Proceedings
of the 4th International Modelica Conference,
Hamburg, March 7-8, 2005,
Gerhard Schmitz (editor)

E. Larsdotter Nilsson, P. Fritzson
Linköping University, Sweden

A Metabolic Specialization of a General Purpose Modelica Library for Biological and Biochemical Systems

pp. 85-93

Paper presented at the 4th International Modelica Conference, March 7-8, 2005,
Hamburg University of Technology, Hamburg-Harburg, Germany,
organized by The Modelica Association and the Department of Thermodynamics, Hamburg University of Technology

All papers of this conference can be downloaded from
http://www.Modelica.org/events/Conference2005/

A Metabolic Specialization of a General Purpose Modelica Library for Biological and Biochemical Systems

Emma Larsdotter Nilsson and Peter Fritzson
Linköpings universitet, PELAB – Programming Environment Laboratory
Department of Computer and Information Science, SE-581 83 Linköping, Sweden
{emmni,petfr}@ida.liu.se

Abstract

In the drug industry the later a substance is discharged from the drug development pipeline, the higher the financial cost. In order to reduce the number of lead compounds a number of systems have been suggested, and in most of these systems modeling and simulation of the lead compound's effects on different metabolic pathways are essential. In these systems, substances that are expected to be harmful or lethal can be removed at an early stage. Consequently, a reduced number of promising lead compounds can be chosen for the concluding tests.

Given Modelica's previous success with huge and complex systems it is likely that it will also be suitable for modeling, simulation, and visualization of metabolic pathway systems, i.e., those systems used in the drug industry. A Modelica library designed to be used for modeling, simulation, and visualization of metabolic pathways is the special-purpose library `Metabolic`, an extension of the abstract Modelica library `BioChem`.

KEYWORDS: Metabolic pathways, pathway modeling, pathway libraries, template models, `BioChem`, `Metabolic`.

1 Introduction

There is currently a great interest in the development of novel analytical technologies for rapid screening of biological dysfunctions in pharmaceutical and clinical applications. In the drug industry the later a substance is discharged from the drug development pipeline, the higher the financial cost. Not only is it costly to test many substances, the price of the tests increase along the development pipeline. Minimizing the number of substances that are fully tested, i.e., becoming lead compounds, is therefore one of the most important aims of all pharmaceutical discovery programs [1].

In order to reduce the number of lead compounds a number of systems have been suggested, out of which some have been realized [2-5]. In most of these systems modeling and simulation of the lead compound's effects on different metabolic pathways are included. A metabolic pathway can be seen a complex web made up out of several hundred substances and more than twice as many reactions. Substances that are expected to interact in a harmful or lethal way with essential metabolic pathways can be removed at an early stage and a reduced number of promising lead compounds can be chosen for the concluding tests.

In theory, simulations of a single or a few interconnected pathways can be useful when the metabolic pathways under study are relatively isolated from each other. In practice, even the simplest and most well-studied metabolic pathways can exhibit complex behavior due to connections in-between different levels of the whole-cell or whole-organism system.

In light of this, the need for a consistent framework for modeling, simulation, and visualization of metabolic pathways is quite obvious. The object-oriented approach for large scale systems has previously been proven successful in many areas and there is no reason to believe that it should not be useful for metabolic pathway systems.

Given Modelica's previous success with huge and complex technical, physical, electrical, and thermodynamic systems it is likely that it will also be suitable language for modeling, simulation, and visualization of metabolic pathway systems.

So far two Modelica libraries for biological and biochemical applications have been specified. The first library, `BioChem`, is an abstract general-purpose library for biological and biochemical systems. The `BioChem` library is not intended, nor designed to be used directly for creating models and running simulations. The intention with the library is to provide some common basic behaviors, attributes, and environmental properties to be used in special-purpose libraries.

