Systems Biology Markup Language (SBML) Level 1: Structures and Facilities for Basic Model Definitions

Michael Hucka, Andrew Finney, Herbert Sauro, Hamid Bolouri
{

1 Introduction

We present the Systems Biology Markup Language (SBML) Level 1

1.2 Scope and Limitations

SBML Level 1 is meant to support non-spatial biochemical models and the kinds of operations that are

The meaning of each component is as follows:

Unit definition

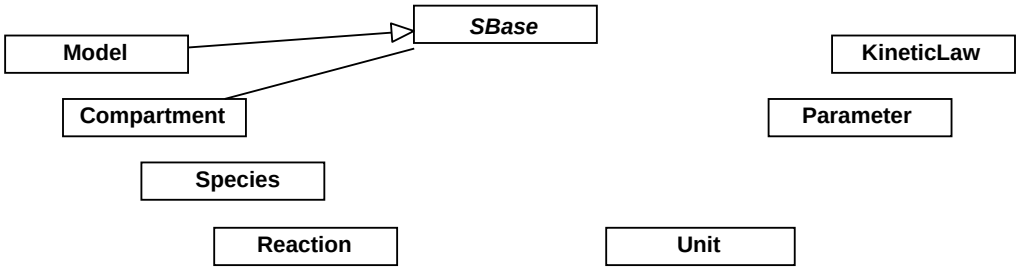
SBase
Model
KineticLaw
Compartment
Parameter
Species
Reaction
Unit

Figure 1:

reference is “http://www.mysim.org/ns” and the prefix is mysim. An example of an annotation might then be as follows:

```
...
```

```
letter   ::= 'a'..'z','A'..'Z'
digit    ::= '0'..'9'
name     ::= ( letter | '_' ) ( letter | digit | '_' )*
```

Figure 3: *The definition of the type SName, expressed in the variant of Extended Backus-Naur Form (EBNF) used by the XML 1.0 specification (Bray et al., 2000). The characters (*

The optional boolean field boundaryCondition determines whether the amount of the species is fixed or variable over the course of a simulation. The value of boundaryCondition defaults to "false", indicating

Figure 9: *The definition of `Rule` and derived types.*

4.6.1 `AlgebraicRule`

The rule type `AlgebraicRule` is used to express equations whose left-hand sides are zero. `AlgebraicRule` does not add any fields to the basic `Rule`; its role is simply to distinguish this case from the other cases.

4.6.2 `SpeciesConcentrationRule`

The `SpeciesConcentrationRule` structure adds one field, `species`, to the basic `AssignmentRule` type. The field `species` has type `SName` and is used to identify the species

4.7.2 KineticLaw

A kineticLaw structure describes the rate of the enclosing reaction. The use of a KineticLaw structure in a Reaction component is optional. (

The XML encoding shown above is quite straightforward. The outermost container is a tag, <smbl>, that identifies the contents as being Systems Biology Markup Language. The first attribute, xmlns, is required

```
                </listOfProducts>
                <kineticLaw formula="(vm * s1)/(km + s1)"/>
            </reaction>
            <reaction name="v2">
                <listOfReactants>
                    <speciesReference species="s1"/>
                </listOfReactants>
                <listOfProducts>
                    <speciesReference species="s2"/>
                </listOfProducts>
                <kineticLaw formula="(vm * s2)/(km + s2)"/>
            </reaction>
            <reaction name="v3">
                <listOfReactants>
                    <speciesReference species="s2"/>
                </listOfReactants>
                <listOfProducts>
                    <speciesReference species="x1"/>
                </listOfProducts>
                <kineticLaw formula="(vm * s1)/(km + s1)"/>
            </reaction>
        </listOfReactions>
    </model>
</sbml>
```

5.3 An Example of Using Rules

"x1"/241"219/908152dsp862es=[(5.3)]TJ-117.682-55.891cm00.compartme0.28k00.890.940.28KBTcompartme0.Td[(<)]TJ
98.399 K 1 0 0 1 0-15.207229.019 1. fomodel whi21 /F59 8.966 Tf 0.08e

