



SBML

SBML Systems Biology
Markup Language

langage de modélisation des réseaux biochimiques

Modéliser un réseau biochimique consiste à décrire les objets biologiques de ce réseau et les fonctions dans lesquelles ils sont impliqués. L'utilisation d'un langage standard de description de modèles facilite leur réutilisation et l'échange d'informations entre les biologistes.

Marie Beurton-Aimar,
Laboratoire
bordelais de
recherche
en informatique
(LaBRI); UMR CNRS
5800-université
Bordeaux 1 ;
laboratoire
de physiopathologie
mitochondriale;
INSERM 688-
université
Bordeaux 2

Standards et modèles

› De l'éprouvette à l'ordinateur

L'ensemble des étapes qui conduisent à l'élaboration et la validation d'un modèle sont : le recueil des données, la mise en équation ou conception à proprement parlé du modèle, la simulation de ce modèle sur un ordinateur, l'analyse des résultats et la comparaison avec ceux obtenus lors des expériences.

Le recueil des données est sans nul doute l'étape dont le coût est le plus élevé, que ce soit directement au moment de la recherche de ces données ou dans les étapes de vérification et de stockage. Ceci explique le besoin de plus en plus évident de rentabilisation de ces données, particulièrement en les réutilisant. Les échanges de données au travers de réseaux comme Internet ont été mis en œuvre grâce à la mise à disposition de grandes bases de données. En effet, la biologie est un des domaines où l'explosion du nombre et de la taille des bases de données a été particulièrement spectaculaire ces dix dernières années du fait, principalement, d'outils d'analyse à grande échelle : séquençage de génomes, *microarrays*, etc. Pour stocker ces différentes informations, des bases de données dédiées ont été développées. On dispose à l'heure actuelle de plus de 900 bases de données en biologie moléculaire selon la revue scientifique *Nucleic Acids Research*¹ (<http://en.wikipedia.org/wiki/Biological_database>).

De nombreuses difficultés se posent lorsqu'il s'agit de réutiliser les informations stockées dans ces bases, par exemple, le fait que certaines informations sont manquantes. En effet, suivant le domaine couvert, métabolisme, génétique, etc., certaines informations ne sont pas mentionnées, soit parce qu'elles sont évidentes (c'est-à-dire implicites), soit parce qu'elles sont considérées comme non pertinentes pour le domaine traité. Parmi d'autres difficultés, on mentionnera l'absence de standardisation du vocabulaire biologique et le fait que malgré l'abondance des données, les connaissances sont (forcément) incomplètes. Pour aider les biologistes à organiser leur travail de recueil et de stockage des données, plusieurs démarches de standardisation ont été initiées, par exemple la description d'ontologies comme Gene Ontology (GO, voir chapitre 5) ou la définition de langages de structuration de données comme CellML (voir chapitre 11) ou SBML. Ce chapitre expose la démarche initiée par un consortium de modélisateurs en biochimie qui a eu pour résultat la réalisation du langage SBML.

› Qu'est ce qu'un modèle dans le contexte des réseaux biochimiques?

Le terme de réactions biochimiques est un terme générique qui désigne un ensemble d'interactions que l'on classe traditionnellement

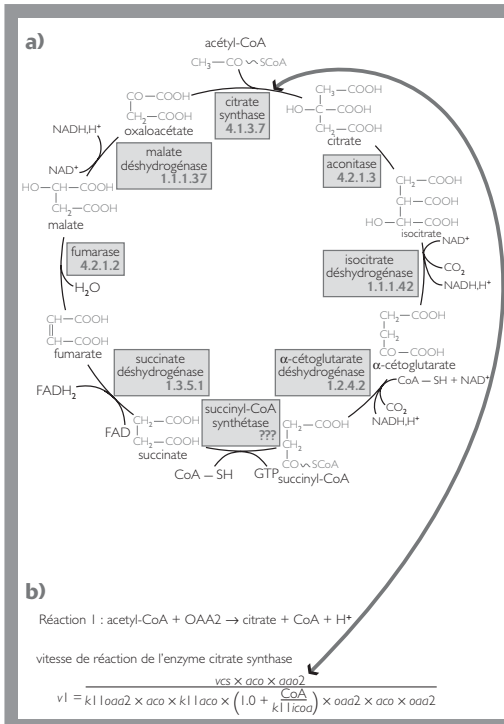
1. *Nucleic Acids Research*, 35, Database issue, 2007 (http://nar.oxfordjournals.org/content/vol35/suppl_1/index.dtl).

des réseaux biochimiques

7.3

en trois familles : les réseaux génétiques, les réseaux de signalisation et le métabolisme. Les chaînes de réactions biochimiques (ou *pathways*) forment un réseau dont le degré de connectivité est en général assez élevé. Une des caractéristiques intrinsèques d'un modèle de réseau biochimique est qu'il est la description d'un processus biologique et donc comporte des informations caractérisant les aspects dynamiques de ce réseau. Classiquement, cette dynamique sera traduite

Un exemple de réseau métabolique : le cycle de Krebs



Le cycle de Krebs, découvert en 1937 par le biochimiste du même nom, est une chaîne de réactions biochimiques qui fait suite à la glycolyse anaérobie. Localisé dans la mitochondrie, il fournit sous forme d'ATP la plus grande partie de l'énergie nécessaire à l'organisme. Le cycle de Krebs de la mitochondrie humaine est reproduit dans la figure 7.1 a : les 6 enzymes apparaissent dans les encadrés avec leur nom et leur numéro EC (classification internationale), tandis que les métabolites, substrats ou produits, sont précisés sur les arcs ; les réactions étant considérées comme réversibles, les arcs ne sont pas orientés. La double flèche (I) met en correspondance la réaction catalysée par la citrate synthétase avec son expression mathématique. Cette expression est utilisée pour illustrer les éléments de constructions du langage SBML.

Figure 7.1 (a) Représentation du cycle de Krebs². (b) Réaction de la citrate synthase du modèle de Wright³.

par une liste des équations mathématiques décrivant l'évolution des quantités de molécules dans un volume et un temps donnés (on parlera alors de concentrations). Ces quantités évoluent sous l'influence de transformations biochimiques catalysées par des enzymes, à partir d'un pool plus ou moins hétérogène de molécules (les substrats) ces réactions produisent un nouveau pool de molécules (les produits) résultat d'une chaîne réactionnelle parfois complexe. La figure 7.1 montre la chaîne réactionnelle du cycle de Krebs, constituée de

2. « Formal TCA cycle description based on elementary actions », P. Mazière *et al.*, *Journal of BioSciences*, 2007, 32 (1).

3. « Systems analysis of the tricarboxylic acid cycle in *Dictyostelium discoideum*. I. », B.E. Wright *et al.*, *Journal of Biological Chemistry*, 1992, 267, p. 3101-3105.

7.4

SBML, langage de modélisation

16 réactions. La conception et la capitalisation du modèle mathématique correspondant supposent, d'une part, de pouvoir référencer l'ensemble des entités biologiques participant à la chaîne réactionnelle et d'autre part de pouvoir stocker les expressions mathématiques et les informations quantitatives spécifiques à chaque réaction. Si la description des entités (liste de substrats, de produits ou d'enzymes catalysant la réaction) peut être réalisée de façon assez consensuelle pour tous les modélisateurs, la valeur des paramètres, les unités de calcul ou le système d'équations sont en général spécifiques du processus modélisé. L'utilisation d'un langage de description de modèles indépendant des outils de simulation favorise la séparation des descriptions des entités de la description des processus biologiques qui permet différents niveaux de réutilisation : réutilisation de la description des éléments de la chaîne réactionnelle ou réutilisation des paramètres et équations. On pourra ainsi construire son propre modèle avec par exemple, des conditions initiales spécifiques ou des hypothèses de fonctionnement différentes, soit réutiliser tout ou partie d'un ensemble d'équations pour le connecter à un autre modèle ou tout simplement reproduire la simulation. C'est ce que propose le langage SBML ; en prenant pour exemple une étape du cycle de Krebs (réaction 1), les éléments du langage sont présentés ci-après.

› Le langage SBML (System biology markup language)

SBML est un langage à balises, basé sur le langage XML (voir chapitre 3). En substance, un document XML est découpé en éléments structurés hiérarchiquement à partir d'un élément racine. Syntactiquement, les éléments d'un document XML sont marqués dans le document lui-même par des paires de balises (ou *tag*) ouvrantes et fermantes ; chaque élément est composé d'un nom qui spécifie son type, d'attributs et d'un contenu formé d'éléments ou de textes ; ce triplet ressource, propriétés et valeurs est nommé (sujet, prédicat, objet).

