libsbml Developer's Manual

Ben Bornstein

bornstei@cds.caltech.edu

The SBML Team
Control and Dynamical Systems, MC 107-81
California Institute of Technology, Pasadena, CA 91125, USA
http://www.sbml.org/

DRAFT

April 26, 2004

libsbml Version 2.0.3

This and other projects of the SBML Team are supported by the following organizations: the National Institutes of Health (USA); the International Joint Research Program of NEDO (Japan); the JST ERATO-SORST Program (Japan); the Japanese Ministry of Agriculture; the Japanese Ministry of Education, Culture, Sports, Science and Technology; the BBSRC e-Science Initiative (UK); the DARuoCon()1(uo)27(a(uo)1(i)1(o

Contents

1 Quick Start

libsbml requires a separate XML library for low-level XML tokenizing and Unicode support. It currently supports the Xerces-C++ and Expat XML libraries on Linux, Solaris, Windows and MacOS X. You will first need to ma1(g)-3 sure one of these libraries is installed on y(g)-our system. Many Linux systems provide one or both of these libraries either as part of their standard distribution or as an optional RPM or Debian package. For more information, see http://xml.apache.org/xerces-c/ for Xerces and http://expat.sf.net

```
./configure --with-xerces=/sw
```

During the installation phase (i.e., during `make install`, discussed below), the `libsbml` installation commands will copy header files to `/usr/local/include/sbml` and (shared and static) library files to `/usr/local/lib`, by default. To specify a different installation location, use the `--prefix` argument to `configure`. For example,

```
./configure --prefix=/my/favorite/path
```

Of course, you can combine the flags to `configure`, giving both `--prefix` and `--with-expat` or `--with-xerces`

3.3.2 Memory Tracing

In addition to the unit tests, a custom memory tracing facility is available. It is disabled by

```
typedef struct
{
  SBASE_FIELDS;
  char   *id;
  char   *name;
  char   *compartment;
  union
  {
    double Amount;
    double Concentration;
  } initial;
  char   *substanceUnits;
  char   *spatialSizeUnits;
  int     hasOnlySubstanceUnits;
  int     boundaryCondition;
  int     charge;
  int     constant;
} Species_t;
```

Figure 1: *Example: the definition of SBML's Species in UML (left) and the corresponding* `Species_t` *C struct (right) in* libsbml. `SBASE_FIELDS` *is part of the OOP-liOe style used to implement objects in C; it is a macro that expands into the fields defined by SBase. The use of a union for amount and concentration reflects that these two fields are mutually exclusive in the SBML Species definition.*

To instantiate (create) an object use either the `XXX_create()` or `XXX_createWith()` constructor. To destroy (free) an object use `XXX_free()`.

4.3 Accessing Fields

int Species_isSetCharge (const Species_t *s)

Returns `1` if the `charge` of this Species has been set, `0` otherwise.

4.4 Lists

The last item in the enumeration, `UNIT_KIND_INVALID`

```
  , SBML_ASSIGNMENT_RULE
  , SBML_RATE_RULE
  , SBML_SPECIES_CONCENTRATION_RULE
  , SBML_COMPARTMENT_VOLUME_RULE
  , SBML_PARAMETER_RULE
} SBMLTypeCode_t;
```

1

```
int writeSBML (SBMLDocument_
```

```
/**
 * The MathMLDocument
 */
typedef struct
{
  ASTNode_t *math;
} MathMLDocument_t;
```

The following are the functions deR014ned for the MathMLDocument class:

---

understands SBML Level 1 mathematical expeessions, or used as part of a translation system. The libsbml disteibution comes with an example progeam in the `examples` subdieectory called `translateMath` that implements an inteeactive command-line demonsteation of teanslating infix foemulas into MathML and vice-versa.

libsbml o□ees the ability to teanslate entire SBML Level 1 models to SBML Level 2, as explained

7 Levels of SBML

At the time of this writing, there exist 3 flavors of SBML: Level 1 Versions 1 and 2, and SBML

```
1 #include <stdio.h>
2
```

A Lists and `ListOf_t`

AST_PLUS	AST_FUNCTION_ARCCOTH	AST_FUNCTION_POWER
AST_MINUS	AST_FUNCTION_ARCCSC	AST_FUNCTION_ROOT
AST_TIMES	AST_FUNCTION_ARCCSCH	AST_FUNCTION_SEC
AST_DIVIDE	AST_FUNCTION_	

```
void ASTNode_
```

References

Ausbrooks, R., Buswell, S., Dalmas, S., Devitt, S., Diaz, A., Hunter, R., Smith, B., Soi□er, N., Sutor, R., and Watt, S. (2001). Mathematical markup language (MathML) version 2.0 (second edition) W3C recommendation 21 October 2003.

Bornstein, B. J. (2004). LibSBML API reference manual. Available on the Internet at http://www.sbml.org/software/libsbml.

Finney, A. M. and Hucka, M. (2003). Systems biology markup language: Level 2 and beyond. *Biochemical Society Transactions*, 31:1472–1473.

[illegible]