., RODRIGUEZ N., VILLEGER A., WILKINSON D.J., WIMALARATNE S., LAIBE C., HUCKA C., N. LE NOVÈRE. Controlled vocabularies and semantics in Systems Biology. (2011) *Molecular Systems Biology*, 7 : 543. [cit=57]

66. M. SCHULTZ, F. KRAUSE, N. LE NOVÈRE, E. KLIPP, W. LIEBERMEISTER. Retrieval, alignment, and clustering of computational models based on semantic annotations. (2011) *Molecular Systems Biology*, 7 : 512. [cit=11]

65. A DRÄGER, N. RODRIGUEZ, M. DUMOUSSEAU, A. DÖRR, C. WRZODEK, N. LE NOVÈRE, A. ZELL, M. HUCKA. JSBML : a flexible Java library for working with SBML. (2011) *Bioinformatics*, 27 : 2167-2168. [cit=26]

64. D. WALTEMATH, R. ADAMS, D.A. BEARD, F.T. BERGMANN, U.S. BHALLA, R. BRITTEN, V. CHELLIAH, M.T. COOLING, J. COOPER, E. CRAMPIN, A. GARNY, S. HOOPS, M. HUCKA, P. HUNTER, E. KLIPP, C. LAIBE, A. MILLER, I. MORARU, D. NICKERSON, P. NIELSEN, M. NIKOLSKI, S. SAHLE, H. SAURO, H. SCHMIDT, J.L. SNOEP, D. TOLLE, O. WOLKENHAUER, N. LE NOVÈRE. Minimum Information About a Simulation Experiment (MIASE). (2011) *PLoS Computational Biology*, 7(4) : e1001122. [cit=25]

63. DEMIR E, CARY MP, PALEY S, FUKUDA K, LEMER C, VASTRIK I, WU G, D'EUSTACHIO P, SCHAEFER C, LUCIANO J, SCHACHERER F, MARTINEZ-FLORES I, HU Z, JIMENEZ-JACINTO V, JOSHI-TOPE G, KANDASAMY K, LOPEZ-FUENTES AC, MI H, PICHLER E, RODCHENKOV I, SPLENDIANI A, TKACHEV S, ZUCKER J, GOPINATHRAO G, RAJASIMHA H, RAMAKRISHNAN R, SHAH I, SYED M, ANWAR N, BABUR O, BLINOV M, BRAUNER E, CORWIN D, DONALDSON S, GIBBONS F, GOLDBERG R, HORNBECK P, LUNA A, MURRAY-RUST P, NEUMANN E, REUBENACKER O, SAMWALD M, VAN IERSEL M, WIMALARATNES, ALLEN K, BRAUN B, CARRILLO M, CHEUNG KH, DAHLQUIST K, FINNEY A, GILLESPIE M, GLASS E, GONG L, HAW R, HONIG M, HUBAUT O, KANE D, KRUPA S, KUTMON M, LEONARD J, MARKS D, MERBERG D, PETRI V, PICO A, RAVENSCROFT D, REN L, SHAH N, SUNSHINE M, TANG R, WHALEY R, LETOVKSY S, BUETOW KH, RZHETSKY A, SCHACHTER V, SOBRAL BS, DOGRUSOZ U, MCWEENEY S, ALADJEM M, BIRNEY E, COLLADO-VIDES J, GOTO S, HUCKA M, LE NOVÈRE N, MALTSEV N, PANDEY A, THOMAS P, WINGENDER E, KARP PD, SANDER C, BADER GD. BioPAX – A Community Standard for Pathway Data Sharing. (2010) *Nature Biotechnology*, 28 : 935–942. [cit=188]

62. R. HENKEL, L. ENDLER, A. PETERS, N. LE NOVÈRE, D. WALTEMATH. Ranked Retrieval of Computational Biology Models. (2010) *BMC Bioinformatics*, 11 :423. [cit=13]

61. C. LI, M. DONIZELLI, N. RODRIGUEZ, H. DHARURI, L. ENDLER, V. CHELLIAH, L. LI, E. HE, A. HENRY, M.I. STEFAN, J.L. SNOEP, M. HUCKA, N. LE NOVÈRE, C. LAIBE. BioModels Database : An enhanced, curated and annotated resource for published quantitative kinetic models. (2010) *BMC Systems Biology*, 4 : 92. [cit=171]

60. C. LI, M. COURTOT, C. LAIBE, N. LE NOVÈRE. BioModels.net Web Services, a free and integrated toolkit for computational modelling software. (2010) *Briefings in Bioinformatics*, 11 : 270-277. [cit=25]

59. D. TOLLE, N. LE NOVÈRE. Brownian Diffusion of AMPA Receptors Is Sufficient to Explain Fast Onset of LTP. (2010) *BMC Systems Biology*, 4 : 25 . [cit=6]

58. D. TOLLE, N. LE NOVÈRE. Meredys, a multi-compartment reaction-diffusion simulator using multistate realistic molecular complexes. (2010) *BMC Systems Biology*, 4 : 24. [cit=12]

57. S.J. EDELSTEIN, M.I. STEFAN, N. LE NOVÈRE. Ligand depletion in vivo modulates the dynamic range of cooperative signal transduction. (2010) *PLoS ONE*, 5(1) : e8449. [cit=5]

56. N. LE NOVÈRE, M. HUCKA, H. MI, S. MOODIE, F. SHREIBER, A. SOROKIN, E. DEMIR, K. WEGNER, M. ALADJEM, S. WIMALARATNE, F.T. BERGMAN, R. GAUGES, P. GHAZAL, K. HIDEYA, L. LI, Y. MATSUOKA, A. VILLÉGER, S.E. BOYD, L. CALZONE, M. COURTOT, U. DOGRUSOZ, T. FREEMAN, A. FUNAHASHI, S. GHOSH, A. JOURAKU, S. KIM, F. KOLPAKOV, A. LUNA, S. SAHLE, E. SCHMIDT, S. WATTERSON, I. GORYANIN, D.B. KELL, C. SANDER, H. SAURO, J.L. SNOEP, K. KOHN, H. KITANO. The Systems Biology Graphical Notation. (2009) *Nature Biotechnology*, 27 : 735-741. [cit=285]

55. M.I. STEFAN, S.J. EDELSTEIN, N. LE NOVÈRE. Computing phenomenologic Adair-Klotz constants from microscopic MWC parameters. (2009) *BMC Systems Biology*, 3 : 68. [cit=4]

54. L. ENDLER, N. RODRIGUEZ, N. JUTY, V. CHELLIAH, C. LAIBE, C., LI, N. LE NOVÈRE. Designing and encoding models for Synthetic Biology. (2009) *Journal of the Royal Society Interface*, 6 : S405-S417. [cit=34]

53. O. WOLKENHAUER, D. FELL, P. DE MEYTS, N. BLÜTHGEN, H. HERZEL, N. LE NOVÈRE, T. HÖFER, K. SCHÜRRLE, I. VAN LEEUWEN. Advancing systems biology for medical applications. (2009) *IET Systems Biology*, 3 : 131-136. [cit=18]

52. M.J. HERRGÅRD, N. SWAINSTON, P. DOBSON, W.B. DUNN, K.Y. ARGA, M. ARVAS, N. BLÜTHGEN, S. BORGER, R. COSTENOBLE, M. HEINEMANN, M. HUCKA, N. LE NOVÈRE, P. LI, W. LIEBERMEISTER, M.L. MO, A.P. OLIVEIRA, D. PETRANOVIC, S. PETTIFER, E. SIMEONIDIS, K. SMALLBONE, I. SPASIĆ, D. WEICHART, R. BRENT, D.S. BROOMHEAD, H.V. WESTERHOFF, B. KIRDAR, M. PENTTILÄ, E. KLIPP, B.Ø. PALSSON, SAUER U., OLIVER S.G., MENDES P., NIELSEN J., KELL D.B.. A consensus yeast metabolic network reconstruction obtained from a community approach to systems biology. (2008) *Nature Biotechnology*, 26 : 1155-1160. [cit=280]

51. G. GUERLET, A. TALY, L. PRADO DE CARVAHLO, A. MARZ, R. JIANG, A. SPECHT, N. LE NOVÈRE, T. GRUTTER. Comparative models of P2X2 receptor support intersubunit ATP-binding sites. (2008) *Biochemical and Biophysical Research Communications*, 375 : 405-409. [cit=8]

50. C.F. TAYLOR, D. FIELD, S.-A. SANSONE, J. AERTS, R. APWEILER, M. ASHBURNER, C.A. BALL, P.-A. BINZ, M. BOGUE, A. BRAZMA, R. BRINKMAN, A.M. CLARK, E.W. DEUTSCH, O. FIEHN, J. FOSTEL, P. GHAZAL, F. GIBSON, T. GRAY, G. GRIMES, N.W. HARDY, H. HERMJAKOB, R.K. JULIAN, M. KANE, C. KETTNER, C. KINSINGER, E. KOLKER, M. KUIPER, N. LE NOVÈRE, J. LEEBENS-MACK, S.E. LEWIS, P. LORD, A.-M. MALLIN, N. MARTHANDAN, H. MASUYA, R. MCNALLY, A. MEHRLE, N. MORRISON, S. ORCHARD, J. QUACKENBUSH, J.M. REECY, D.G. ROBERTSON, P. ROCCA-SERRA, H. RODRIGUEZ, H. ROSENFELD, J. SANTOYO-LOPEZ,

R.H. SCHEUERMANN, D. SCHOBER, B. SMITH, J. SNAPE, P. STERK, K. TIPTON, A. UNTERGASSER, J. VANDESOMPELE, S. WIEMANN. Promoting Coherent Minimum Reporting Requirements for Biological and Biomedical Investigations : The MIBBI Project. (2008) *Nature Biotechnology*, 26 : 889-896. [cit=268]

49. M.I. STEFAN, S.J. EDELSTEIN, N. LE NOVÈRE. An allosteric model of calmodulin explains differential activation of PP2B and CaMKII. (2008) *Proceedings of the National Academy of Sciences USA*, 105 : 10768-10773. [cit=33]

48. N. LE NOVÈRE, L. LI, J.-A. GIRAULT. DARPP-32 : Molecular integration of phosphorylation potential. (2008) *Cellular and Molecular Life Sciences*, 65 : 2125-2127. [cit=10]

47. C. LAIBE, N. LE NOVÈRE. MIRIAM Resources : tools to generate and resolve robust cross-references in Systems Biology. (2007) *BMC Systems Biology*, 1 : 58. [cit=68]

46. N. RODRIGUEZ, M. DONIZELLI, N. LE NOVÈRE. SBMLeditor : effective creation of models in the Systems Biology Markup Language (SBML). (2007) *BMC Bioinformatics*, 8 : 79. [cit=28]

45. E. FERNANDEZ, R. SCHIAPPA, J.A. GIRAULT, N. LE NOVÈRE. DARPP-32 is a robust integrator of dopamine and glutamate signals. (2006) *PLoS Computational Biology*, 2 : e176. [cit=61]

44. A.H.V. SCHAPIRA, E. BEZARD, J. BROTCHIE, F. CALON, G.L. COLLINGRIDGE, B. FERGER, B. HENGERER, E. HIRSCH, P. JENNER, N. LE NOVÈRE, J.A. OBESO, M.A. SCHWARZSCHILD, U. SPAMPINATO, G. DAVIDAI. Novel pharmacological targets for the treatment of Parkinson's disease. (2006) *Nature Reviews Drug Discovery*, 5 : 845-854. [cit=167]

43. D. TOLLE, N. LE NOVÈRE. Particle-based Stochastic Simulation in Systems Biology. (2006) *Current Bioinformatics*, 1 : 315-320. [cit=42]

42. SCHILSTRA M.J., LI L., MATTHEWS J., FINNEY A., HUCKA M., N. LE NOVÈRE. CellML2SBML : Conversion of CellML into SBML. (2006) *Bioinformatics*, 18 : 1018-1020. [cit=30]

41. M. DONIZELLI, M.A. DJITE, N. LE NOVÈRE. LGICdb. A manually curated sequence database after the genomes. (2006) *Nucleic Acids Research*, 34 : D267-D269. [cit=14]

40. N. LE NOVÈRE, B. BORNSTEIN, A. BROICHER, M. COURTOT, M. DONIZELLI, H. DHARURI, L. LI, H. SAURO, M. SCHILSTRA, B. SHAPIRO, J.L. SNOEP, M. HUCKA. BioModels Database : A Free, Centralized Database of Curated, Published, Quantitative Kinetic Models of Biochemical and Cellular Systems. (2006) *Nucleic Acids Research*, 34 : D689-D691. [cit=491]

39. N. LE NOVÈRE, A. FINNEY, M. HUCKA, U. BHALLA, F. CAMPAGNE, J. COLLADO-VIDES, E.D. CRAMPIN, M. HALSTEAD, E. KLIPP, P .MENDES, P. NIELSEN, H. SAURO, B. SHAPIRO, J.L. SNOEP, H.D. SPENCE, B.L. WANNER. Minimal Information Requested In the Annotation of biochemical Models (MIRIAM). (2005) *Nature Biotechnology* 23 : 1509-1515. [cit=337]

38. A. TALY, M. DELARUE, T. GRUTTER, M. NILGES, N. LE NOVÈRE, P.-J. CORRINGER, J.-P. CHANGEUX. Normal mode analysis suggest a quaternary twist model for the nicotinic receptor gating mechanism. (2005) *Biophysical Journal*, 88 : 3954-3965. [cit=142]

37. N. LE NOVÈRE N., M. DONIZELLI. The Molecular Pages of the Mesotelencephalic Dopamine Consortium (DopaNet). (2004) *BMC bioinformatics*, 5 : 174. [cit=7]

36. T. GRUTTER, N. LE NOVÈRE, J.-P. CHANGEUX. Rational understanding of nicotinic receptors drug binding. (2004) *Current Topics in Medicinal Chemistry*, 4 : 645-651. [cit=24]

35. J. SALLETTE, S. BOHLER, P. BENOIT, M. SOUDANT, S. PONS, N. LE NOVÈRE, J.-P. CHANGEUX, P.-J. CORRINGER. An extracellular microdomain controls up-regulation of neuronal nicotinic acetylcholine receptors by nicotine. (2004) *Journal of Biological Chemistry*, 279 : 18767-18775. [cit=69]

34. N. CHAMPTIAUX, C. GOTTI, M. CORDERO-ERAUSQUIN, D. DAVID, C. PRZYBYLSKI, C. LÉNA, F. CLEMENTI, M. MORETTI, F.M. ROSSI, N. LE NOVÈRE, J.M. MCINSTOSH, A.M. GARDIER, J.-P. CHANGEUX. Subunit composition of functional nicotinic receptors in dopaminergic neurons investigated with knock-out mice. (2003) *Journal of Neuroscience*, 23 : 7820-7829. [cit=357]

33. Z.-Y. HAN, M. ZOLI, A. CARDONA, J.-P. BOURGEOIS, J.-P. CHANGEUX, N. LE NOVÈRE Localization of [^{3}H]-nicotine, [^{3}H]-cytisine, [^{3}H]-epibatidine and [^{125}I]-α-bungarotoxin binding sites in the brain of Macaca Mulatta. (2003) *Journal of Comparative Neurology*, 461 :49-60. [cit=62]

32. T. GRUTTER, L. PRADO DE CARVALHO, N. LE NOVÈRE, P.-J. CORRINGER, S. EDELSTEIN, J.-P. CHANGEUX. An interaction between two residues from different loops of the acetylcholine-binding site contributes to the activation mechanism of nicotinic receptors. (2003) *EMBO Journal*, 22 : 1990-2203. [cit=47]

31. V. COURTOIS, G. CHATELAIN, Z.-Y. HAN, N. LE NOVÈRE, G. BRUN, T. LAMONERIE. New Otx2 mRNA isoforms expressed in the mouse brain. (2003) *Journal of Neurochemistry*, 84 : 840-853. [cit=24]

30. M. HUCKA M., H. BOLOURI, A. FINNEY, H.M. SAURO, J.C. DOYLE, H. KITANO, A.P. ARKIN, B.J. BORNSTEIN, D. BRAY, A.A. CUELLAR, S. DRONOV, M. GINKEL, V. GOR, I.I. GORYANIN, W.J. HEDLEY, T.C. HODGMAN, P.J. HUNTER, N.S. JUTY,

J.L. KASBERGER, A. KREMLING, U. KUMMER, N. LE NOVÈRE, L.M. LOEW, D. LUCIO, P. MENDES, E.D. MJOLSNESS, Y. NAKAYAMA, M.R. NELSON, P.F. NIELSEN, T. SAKURADA, J.C. SCHAFF, B.E. SHAPIRO, T.S. SHIMIZU, H.D. SPENCE, J. STELLING, K. TAKAHASHI, M. TOMITA, J. WAGNER, J. WANG. The Systems Biology Markup Language (SBML) : A Medium for Representation and Exchange of Biochemical Network Models. (2003) *Bioinformatics*, 19 : 524-531. [cit=1827]

29. C. FRUCHART-GAILLARD, B. GILQUIN, S. ANTIL-DELBEKE, N. LE NOVÈRE, T. TAMIYA, P.-J. CORRINGER, J.-P. CHANGEUX, A. MÉNEZ, D. SERVENT. Experimentally-based model of a complex between a snake toxin and the $\alpha 7$ nicotinic receptor. (2002) *Proceedings of the National Acadademy of Science USA*, 99 : 3216-3221. [cit=111]

28. N. LE NOVÈRE, T. GRUTTER, J.-P. CHANGEUX. Models of the extracellular domain of the nicotinic receptors and of agonist and Ca^{++} binding sites. (2002) *Proceedings of the National Acadademy of Science USA*, 99 : 3210-3215. [cit=268]

27. N. LE NOVÈRE, P.-J. CORRINGER, J.-P. CHANGEUX. The diversity of subunit composition in nAChRs : evolutionary origins, physiological and pharmacological consequences. (2002) *Journal of Neurobiology*, 53 : 447-456. [cit=285]

26. R. FERRARI, N. LE NOVÈRE, M.R. PICCIOTTO, J.-P. CHANGEUX, M. ZOLI. Acute and long-term changes in dopamine mesoaccumbens pathway after systemic or local single nicotine injections. (2002) *European Journal of Neuroscience*, 15 : 1810-1818. [cit=97]

25. N. LE NOVÈRE. MELTING, a free tool to compute the melting temperature of nucleic acid duplex. *Bioinformatics*, 17 : 1226-1227. [cit=112]

24. N. LE NOVÈRE, T.S. SHIMIZU. STOCHSIM : Modelling of stochastic biomolecular processes. *Bioinformatics*, 17 : 575-576. [cit=206]

23. N. LE NOVÈRE, J.-P. CHANGEUX. LGICdb : The Ligand Gated Ion Channel database. (2001) *Nucleic Acids Research*, 29 : 294-295. [cit=69]

22. N. LE NOVÈRE, J.-P. CHANGEUX. The Ligand Gated Ion Channel database (LGICdb), an example of a sequence database in neuroscience. (2001) *Philosophical Transactions of the Royal Society*, 356 : 1121-1130. [cit=22]

21. T.A.J. DUKE, N. LE NOVÈRE, D. BRAY. Conformational spread in a ring of proteins : a stochastic approach to allostery. (2001) *Journal of Molecular Biology*, 308 : 541-553. [cit=137]

20. P.-J. CORRINGER, N. LE NOVÈRE, J.-P. CHANGEUX. Nicotinic Receptors at the Amino Acid Level. (2000) *Annual Reviews of Pharmacology and Toxicology*, 40 : 431-458. [cit=718]

19. Z.-H. HAN, N. LE NOVÈRE, M. ZOLI, N. CHAMPTIAUX, J.A. HILL, J.-P. CHANGEUX. Localisation of nAChR subunit mRNAs in the brain of *Macaca mulatta*. (2000) *European Journal of Neuroscience*, 12 : 3664-3674. [cit=91]

18. T.S. SHIMIZU†, N. LE NOVÈRE†, M.D. LEVIN, A. BEAVIL, B. SUTTON, D. BRAY. Molecular model of a lattice of signalling proteins involved in bacterial chemotaxis. (2000) *Nature Cell Biology*, 2 : 792-796. (†auteurs ex-æquo). [cit=184]

17. U.C. ROGNER, D.D. SPYROPOULOS, N. LE NOVÈRE, J.-P. CHANGEUX, P. AVNER. Control of neurulation by the nucleosomal assembly protein 1l2. (2000) *Nature genetics*, 25 : 431-435. [cit=59]

16. N. LE NOVÈRE, M. ZOLI, C. LÉNA, R. FERRARI, M.R. PICCIOTTO, E. MERLO-PICH, J.-P. CHANGEUX. Involvement of α6 nicotinic receptor subunit in nicotine-elicited locomotion, demonstrated by *in vivo* antisense oligonucleotide infusion. (1999) *NeuroReport*, 10 : 2497-2401. [cit=84]

15. N. LE NOVÈRE, P.-J. CORRINGER, J.-P. CHANGEUX. Improved secondary structure prediction of a nicotinic receptor subunit. Incorporation of solvent accessibility and experimental data into a 2D representation. (1999) *Biophysical Journal*, 76 : 2329-2345. [cit=97]

14. N. LE NOVÈRE, J.-P. CHANGEUX. The Ligand Gated Ion Channel database. (1999) *Nucleic Acids Research*, 27 : 340-342. [cit=84]

13. R.J. LUKAS, J.-P. CHANGEUX, N. LE NOVÈRE, E.X. ALBUQUERQUE, D.J.K. BALFOUR, D.K. BERG, D. BERTRAND, V.A. CHIAPPINELLI, P.B.S. CLARKE, A.C. COLLINS, J.A. DANI, S.R. GRADY, K.J KELLAR, J.M. LINDSTROM, M.J. MARKS, M. QUIK, P.W. TAYLOR, S. WONNACOTT. International Union of Pharmacology Recommendations for the Nomenclature of Nicotinic Acetylcholine Receptors. (1999) *Pharmacological reviews*, 51 : 397-401. [cit=475]

12. C. LÉNA, A. DE KERCHOVE D'EXAERDE, M. CORDERO-ERAUSQUIN, N. LE NOVÈRE, M.M. ARROYO-JIMENEZ, J.-P. CHANGEUX. Diversity and distribution of nicotinic acetylcholine receptors in the *locus ceruleus* neurons. (1999) *Proceedings of the National Acadademy of Science USA*, 96 : 12127-12131. [cit=103]

11. L.M. MARUBIO, M.M. ARROYO-JIMENEZ, M. CORDERO-ERAUSQUIN, N. LE NOVÈRE, C. LÉNA, M. HUCHET, M.I. DAMAJ, J.-P. CHANGEUX. Loss of nicotine-induced antinociception in mice lacking the neuronal nicotinic α4 subunit. (1999) *Nature*, 398 : 805-810. [cit=475]

10. J.-P. CHANGEUX, D. BERTRAND, P.-J. CORRINGER, S. DEHAENE, S. EDELSTEIN, C. LÉNA, N. LE NOVÈRE, L.M. MARUBIO, M.R. PICCIOTTO, M. ZOLI. Brain nicotinic receptors : structure and regulation, role in learning and reinforcement. (1998) *Brain Research Reviews*, 26 : 198-216. [cit=254]

9. R. BEERI, N. LE NOVÈRE, R. MERVIS, T. HUBERMAN, E. GRAUER, J.-P. CHANGEUX, H. SOREQ. Enhanced hemicholinium binding and attenuated dendrite branching in cognitively impaired acetylcholinesterase-transgenic mice. (1997) *Journal of Neurochemistry*, 68 : 2441-2451. [cit=99]

8. N. LE NOVÈRE, M. ZOLI, J.-P. CHANGEUX. Neuronal nicotinic receptor α6 subunit mRNA is selectively concentrated in catecholaminergic nuclei of the rat brain. (1996) *European Journal of Neuroscience*, 8 : 2428-2439. [cit=341]

7. J.-P. CHANGEUX, A. BESSIS, J.-P. BOURGEOIS, P.-J. CORRINGER, A. DEVILLERS-THIÉRY, J.-L. EISELÉ, M. KERSZBERG, C. LÉNA, N. LE NOVÈRE, M.R. PICCIOTTO, M. ZOLI. Nicotinic receptors and brain plasticity. (1996) *Cold Spring Harbor Symposia in Quantitative Biology*, LXI : 343-362. [cit=19]

6. N. LE NOVÈRE, J.-P. CHANGEUX. Molecular evolution of the nicotinic acetylcholine receptor subunit family : an example of multigene family in excitable cells. (1995) *Journal of Molecular Evolution*, 40 : 155-172. [cit=353]

5. M. ZOLI, N. LE NOVÈRE, J.A. HILL, J.-P. CHANGEUX. Developmental regulation of nicotinic ACh receptor subunit mRNAs in the rat central and peripheral nervous systems. (1995) *Journal of Neuroscience*, 15 : 1912-1939. [cit=274]

4. A. BESSIS, A.-M. SALMON, M. ZOLI, N. LE NOVÈRE, M.R. PICCIOTTO, J.-P. CHANGEUX. Promoter elements conferring neuron-specific expression of the beta2 subunit of the neuronal nicotinic acetylcholine receptor studied in vitro and in transgenic mice. (1995) *Neuroscience*, 69 : 807-819. [cit=60]

3. X.-M. LI, M. ZOLI, U.-B. FINNMAN, N. LE NOVÈRE, J.-P. CHANGEUX, K. FUXE. A single (-)–nicotine injection causes change with a time delay in the affinity of striatal D2 receptors for antagonist, but not for agonist, nor in the D2 receptor mRNA levels in the rat substantia nigra. (1995) *Brain Research*, 679 : 157-167. [cit=17]

2. M.R. PICCIOTTO, M. ZOLI, C. LÉNA, A. BESSIS, Y. LALLEMAND, N. LE NOVÈRE, P. VINCENT, E. MERLO-PICH, P. BRÛLET, J.-P. CHANGEUX. Abnormal avoidance learning in mice lacking functional high-affinity nicotine receptor in the brain. (1995) *Nature*, 374 : 65-67. [cit=523]

1. N. LE NOVÈRE, A. BESSIS, C. LÉNA, M.R. PICCIOTTO, M. ZOLI. Le récepteur nicotinique neuronal de l'acétylcholine : du gène au tabagisme. (1993) *Médecine-Sciences*, 9 : 41-49. [cit=1]

C.2 Publications dans des comptes rendus de conférences à comités de lecture

11. WALTEMATH D., WOLKENHAUER O., LE NOVÈRE N., DUMONTIER M. Possibilities for Integrating Model-related Data in Computational Biology. (2013) *Proceedings of the 9th International workshop Data Integration in the Life Sciences*

10. JUTY N., LE NOVÈRE N., HERMJAKOB H., LAIBE C. Towards the collaborative curation of the registry underlying Identifiers.org. (2013) *Database* 2013 :bat017

9. JUTY N., LE NOVÈRE N., HERMJAKOB H., LAIBE C. Delivering "Cool URIs" that do not change. (2012) *Proceedings of SWAT4LS 2012*, CEUR Workshop Proceedings 952

8. HENKEL R., LE NOVÈRE N., WOLKENHAUER O., WALTEMATH D. Considerations of graph-based concepts to manage of computational biology models and associated simulations. (2012) *Proceedings of INFORMATIK 2012*. [cit=3]

7. HASTINGS J., LE NOVÈRE N., CEUSTERS W., MULLIGAN K., SMITH B. Wanting what we don't want to want : Representing addiction in interoperable bio-ontologies. (2012) *Proceedings of the 3rd International Conference on Biomedical Ontology*, CEUR Workshop Proceedings 897.

6. D. KÖHN, N. LE NOVÈRE. SED-ML - An XML Format for the Implementation of the MIASE Guidelines. (2008) *Proceedings of the 6th conference on Computational Methods in Systems Biology*, M. Heiner and A.M. Uhrmacher eds, *Lecture Notes in Bioinformatics*, 5307 : 176-190. [cit=38]

5. N. LE NOVÈRE. Neurologic diseases : are systems approaches the way forward ? (2008) *Pharmacopsychiatry*, 41 : S28-S31. [cit=2]

4. N. LE NOVÈRE, M. COURTOT, C. LAIBE. Adding semantics in kinetics models of biochemical pathways. (2007) *Proceedings of the 2nd International Symposium on experimental standard conditions of enzyme characterizations*, available at `http://www.beilstein-institut.de/index.php?id=196` [cit=38]

3. N. LE NOVÈRE. Model storage, exchange and integration. (2006) *BMC Neuroscience*, 7 : S11. [cit=77]

2. J.A. HILL, N. LE NOVÈRE, M. ZOLI, J.-P. CHANGEUX. Ontogeny of nicotinic receptor subunit expression in cardiac autonomic circuit in rat. (1994) *Circulation* 90 part 2 : 360.

1. N. LE NOVÈRE, M. ZOLI, J.-P. CHANGEUX. Effets de l'injection de nicotine sur les ARNs messagers codant pour des protéines caractéristiques de la transmission dans le système dopaminergique mésostrié. (2004) *Semaine des hôpitaux de Paris* 70 : 403.

C.3 Chapitres, éditoriaux, publications dans des revues sans comité de lecture etc.

25. CHELLIAH V., LAIBE C., LE NOVÈRE N. BioModels Database : a repository of mathematical models of biological processes. (2013) *Encyclopedia of Systems Biology*, Springer (2013) ISBN 978-1-4419-9864-4

24. SCHREIBER F., LE NOVÈRE N. Exchange Formats for Systems Biology : SBGN. (2013) *Encyclopedia of Systems Biology*, Springer (2013) ISBN 978-1-4419-9864-4

23. KEATING S.M., LE NOVÈRE N. Supporting SBML as a Model Exchange Format in Software Applications. (2013) *Methods in Molecular Biology. In silico Systems Biology : A systems-based approach to understanding biological processes*, pp 201-225.

22. CHELLIAH V., LAIBE C., LE NOVÈRE N. BioModels Database : a repository of mathematical models of biological processes. (2013) *Methods in Molecular Biology. In silico Systems Biology : A systems-based approach to understanding biological processes*, pp 189-199. [cit=3]

21. JUTY, N., LAIBE C., LE NOVÈRE N. Controlled annotations for Systems Biology. (2013) *Methods in Molecular Biology. In silico Systems Biology : A systems-based approach to understanding biological processes*, pp 227-245.

20. LE NOVÈRE N., ENDLER L. Using chemical kinetics to model biochemical pathways. (2013) *Methods in Molecular Biology. In silico Systems Biology : A systems-based approach to understanding biological processes*, pp 147-167.

19. KEATING S., LE NOVÈRE N. Encoding neuronal models in SBML. *Computational Systems Neurobiology*, (2012) Le Novère ed, Springer ISBN-13 : 978-9400738577

18. ENDLER L., STEFAN M.I., EDELSTEIN S., LE NOVÈRE N. Using chemical kinetics to model neuronal signalling pathways.*Computational Systems Neurobiology*, (2012) Le Novère ed, Springer ISBN-13 : 978-9400738577

17. LE NOVÈRE N., HUCKA M., ANWAR N., BADER G., DEMIR E., MOODIE S., SOROKIN A. Meeting report from the first meetings of the Computational Modelling in Biology Network (COMBINE). (2011) *SIGS*, 5(2) :577. [cit=1]

16. M. HUCKA, N. LE NOVÈRE. Software that goes with the flow. (2010) *BMC Biology*, 8 :140. [cit=3]

15. N. LE NOVÈRE. Realistic models of neuron require quantitative information at the single-cell level. (2010) in *Unravelling Single Cell Genomics*, Bontoux, Dauphinot and Potier eds, pp 45-53.

14. B.F. BALDI, C. HOYER, N. LE NOVÈRE. Schizophrenic : Forever young ? (2010) *Genome Medecine*, 2 : 32. [cit=2]

13. V. CHELLIAH, L. ENDLER, N. JUTY, C. LAIBE, C. LI, N. RODRIGUEZ, N. LE NOVÈRE . Data Integration and Semantic Enrichment of Systems Biology Models and Simulations. (2009) *Lecture Notes in Bioinformatics*, 5647 : 5-15. [cit=5]

12. T.S. SCHIMIZU, N. LE NOVÈRE. Looking inside the box : bacterial transistor arrays. (2008) *Molecular Microbiology*, 69 : 5-9. [cit=1]

11. B.E. SHAPIRO, A. FINNEY, M. HUCKA, B. BORSTEIN, A. FUNAHASHI, A. JOURAKI, S.M. KEATING, N. LE NOVÈRE, J.A. MATTHEWS, M.J. SCHILSTRA. SBML Models and MathSBML. (2007) *Introduction to Systems Biology*, Humana Press, USA. ISBN-10 : 1588297063, ISBN-13 :978-1588297068.

10. N. LE NOVÈRE. The long journey to a Systems Biology of neuronal function. (2007) *BMC Systems Biology*, 1 : 28. [cit=17]

9. N. LE NOVÈRE. BioModels.net, tools and resources to support Computational Systems Biology. (2005) *Proceedings of the 4th Workshop on Computation of Biochemical Pathways and Genetic Networks*, Logos, Berlin, pp. 69-74. [cit=1]

8. N. LE NOVÈRE, D. TOLLE. Particle-based stochastic simulations. (2005) *Proceedings of the 4th Workshop on Computation of Biochemical Pathways and Genetic Networks*, Logos, Berlin, pp. 41-45.

7. N. LE NOVÈRE. La communication neuronale à l'ère post-génomique : le projet DopaNet. (2003) *Biofutur*, 237 : 20-24.

6. B. LACOMBE B., D. BECKER, R. HEDRICH, R. DESALLE, M. HOLLMANN, J. KWAK, J.I. SCHROEDER, N. LE NOVÈRE, H.G. NAM, E.P. SPALDING, M. TESTER, F.J. TURANO, J. CHIU, G. CORUZZI. On the identity of plant glutamate receptors. (2001) *Science*, 292 : 1486-1487. [cit=128]

5. U.C. ROGNER, D.D. SPYROPOULOS, N LE NOVÈRE, J.-P. CHANGEUX, P. AVNER. Mutation of the murine X-linked gene NAP1L2 by homologous recombination. (2000) *European Journal of Neuroscience*, 12 : S247.

4. M. CORDERO-ERAUSQUIN, L.M. MARUBIO, M.M.M. ARROYO-JIMENEZ, N. LE NOVÈRE, M. HUCHET, C. LÉNA, J.-P. CHANGEUX. Characterization of mice lacking the nicotinic receptor α4 subunit. (1998) *Journal of Physiology-Paris*, 92 : 423.

3. N. LE NOVÈRE, P.-J. CORRINGER, J.-P. CHANGEUX. Predicted secondary structure of a nicotinic acetylcholine receptor subunit. Incorporation of predicted solvent accessibility and experimental data into a two dimensional representation. (1998) *Journal of Physiology-Paris*, 92 : 458.

2. N. LE NOVÈRE. Thèse de doctorat de 3ème cycle : Contribution à l'étude de la relation structure-fonction dans la famille des sous-unités des récepteurs nicotiniques de l'acétylcholine. (1998)`http://www.ebi.ac.uk/~lenov/PUBLIS/THESELENOV/these.html`.