SBML est un ensemble d'éléments de constructions spécifiques à la biologie systémique définis dans un schéma XML. C'est en avril 2000, pendant un *workshop* qui se tenait à Caltech (Institut de technologie de Californie, États-unis) que Hamid Bolouri, Hiroaki Kitano et John C. Doyle amorcent ce projet ; ils sont très vite rejoints par de nombreux chercheurs de la communauté des modélisateurs⁴. SBML est en constante évolution, et les améliorations sont apportées au cours de réunions annuelles, les Hackathons SBML, qui réunissent développeurs et utilisateurs de SBML et dont la première manifesta-

La construction des noms de *tag* et d'attributs respecte une convention issue de la programmation orientée objet et plus particulièrement du langage de programmation Java. Les noms de type commencent par des majuscules et celui des variables par des minuscules mais sont construits à l'aide des noms de leur type qui conserve leur majuscule. Par exemple, le champ *listOfCompartments* est un container de *Compartments*, le mot *listOf* désignant par convention un « ensemble de ». Le respect de ces conventions permet une meilleure standardisation de l'écriture des fichiers et donc une lecture plus aisée pour les utilisateurs de ces fichiers.

4. « The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models », M. Hucka et al., *Bioinformatics*, 2003, 19 (4), p. 524-531.

des réseaux biochimiques

7.5

tion a été organisée en juillet 2003. Les principales versions de SBML sont appelées des niveaux et la version 2, niveau 2, qui fait actuellement référence, a été publiée en juin 2006. Un projet de la version 3 est actuellement disponible (http://sbml.org/wiki/SBML_Level_3_Efforts) sous la forme d'un document de travail soumis à la communauté scientifique pour avis.

» Le schéma général de SBML

Le langage SBML est décrit par un schéma XML (<http://sbml.org/xml-schemas/>) et contenu dans un fichier appelé *sbml.xsd*.

Ce fichier est découpé en éléments structurés hiérarchiquement qui constituent l'arbre syntaxique du langage sous forme d'un schéma XML. La figure 7.2 montre l'organisation générale des TAG SBML. Si l'on considère l'élément défini par le TAG `<xsd:complexType name="Sbml">`, on peut voir qu'il est une séquence d'éléments `<xsd:element name="model" type="Model"/>`, cet élément étant la racine de l'arborescence qui contient la description du réseau biochimique en tant que suite de réactions.

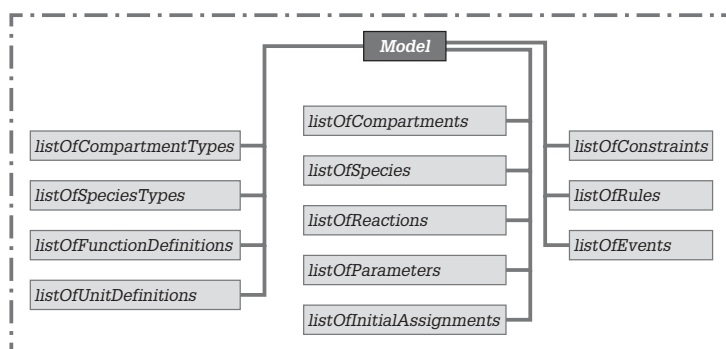


Figure 7.2
Organisation générale du langage SBML.
 On peut voir qu'un modèle est composé d'un ensemble de listes contenant les descriptions : des molécules, des réactions les impliquant, des paramètres, des conditions initiales, des unités standards, etc.

Un fichier SBML déclare deux espaces de noms, celui qui est propre à SBML et celui de MathML. Les directives sont détaillées à la figure 7.3.

Une description brève des éléments principaux et de leur utilisation fait l'objet des paragraphes suivants, non pas dans l'ordre du fichier *sbml.xsd* (figure 7.2), mais plutôt en suivant une logique d'écriture de modèle.

Compartment représente un espace clos dans lequel les espèces (*species*) sont localisées. Il ne correspond pas obligatoirement à une véritable structure dans la cellule (ou hors de celle-ci). Puisque toutes

Un espace de noms permet d'importer différents éléments ou contenus externes dans un même document XML tout en évitant des conflits de noms par la définition d'un espace de noms. Un espace de noms est identifié de façon unique par une URI (*Uniform Resource Locator*).

MathML est un langage basé sur XML permettant l'affichage de symboles mathématiques.

7·6

SBML, langage de modélisation

```

<xsd:schema
  targetNamespace="http://www.sbml.org/sbml/level2/version2"
  xmlns="http://www.sbml.org/sbml/level2/version2"
  xmlns:mml="http://www.w3.org/1998/Math/MathML"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"attributeFormDefault="unqualified"
  version="$Header:/cvsroot/sbml/specifications/sbmllevel2/version2/schema/sbml.xsd,v1.17
    2006/09/23 02:03:39 mhucka Exp $">
  <xsd:import
    namespace="http://www.w3.org/1998/Math/MathML"
    schemaLocation="http://www.w3.org/Math/XMLSchema/mathml2/mathml2.xsd"/>

```

^ Figure 7.3

Déclaration d'un espace de noms

La déclaration d'un espace de noms et de son préfixe associé consiste à insérer dans la balise ouvrante d'un élément contenant des noms (d'éléments ou d'attributs) issus de cet espace, l'attribut : « *xmlns:préfixe*="URI de l'espace de noms" ». L'espace de noms associé à XML Schema est : « *http://www.w3.org/2001/XMLSchema* » dont le préfixe est *xsd*.

les espèces définies dans un fichier SBML doivent être localisées, cela suppose qu'au moins un compartiment soit défini.

En dehors d'un identifiant unique qui est obligatoire, toutes les autres informations sont optionnelles. Les attributs : *spatialDimensions*, *constant*, *size* et *units* sont destinés à gérer les informations sur les dimensions du compartiment. Les différentes combinaisons de valeur de ces attributs autorisent ou non le redimensionnement et permettent d'interpréter les valeurs lorsque les quantités de *species* sont exprimées en concentration. Par défaut, le compartiment est déclaré en trois dimensions et de taille constante. *Size* et *units* n'ont pas de valeur par défaut. Ceci doit d'ailleurs faire l'objet de la plus grande attention lors d'un passage de modèle de la version 1 à la version 2 de SBML, car précédemment le volume (seul attribut existant pour gérer les dimensions) était considéré par défaut de valeur 1. Une initialisation spécifique peut être précisée par un *initialAssignment* au niveau du modèle et peut prendre la forme d'une formule mathématique. Il est obligatoire, dans ce cas, que le compartiment ne soit pas déclaré constant (*constant=false*).

La composition de compartiments est rendue possible par la désignation de l'identifiant (*Sid*) d'un autre compartiment comme valeur du champs *outside*. La figure 7.4 montre comment exprimer la localisation de la mitochondrie dans le cytosol.

des réseaux biochimiques

7.7

```
<lisOfCompartments>
  <Compartment id="cytosol"/>
  <Compartment id="mitochondria" outside="cytosol"/>
</lisOfCompartments>
```

◀ **Figure 7.4**
Déclaration de la mitochondrie dans le cytosol.

Species permet de spécifier les différentes entités du modèle quelle que soit leur nature depuis un ion simple jusqu'au complexe protéique. Un identifiant unique et le compartiment sont obligatoires. Comme nous l'avons vu, ce compartiment doit avoir été déclaré au niveau du modèle. Le type de l'espèce (*speciesType*) peut être précisé. La figure 7.5 correspond à la déclaration des substrats et produits de la réaction citrate synthase (réaction 4.1.3.7).

```
<listOfSpecies>
  <species id="OAA"           Compartment="mitochondria"/>
  <species id="acetyl-CoA"   Compartment="mitochondria"/>
  <species id="H2O"          Compartment="mitochondria"/>
  <species id="citrate"      Compartment="mitochondria"/>
  <species id="CoA"          Compartment="mitochondria"/>
</listOfSpecies>
```

◀ **Figure 7.5**
Déclaration des espèces moléculaires.

Reaction permet d'énumérer la liste des intervenants pour une réaction (figure 7.3). La *lisOfReactants* est constituée des molécules qui se trouvent en entrée de la réaction, la *listOfProducts* contient pour sa part les molécules produites par la réaction. Un attribut *ModifierSpeciesReferences* permet de préciser si d'autres molécules participent à la réaction sans être pour autant créés ou détruits, par exemple les inhibiteurs ou activateurs d'une réaction. Toutes ces molécules doivent avoir été préalablement décrites comme espèces. La stoechiométrie de la réaction est un attribut de chaque entité. La figure 7.6 décrit la réaction de citrate synthase.