This section contains a model which simulates a system containing a fast reaction. This model uses rules to express the mathematics of the fast reaction explicitly rather than using the implicit `fast` field on a reaction element. The system modeled is

$$X_0 \xrightarrow{k_1 X_0} S_1$$

$$S_1 \xrightarrow{k_f S_1 - k_r S_2} S_2$$

$$S_2 \xrightarrow{k_2 S_1} X_1$$

$$k_1 = 0.1, \quad k_2 = 0.15, \quad k_f = K_{eq} 10000, \quad k_r = 10000, \quad K_{eq} = 2.5.$$

```
<species id="T"  compartment="cell" initialAmount="0"/>
```

Appendix

A Summary of Notation

```
</xsd:annotation>
<xsd:restriction base="xsd:string">
  <xsd:pattern value="(_|[a-z]|[A-Z])sd:ssns|[0-950_|[*"/sd:str-ing">
  <xsd:str-ing">
```

```
  </xsd:restriction>
</xsd:simpleType>
<!--The definition of Unit follows.-->
<xsd:complexType name="Unit">
  <xsd:complexContent>
```

```
  </xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="Rule" abstract="true">
  <xsd:complexContent>
    <xsd:extension base="SBase">
     <xsd:attribute name="formula" type="xsd:string" use="required"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="AlgebraicRuleTJ-9.414-9.315Td[(</xsd:complexType>)]TJ0-9.315Td[(<xsd:complexType)-525(name
```

C Predefined Functions in SBML

Table 6 lists the basic mathematical functions that are define in SBML Level 1 at this time.

	Argument

Name	Arguments	Meaning	Formula
massi	S_i, k	Irreversible Mass Action Kinetics	$v = k$

Name	Arguments	Meaning	Formula
usii	S, V, K_m, K_i	Substrate Inhibition Kinetics (Irreversible)	$v = V \frac{S/K_m}{1 + S/K_m + S^2/K_i}$
usir	S, P, V_f, V_r, K_{ms}, K_{mp}, K_i	Substrate Inhibition Kinetics (Reversible)	$v = \frac{V_f S/K_{ms} + V_r P/K_{mp}}{1 + S/K_{ms} + P/K_{mp} + S^2/K_i}$
uai	S, V, K_{sa}, K_{sc}	Substrate Activation	$v = \frac{V\,(S/K_{sa})^2}{1 + S/K_{sc} + (S/K_{sa})^2 + S/K_{sa}}$
ucii	S, I, V, K_m, K_i	Competitive Inhibition (Irreversible)	$v = \frac{VS/K_m}{1 + S/K_m + I}$

Name	Arguments	Meaning	Formula
uctr	S, P, A_c, V_f, V_r, K_{ms}, K_{mp}, K_a	Catalytic Activation (Reversible)	$v = \frac{V_f S/K_{ms} - V_r P/K_{mp}}{1 + K_a/A_c + (S/K_{ms} + P/K_{mp})(1 + K_a/A_c)}$
umai	S, A_c, V, K_m, Kas, Kac	Mixed Activation Kinetics (Irreversible)	$v = VS/K$

Symbol	Meaning
α	E□ect of S and

References

Abbott, A. (1999). Alliance of US labs plans to build map of cell signalling pathways. *Nature*, 402:219–200.

Arkin, A. P. (2001). *Simulac* and *Deduce*. Available via the World Wide Web at `http://gobi.lbl.gov/~aparkin/Stuff/Software.html`.

Biron, P. V. and Malhotra, A. (2000). XML Schema part 2: Da,yes (W3C candidate recommendation 24 October 2000). Available via , World Wide Web at `http://www.w3.org/TR/xmlschema-2/`.

Bosak, J. and Bray, T. (1999). XML and the second-generation web. *Scientific American*, 280(5):89–93.

Hucka, M., Finney, A., Sauro, H. M., and Bolouri, H. (2001). Systems Biology Markup Language (SBML) Level 1: Structures and facilities for basic model definitions. Available via the World Wide Web at http://www.cds.caltech.edu/erato.

Kernighan, B. W. and Ritchie, D. M. (1988). *The C Programming Language*. Prentice-Hall, New Jersey: