systems biology research group

Support for Flux Balance Constraints Models in JSBML

Nicolas Rodriguez[1,2], Alex Thomas[3], Michael Hucka[4], Nicolas Le Novère[1,2],
Bernhard Ø. Palsson[3], Andreas Dräger[3,5]

[1]European Bioinformatics Institute (EBI), Hinxton, UK, [2]Babraham Insitute, Babraham Research Campus, Cambridge, UK, [3]University of California, San Diego, La Jolla, CA, USA,
[4]California Institute of Technology, Pasadena, CA, USA, [5]Center for Bioinformatics Tuebingen (ZBIT), University of Tuebingen, Tübingen, Germany

ABSTRACT

SBML (a Systems Biology XML format) is the most widely used data format to encode biochemical network models. Many aspects required for COBRA modeling, however, are difficult to encode in this format. The recently proposed Flux Balance Constraints extension (FBC) adds many features of COBRA to the SBML standard. More features of COBRA are being added with the next release of FBC. In order to parse, write, and manipulate the data structures of SBML, an efficient programming library is prerequisite. Here we introduce JSBML, a comprehensive implementation of the SBML standard for the Java™ programing language as an alternative to libSBML. The JSBML team actively maintains and updates the project.

INTRODUCTION

- The XML-based format SBML is the *de facto* encoding standard for biochemical network models [1].
- The most recent specification of SBML (Level 3 Version 1) has introduced the possibility to extend the core language with specific packages in order to support more aspects of systems biology models.
- One particularly important package for dealing with constraint-based models is FBC (Flux Balance Constraints) [2]. This package is intended to standardize the content of COBRA-specific model extensions.
- For efficient creation, manipulation, parsing, and writing of SBML models, a specialized programming library is useful. The JSBML project has been developed with the aim to provide platform-independent open-source implementation of the latest SBML specification for Java [3].

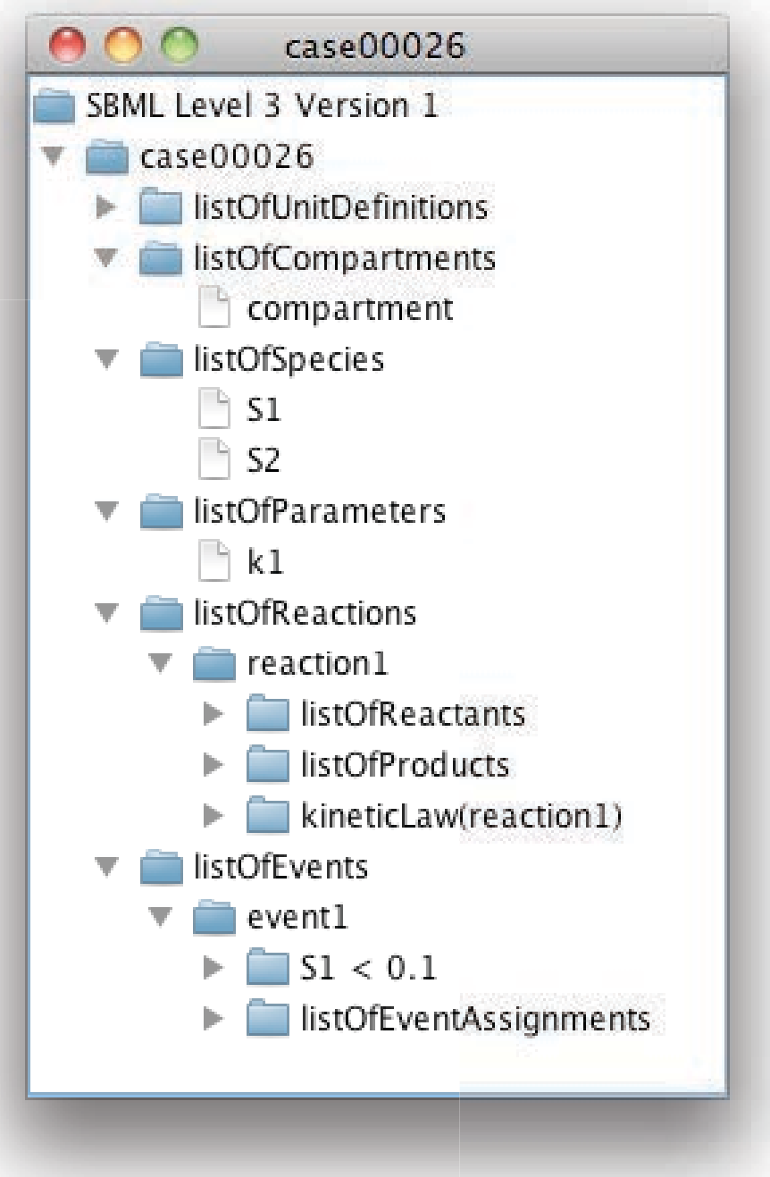
Java source code example to parse an SBML file and to display its content in a simple graphical user interface using JSBML:

```
import javax.swing.*;
import org.sbml.jsbml.*;

/** Displays the content of an SBML file in a {@link JTree} */
public class JSBMLvisualizer extends JFrame {

  public JSBMLvisualizer(SBMLDocument doc) {
  super(doc.getModel().getId());
  getContentPane().add(new JScrollPane(new JTree(doc)));
  setDefaultCloseOperation(EXIT_ON_CLOSE);
  pack();
  setVisible(true);
  }
  /** @param args Expects a valid path to an SBML file. */
  public static void main(String[] args) throws Exception {
  UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());
  new JSBMLvisualizer(SBMLReader.read(new java.io.File(args[0])));
  }

}
```

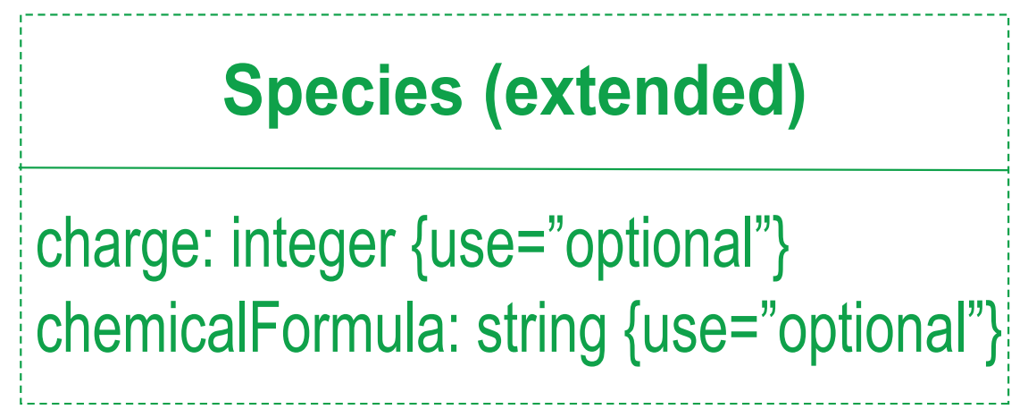


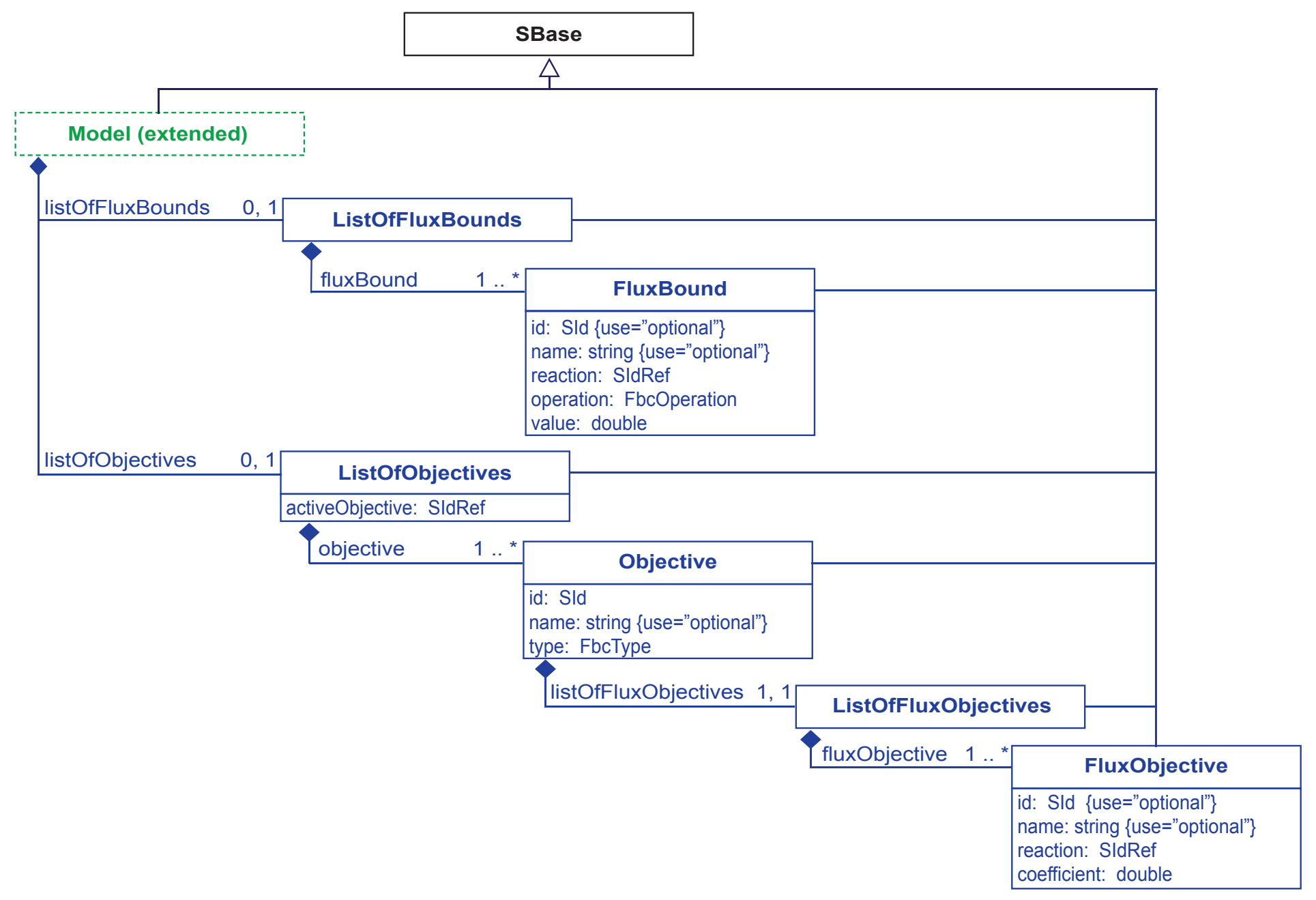
REFERENCES

[1] Hucka M, Finney A, Sauro HM et al. The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. Bioinformatics 19 (4): 524–31, 2003.
[2] Olivier B, Bergmann FT. SBML Level 3 Package: Flux Balance Constraints ('fbc'), 2013.
[3] Dräger A, Rodriguez N, Dumousseau M, Dörr A, Wrzodek C, Le Novère N, Zell A, Hucka M. JSBML: a flexible Java library for working with SBML. Bioinformatics, 27(15):2167-2168, June 2011.
[4] Bornstein BJ, Keating SM, Jouraku A, Hucka M. LibSBML: An API Library for SBML. Bioinformatics, 24(6):880–881, 2008.

FLUX BALANCE CONSTRAINTS

Main data structures of the FBC package: JSBML closely follows this specification.


Species (extended)
charge: integer {use="optional"}
chemicalFormula: string {use="optional"}



Additional elements added to the model: flux bounds and objective functions for flux optimization (maximization or minimization)
Additional properties can be stored in the species: charge and chemical formula.

BENEFITS OF JSBML

- Comprehensive open-soruce implementation of SBML
- Platform-independent, based on Java™ 1.6
- Support for latest SBML standards including extension packages
- Already used by several research groups in diverse software projects, e.g., a Jython-based version of COBRApy
- Well established and maintained project
- International community effort
- Usable as back-end for several modeling platforms in diverse programming environments
- JSBML development team involved in further development of FBC package for SBML
- Support for new developments and further COBRA features
- Next step: a standard for the representation of gene associations

ACKNOWLEDGMENTS

This work is funded by a Marie-Curie International Outgoing Fellowship (IOF) within the European Commision's 7th Framework Programme for Research and Technological Development (project "AMBiCon," grant number 332020) and a grant of the National Institute of Health for the continued development of essential SBML software support (NIH, United States, award number GM070923). Google supports this work as part of the Google Summer of Code 2014 initiative.

FBC EXAMPLE

Java source code example to create a fully valid SBML model including flux balance constraints with the objective to maximize the catalase reaction 2 H_2O_2 → O_2 + 2 H_2O:

```
import org.sbml.jsbml.*;
import org.sbml.jsbml.ext.fbc.*;

public class FBCTest {
  public static void main(String[] args) throws Exception {
    SBMLDocument doc = new SBMLDocument(3, 1);
    Model model = doc.createModel("COBRA_Model");
    FBCModelPlugin plugin = (FBCModelPlugin) model.getPlugin("fbc");
    String namespace = plugin.getElementNamespace();

    Compartment c1 = model.createCompartment("c1");
    c1.setConstant(true);
    Species s1 = model.createSpecies("s1", c1);
    Species s2 = model.createSpecies("s2", c1);
    Species s3 = model.createSpecies("s3", c1);
    for (Species s : model.getListOfSpecies()) {
      s.setConstant(false);
      s.setHasOnlySubstanceUnits(true);
      s.setBoundaryCondition(false);
      s.addExtension(namespace, new FBCSpeciesPlugin(s));
    }
    ((FBCSpeciesPlugin) s1.getExtension(namespace)).setChemicalFormula("H2O2");
    ((FBCSpeciesPlugin) s2.getExtension(namespace)).setChemicalFormula("O2");
    ((FBCSpeciesPlugin) s3.getExtension(namespace)).setChemicalFormula("H2O");

    Reaction r1 = model.createReaction("r1");
    r1.setName("catalase");
    r1.setReversible(true);
    r1.setFast(false);
    SpeciesReference sr1 = r1.createReactant(s1);
    sr1.setConstant(true);
    sr1.setStoichiometry(2d);
    SpeciesReference sr2 = r1.createProduct(s2);
    sr2.setConstant(true);
    sr2.setStoichiometry(1d);
    SpeciesReference sr3 = r1.createProduct(s3);
    sr3.setConstant(true);
    sr3.setStoichiometry(2d);

    FluxBound fb1 = plugin.createFluxBound("fb1");
    fb1.setReaction(r1);
    fb1.setOperation(FluxBound.Operation.GREATER_EQUAL);
    fb1.setValue(10d);

    Objective ob1 = plugin.createObjective("ob1");
    ob1.setType(Objective.Type.MAXIMIZE);
    FluxObjective fo1 = ob1.createFluxObjective("fo1");
    fo1.setReaction(r1);
    fo1.setCoefficient(1d);
    plugin.setActiveObjective(ob1.getId());

    SBMLWriter.write(doc, System.out, ' ', (short) 2);
  }
}
```

Resulting SBML code:

```
<?xml version='1.0' encoding='UTF-8' standalone='no'?>
<sbml xmlns="http://www.sbml.org/sbml/level3/version1/core" level="3" version="1" fbc:required="false"
  xmlns:fbc="http://www.sbml.org/sbml/level3/version1/fbc/version1">
  <model id="COBRA_Model">
    <fbc:listOfFluxBounds xmlns:fbc="http://www.sbml.org/sbml/level3/version1/fbc/version1">
      <fbc:fluxBound fbc:reaction="r1" fbc:value="10" fbc:operation="greaterEqual" fbc:id="fb1"/>
    </fbc:listOfFluxBounds>
    <fbc:listOfObjectives xmlns:fbc="http://www.sbml.org/sbml/level3/version1/fbc/version1" fbc:activeObjective="ob1">
      <fbc:objective fbc:type="maximize" fbc:id="ob1">
        <fbc:listOfFluxObjectives>
          <fbc:fluxObjective fbc:reaction="r1" fbc:coefficient="1" fbc:id="fo1"/>
        </fbc:listOfFluxObjectives>
      </fbc:objective>
    </fbc:listOfObjectives>
    <listOfCompartments>
      <compartment id="c1" constant="true"/>
    </listOfCompartments>
    <listOfSpecies>
      <species id="s1" constant="false" hasOnlySubstanceUnits="true" boundaryCondition="false" compartment="c1" fbc:chemicalFormula="H2O2"/>
      <species id="s2" constant="false" hasOnlySubstanceUnits="true" boundaryCondition="false" compartment="c1" fbc:chemicalFormula="O2"/>
      <species id="s3" constant="false" hasOnlySubstanceUnits="true" boundaryCondition="false" compartment="c1" fbc:chemicalFormula="H2O"/>
    </listOfSpecies>
    <listOfReactions>
      <reaction id="r1" name="catalase" reversible="true" fast="false">
        <listOfReactants>
          <speciesReference constant="true" species="s1" stoichiometry="2"/>
        </listOfReactants>
        <listOfProducts>
          <speciesReference constant="true" species="s2" stoichiometry="1"/>
          <speciesReference constant="true" species="s3" stoichiometry="2"/>
        </listOfProducts>
      </reaction>
    </listOfReactions>
  </model>
</sbml>
```