Une réaction contient également les informations liées au processus biologique modélisé. Le champ *kineticLaw* référence une structure qui permet de coder l'équation mathématique au format *MathML* et de préciser une liste de paramètres, *listOfParameters*, locaux à cette réaction. Ces paramètres sont obligatoirement constants. Une *kineticLaw* peut également déclarer un *SBOTerm* (voir ci-après le paragraphe *SBOTerm*). Une réaction est considérée par défaut comme réversible (*champs reversible=true*). L'attribut *fastfield*

Figure 7.6 >
Déclaration des espèces
intervenant dans la réaction
de citrate synthase.

```
<Reaction id="R1" name="citrate synthase">
  <listOfReactants>
    <specieReference specie="OAA" stoichiometry="1"/>
    <specieReference specie="acetyl-CoA" stoichiometry="1"/>
    <specieReference specie="H2O" stoichiometry="1"/>
  </listOfReactants>
  <listOfProducts>
    <specieReference specie="citrate"/>
    <specieReference specie="CoA"/>
  </listOfProducts>
</Reaction>
```

(*false* par défaut) indique que cette équation doit être résolue à une échelle temporelle différente des autres.

2. « Formal TCA cycle description based on elementary actions », P. Mazière et al., *Journal of BioSciences*, 2007, 32 (1).

La figure 7.7 présente le code de l'équation liée à la réaction citrate synthase².

a) Équation de la réaction cytrate synthase

$$vcs \cdot ACO \cdot OAA / (k11oaa \cdot ACO + k11aco \cdot (1.0 + CoA / k11icoa) \cdot OAA + ACO \cdot OAA)$$

b) Traduction SBML

```
<kineticLaw>
  <math xmlns="http://www.w3.org/1998/Math/MathML">
    <apply><times/><ci>vcs</ci><ci>ACO</ci>
    <apply>
      <divide/><ci>OAA</ci>
    <apply>
      <plus/><apply><times/><ci>k11oaa</ci><ci>ACO</ci></apply>
      <apply><times/><ci>k11aco</ci>
      <apply>
        <plus/>1.0 <apply><divide/><ci>coa</ci><ci>k11icoa</ci></apply>
      </apply>
      <ci>oaa</ci>
    </apply>
    <apply><times/><ci>aco</ci><ci>oaa</ci></apply>
  </math>
  ...
```



```

...
<listOfParameters>
  <parameter id= "vcs" value="8.23"/>
  <parameter id= "k11aco" value="0.01"/>
  <parameter id= "k11oaa2" value="0.007"/>
  <parameter id= "k11icoa" value="0.11"/>
</listOfParameters>
</kineticLaw>

```

Parameter permet de déclarer les variables utilisées dans les différentes équations. Par défaut, leur valeur est constante pour une simulation donnée. Lorsqu'un paramètre est défini dans la liste des paramètres du modèle, il est considéré comme global au modèle; s'il est déclaré dans la *KineticLaw* d'une réaction, il est local à cette réaction (figure 7.4).

Rules est une alternative fournie par SBML. En effet, dans certains cas, l'évolution de la concentration d'une molécule ne peut-être décrite sous la forme d'une réaction SBML (le plus souvent parce que celle-ci n'est pas connue). SBML permet la description de règles (*Rules*) qui peuvent se décliner sous trois formes : *AlgebraicRules*, *AssignmentRules* et *RateRules*. Une règle possède un attribut *math* qui permet de stocker une équation au format MathML. Les *AssignmentRules* et *RateRules* déclarent un attribut dont la valeur est la variable concernée par l'équation.

Events est utilisé pour décrire des conditions déclenchant des modifications de comportements d'une variable. Il contient obligatoirement une expression booléenne, champ *trigger* (« déclenchement »), et la description de la réalisation de l'évènement dans le champ *eventAssignment*. Il est possible de préciser un délai d'application de cet évènement et une unité de temps. Par défaut ce délai est exprimé dans l'unité *time* définie dans SBML qui est équivalente à seconde. La description de ce qui se produit si cet évènement est déclenché (clause *trigger=true*) est donné par *EventAssignment* qui contient le nom de la variable concernée et l'équation mathématique à appliquer.

Il est possible de déclarer un ensemble de conditions sur le fonctionnement général du modèle. *Constraints* permet de déclarer un ensemble de conditions sur le fonctionnement général du modèle. Par exemple, la figure 7.8 exprime que la concentration de molécules OAA ne peut varier qu'entre 1 et 100.

Figure 7.7 ▼

(a) Réaction catalysée par la citrate synthase.
(b) Sa déclaration dans SBML.

Figure 7.8 >
Déclaration de contraintes
pour la concentration
de l'oxaloacétate (OAA).

```
<listOfConstraints>
  <constraint>
    <math xmlns="http://www.w3.org/1998/Math/MathML">
      <apply>
        <and/>
        <apply><gt/><cn> 1 </cn><ci> OAA </ci> </apply>
        <apply><lt/><cn> 100 </cn> <ci> OAA</ci> </apply>
      </apply>
    </math>
    <message>
      <p xmlns="http://www.w3.org/1999/xhtml">
        Species OAA is out of range. </p>
      </message>
    </constraint>
</listOfConstraints>
```

*SBO*Term fait référence à la *System biology ontology* (SBO) qui est développée séparément de SBML mais qui est prévue pour être utilisée suivant la même logique et la même structuration. Attacher un identifiant SBO à un composant, quel qu'il soit, permet de lui associer un terme unique défini dans l'ontologie. Par exemple, si la valeur du SBOTerm associée à une réaction est *SBO:0000049* cela signifie qu'il s'agit d'une *first-order irreversible mass-action kinetics* réaction, cette information pourra être réutilisée par tout programme ayant intégré la nomenclature SBO.

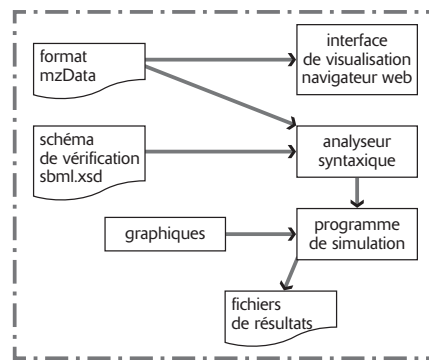
» Les modifications attendues dans la version 3 de SBML

La version 3 de SBML est en cours de développement et tout le monde peut y participer. Un forum de discussions est d'ailleurs ouvert au travers d'une page wiki (<http://sbml.org/wiki/>). La liste des points discutés contient entre autres : la définition d'un format pour stocker les modèles sous forme de graphiques, la composition spatiale entre les compartiments, la composition hiérarchique de modèles, la définition de structures dynamiques, la décomposition des espèces, la gestion des espaces de paramètres, etc. Tous ces points sont pris en charge de façon collaborative au sein de la communauté des modélisateurs.

» Les outils disponibles

La figure 7.9 montre l'organisation générale d'une application utilisant des fichiers de description SBML. Le modèle est contenu dans le

fichier *exemple.xml*. Un navigateur web peut charger ce fichier et l'afficher. Pour réaliser la simulation de ce modèle, un programme de simulation doit non seulement charger le fichier *exemple.xml* mais aussi la description du schéma SBML. L'analyse de ces deux fichiers est réalisée par un analyseur syntaxique qui fournit au simulateur les valeurs attendues dans le format correspondant à ce programme. Tout langage basé sur le principe des TAG peut être utilisé avec les analyseurs syntaxiques (*parser*) standards comme Xpath, XSLT, etc. Ces analyseurs permettent de sélectionner un ou plusieurs éléments d'un fichier de données XML au moyen de requêtes.



^ **Figure 7.9**
Schéma d'utilisation d'un modèle SBML par un navigateur web et un programme de simulation.

La visualisation d'un fichier de données est aussi possible au moyen d'un navigateur web : FireFox, Safari, Internet Explorer ou Mozilla. Il est à noter que la grande variété des langages de type XML rend le traitement « manuel » des fichiers réalisés extrêmement difficile. Une approche automatique pour la restitution formatée des données est donc indispensable. SBML est reconnu par un très grand nombre de logiciels de simulation ou d'affichage des réseaux. Le site web SBML affiche à ce jour plus d'une centaine de logiciels sachant reconnaître le langage. Réciproquement, il existe des outils de transformation des données depuis un format spécifique XML vers SBML. Par exemple, un fichier en KEGGML, le langage utilisé par la base de données KEGG, est convertible en SBML grâce à KEGGML2SBML. CellML2SBML fournit les mêmes fonctionnalités pour convertir du CellML (voir chapitre 11). Des outils comme la librairie libSBML permettent d'intégrer facilement la fonctionnalité de traduction et d'interrogation d'un fichier SBML à tout programme écrit en langage C ou C++. Cette librairie fournit aussi une interface vers Matlab et Python. Tous ces outils utilisent le schéma SBML pour effectuer leurs vérifications.

» Simulation de modèles

SBML étant fortement influencé par la communauté des modélisateurs en biochimie, des logiciels de simulation très populaires dans cette communauté comme Copasi (nouvelle version de Gepasi), Gillespie, StochSim, Jarnac ou FluxAnalyser peuvent relire directement un fichier SBML pour effectuer une simulation. Il en est de même pour les logiciels qui fournissent des représentations graphiques des réseaux : CellDesigner, JDesigner, Netbuilder, etc. Les références sur ces

différents outils sont présentées sur le site SBML (<<http://sbml.org>>). La plateforme Systems Biology Workbench (SBW, <<http://sbw.sourceforge.net/>>) propose un ensemble de services qui permet de faire communiquer des modèles écrits pour différents simulateurs, dans des environnements de programmation ou des systèmes d'exploitation différents. Basé sur la technologie client/serveur SBW permet ainsi de faire communiquer au travers du réseau des programmes de modélisation, de traitement de données, d'affichage graphique de résultats. Dans ce contexte, l'utilisation d'un langage standard de description des modèles est un outil indispensable pour assurer la bonne collaboration de ces différents programmes.

» La base de données BioModels

Le centre européen de recherche EBI (European Bioinformatics Institute) héberge la base de données BioModels (<<http://www.ebi.ac.uk/biomodels/>>). Les concepteurs de cette base désirent créer un espace accessible librement où les concepteurs de modèles peuvent déposer les fichiers relatifs à ces modèles. À ce jour, une centaine de modèles sont disponibles sur le site. Une interface web permet d'obtenir chaque description au format SBML, CellML et BioPAX mais aussi de télécharger directement les équations au format Scilab ou XPP. Le graphe des réactions et les courbes de simulation sont également disponibles. Ces fichiers sont déposés par les auteurs des modèles et sous leur responsabilité. Toutefois, BioModels propose une étape de vérification (*curation*) en plusieurs points. Une vérification du modèle par rapport à l'article qui le référence, une vérification appelée sémantique qui contrôle si l'utilisation des TAG est correcte, si certaines simplifications mathématiques ou utilisation de valeurs par défaut ne se contredisent pas et enfin un test de simulation pour vérifier que les résultats obtenus correspondent à l'article associé (seuls les modèles vérifiés sont fournis dans l'ensemble des formats, soit environ les deux tiers de la base).

Même si BioModels est fortement liée à SBML, les plus récentes versions du site montre une volonté d'ouvrir la base aux principaux formats standard.


» Discussion

La quasi-totalité des bases de données biologiques propose désormais une exportation dans un format de type XML, que ce soit un format propre ou un format plus généraliste. Ceci est rendu possible par la mise en correspondance des schémas respectifs, travail qui ne peut être effectué que par des spécialistes de ces langages. On peut regret-

des réseaux biochimiques

7.13

ter qu'il y ait peu de discussions autour du problème de la perte d'informations lorsque l'on passe d'un langage à un autre. Ainsi, que faire des équations mathématiques lors de la traduction d'un fichier SBML vers un fichier BioPAX? Comment convertir une composition de *pathways* BioPAX en SBML? Le plus souvent, ce sont les champs *comment* (« commentaires ») qui sont utilisés pour pallier les déficiences de l'un ou l'autre langage, mais cette façon de faire n'étant pas standardisée, il n'est pas possible de garantir le résultat lors d'une interprétation automatique. La question des valeurs par défaut qui ne sont

Figure 7.10 
 Page web de la base de données BioModels
 L'ensemble des champs SBML sont affichables séparément; une visualisation du résultat des simulations est proposée avec un affichage des graphiques.



pas les mêmes d'un logiciel de simulation à l'autre, est aussi un point d'achoppement qui peut gravement compromettre l'échange de modèles en les rendant impossible à re-simuler dans les mêmes conditions. Une intervention humaine est indispensable pour traiter au cas par cas ces questions. Toutefois, il est à noter que les langages de structuration connaissent un certain succès auprès des biologistes car ils offrent un début de réponse aux problèmes qui se posent.

Avec le développement des programmes de simulations, d'autres besoins ont vu le jour: Les simulations graphiques comme celles réalisées au moyen de systèmes multi-agents requièrent des informations sur la position (en trois dimensions) des molécules, la valeur des paramètres d'environnement, etc., qui ne peuvent être spécifiées actuellement par des langages comme SBML destiné principalement à la modélisation au moyen des équations différentielles.

Le concept de fonctions est extrêmement statique dans les descriptions qui sont faites en biologie, et cela explique qu'il existe de nombreuses nomenclatures pour les énumérer; mais encore peu d'outils de description consacrés aux aspects dynamiques en dehors des programmes de simulation.

► Autres projets de standardisation des réseaux biochimiques

» BioPAX

Le projet BioPAX (<<http://www.biopax.org/>>) est né en 2002 à l'occasion de l'atelier BioPathways à Edmonton au Canada lors de la conférence ISMB'02. Le groupe de travail BioPAX fut créé par Chris Hogue (BIND, UToronto), Peter Karp (BioCyc, SRI), et Chris Sander (Memorial Sloan-Kettering Cancer Center). Le premier objectif de ce groupe était de créer un format de description des données biochimiques; la première application envisagée traitait de la représentation des réseaux métaboliques. La version courante de BioPAX est le niveau 2 de la version 1. Le niveau 3 est en cours d'élaboration. BioPAX s'appuie sur le format OWL Web Ontology Language. Voici une description des éléments principaux de BioPax et de leur utilisation.

Un *pathway* au sens BioPAX est un réseau biologique est un ensemble de réactions ou interactions, quelle que soit leur nature: biochimique, transport, voire même historique ou bibliographique. C'est une boîte noire destinée à contenir un ensemble d'objets que l'on désire associer quelle qu'en soit la raison. Les éléments sont appelés des *pathway-components*. Ce modèle est totalement générique et applicable quel que soit le type de réseau à modéliser. On notera que,

des réseaux biochimiques

7.15

contrairement à SBML, BioPAX permet de déclarer la séquence des réactions, et ainsi d'ordonner la description.

Une *interaction* est un cadre générique à la description de chaque unité d'un processus biochimique. Une interaction physique peut se décliner en conversion (réaction biochimique, assemblage de complexe, transport) ou contrôle (catalyse, modulation). C'est à cette étape que l'on précise l'équation de réaction dans le cas de la conversion (attributs *left* et *right*), ou bien le contrôleur et le contrôlé (*controler* et *controled*) dans l'autre cas.

PhysicalEntity caractérise toute structure physique, c'est l'équivalent des termes *interactor* dans PSI-MI, *chemicals* dans BioCyc ou encore *object* dans BINDs. Il y a cinq sous-types de *PhysicalEntity* : *DNA*, *RNA*, *smallMolecule*, *Complex* et *Protein*. Pour décrire une interaction, on précisera un ou plusieurs *PhysicalEntityParticipant* dont l'un des attributs est de type *PhysicalEntity*. À cette étape, il sera alors possible de préciser le coefficient de stœchiométrie de la molécule et sa localisation cellulaire pour cette interaction.

Il existe de nombreuses bases de données biologiques qui proposent l'exportation de leurs données au format BioPAX. C'est le cas des bases généralistes comme KEGG ou BIND, des bases de la plateforme BioCyc et du projet Reactome. Le standard pour les bases de données en protéomique : PSI-MI est compatible avec BioPAX. Cell-Designer propose à la fois un convertisseur son propre format vers BioPAX et un convertisseur de BioPAX vers SBML. Enfin, Protégé qui est un outil de développement de bases de connaissances peut être utilisé pour travailler avec BioPAX ainsi que tous les outils développés autour du web sémantique et de OWL comme Jena. En revanche, contrairement à SBML, l'aspect simulation n'est pas traité dans BioPAX, les seules informations sur les processus biologiques sont la description des mécanismes grâce aux équations de réactions, les valeurs de stœchiométrie et les cinétiques. Il n'existe pas de *tag* qui permette la saisie d'équations mathématiques.

» Reactome

L'EBI héberge aussi un autre projet, Reactome (<http://www.reactome.org>), dont l'objectif principal est de mettre à disposition sur le web une base de données contenant la description de l'ensemble des processus biologiques humains. Si cette base utilise son propre modèle de données, elle exporte aussi à la demande dans le format SBML, BioPAX, Protege, SVG, etc. L'interface web est très orientée vers la représentation graphique interactive des réseaux.