1. N. LE NOVÈRE. Approches théoriques et pratique en hybridation *in situ* et autoradiographie réceptorielle. (1998) *Annales de l'Institut Pasteur / actualités*, 9 : 259-270.

C.4 Rapports techniques

17. N. LE NOVÈRE, E. DEMIR, H. MI, S. MOODIE, A. VILLEGER Systems Biology Graphical Notation : Entity Relationship language Level 1 (Version 1.2) (2011) *Nature Precedings*, `doi:10.1038/npre.2011.5902.1`

16. D. WALTEMATH, F.T. BERGMANN, R. ADAMS, LE NOVÈRE N. Simulation Experiment Description Markup Language (SED-ML) : Level 1 Version 1. (2011) *Nature Precedings*, `doi:10.1038/npre.2011.5846.1`

15. S. MOODIE, N. LE NOVÈRE, E. DEMIR, H. MI, F. SCHREIBER Systems Biology Graphical Notation : Process Description language Level 1. (2011) *Nature Precedings*, `doi:10.1038/npre.2011.3721.3` [This is Level 1 Version 1.3] [cit=15]

14. D. WALTEMATH, N. SWAINSTON, A. LISTER, F. BERGMANN, R. HENKEL, S. HOOPS, M. HUCKA, N. JUTY, S. KEATING, C. KNÜPFER, F. KRAUSE, C. LAIBE, W. LIEBERMEISTER, C. LLOYD, G. MISIRLI, M. SCHULZ, M. TASCHUK, N. LE NOVÈRE SBML Level 3 Package Proposal : Annotation. (2011) *Nature Precedings*, `doi:10.1038/npre.2011.5610.1`

13. N. LE NOVÈRE, S. MOODIE, A. SOROKIN, F. SCHREIBER, H. MI Systems Biology Graphical Notation : Entity Relationship language Level 1. (2010) *Nature Precedings*, `doi:10.1038/npre.2010.3719.2` [This is Level 1 Version 1.1]

12. S. MOODIE, N. LE NOVÈRE, E. DEMIR, F. SCHREIBER, H. MI Systems Biology Graphical Notation : Process Description language Level 1. (2010) *Nature Precedings*, `doi:10101/npre.2010.3721.2` [This is Level 1 Version 1.2]

11. H. MI, F. SCHREIBER, N. LE NOVÈRE, S. MOODIE, A. SOROKIN Systems Biology Graphical Notation : Activity Flow language Level 1. (2009) *Nature Precedings*, doi:10.1038/npre.2009.3724.1 [cit=5]

10. N. LE NOVÈRE, S. MOODIE, A. SOROKIN, F. SCHREIBER, H. MI Systems Biology Graphical Notation : Entity Relationship language Level 1. (2009) *Nature Precedings*, doi:10.1038/npre.2009.3719.1

9. S. MOODIE, N. LE NOVÈRE, A. SOROKIN, F. SCHREIBER, H. MI. Systems Biology Graphical Notation : Process Description language Level 1. (2009) *Nature Precedings*, doi:10.1038/npre.2009.3721.1 [This is Level 1 Version 1.1]]

8. O. WOLKENHAUER, D. FELL, P. DE MEYTS, N. BLÜTHGEN, H. HERZEL, N. LE NOVÈRE, T. HÖFER, I. VAN LEEUWEN Advancing systems biology for medical applications. (2008) *ESF science policy briefing* 35.

7. M. HUCKA, S. HOOPS, S. KEATING, N. LE NOVÈRE, S. SAHLE, D.J. WILKINSON. Systems Biology Markup Language (SBML) Level 2 : Structures and Facilities for Model Definitions. (2008) *Nature Precedings*, doi:10.1038/npre.2008.2715.1 [This is Level 2 Version 4]

6. N. LE NOVÈRE, S. MOODIE, A. SOROKIN, M. HUCKA, F. SCHREIBER, E. DEMIR, H. MI, Y. MATSUOKA, K. WEGNER, H. KITANO. Systems Biology Graphical Notation : Process Diagram Level 1. (2008) *Nature Precedings*, doi:10101/npre.2008.2320.1. [cit=15]

5. N. LE NOVÈRE, A. OELLRICH. Systems Biology Markup Language (SBML) Level 3 Proposal : multistate components. (2007).

4. M. HUCKA, A. FINNEY, S. HOOPS, S. KEATING, N. LE NOVÈRE. Systems Biology Markup Language (SBML) Level 2 : Structures and Facilities for Model Definitions. (2007) *Nature Precedings*, doi:10101/npre.2007.58.1 [This is Level 2 Version 3]

3. M. HUCKA, A. FINNEY, N. LE NOVÈRE. Systems Biology Markup Language (SBML) Level 2 : Structures and Facilities for Model Definitions. (2006) http://sbml.org/documents/. [This is Level 2 Version 2] [cit=50]

2. N. LE NOVÈRE, A. FINNEY. A simple scheme for annotating SBML with references to controlled vocabularies and database entries. (2005).

1. N. LE NOVÈRE. Effets cellulaires de la nicotine. In : Tabac : comprendre la dépendance pour agir. (2004) INSERM Collective expertise.