The second library, `Metabolic`, is a special-purpose library extended from the partial models in `BioChem`. `Metabolic` is designed to be used for modeling, simulation, and visualization of metabolic pathways. The models specified in the library describe basic sub-

stances and general reactions that are common in metabolic pathways.

Provided with the reactions in `Metabolic` it is possible to build a library of metabolic pathway templates. The idea is that these general model-templates can easily be extended and adapted to concrete species-specific models. The concrete models can then be used in standalone and connected simulations of metabolic pathways.

1.1 Outline

Modelica has so far mainly been used to model technical, physical, electrical, and thermodynamic systems. Hence the area of biology and biochemistry might be somewhat unfamiliar to some of the readers. For those readers not familiar with some basic concepts and notions in biological and biochemical science this paper will first give an introduction to the area of research. The reader will be acquainted with the concept of seeing the cell as a system and the different levels within this system. A short overview of the data used for modeling and simulation of metabolic pathways is also given. The readers who are familiar to the information presented in the first part of the paper can skip to the fourth section where the work on using Modelica for modeling, simulation, and visualization of biological and biochemical systems is presented.

The second part of the paper starts with pointing out the most significant reasons to use Modelica for biological and biochemical systems, i.e., the benefits of performing modeling and simulation of such systems using Modelica. Subsequently the development of the two Modelica libraries, `BioChem` and `Metabolic` will be in focus, i.e., out-lining the basic design idea behind the two libraries and the environment that they have been developed in. From here on, the paper is concerned with the details of the two libraries and their use. The paper is concluded with some conclusion of the work done so far, and some future work and possible improvements.

2 Introduction to the Area of Research: The Cell as a System

During the past ten to fifteen years the development and introduction of new analytical techniques in the area of biology and biochemistry have greatly increased the amount of experimental data obtained from experiments performed in the area. Automated DNA sequencing, microarray-analysis of gene expressions, and protein profiling are just a few of the methods that have made a significant contribution to the extensive amount of data available. The obtained data can be useful in modeling, simulation, and visualization of cellular processes, addressing the whole chain of processes starting with DNA, on to the transcription of DNA into RNA, further on to the translation of RNA into proteins, and finally all the way to the end-concentrations of proteins.

2.1 Chemical Reactions

A chemical reaction involves one or more transformations of one or several substances, called substrates, resulting in one or several new substances, called products. A reaction can be either irreversible, meaning transforming substrate into product, or reversible, meaning not only transforming substrate into product but also the other way around. Strictly speaking, all reactions can be seen as reversible, but for irreversible reactions the re-transformation of substrate into product is essentially so small and/or slow that it is ignored. A reversible reaction can also be seen as two separate irreversible reactions.

Nature's struggle to reach balance is the driving force for all chemical reactions. The speed with which this balance is reached is highly dependent on the environment surrounding the substrates in question. A specific set of substrates, physical variables, and other substances present during the reaction should always result in the same reaction type, progress, and result as long as all the initial values and conditions are the same.

2.2 Reaction Networks

A number of sequential and/or parallel substance transformations can be arranged into a graph, with the edges representing the reactions and the nodes representing the substances. Depending on the reaction in focus most of the substances in a network can function both as substrates and products. Each reaction network will have in-flows and out-flow points, which in turn can be viewed as the substrates and products of a reaction network at a higher level. At this higher level, several of the more specialized reaction networks can be connected through these in-flows and out-flows to form a large super-network.

2.3 Metabolic Pathways in Cells

Cells are the basic building blocks of all living organisms. No matter if the cells are part of a multi-cellular organism, or constitute uni-cellular organisms, the processes inside them do not differ greatly. A cell's metabolism involves the uptake, decomposition, and rebuilding of different compounds and can be seen as several complex webs transporting matter and energy. These complex webs, made out of several hundred substances and more than twice as many reactions, are referred to as cellular or metabolic pathways e.g. the

Starch and sucrose metabolism, the Glycolysis, the Gluconeogenesis, and the Citrate cycle (Figure 1). Many of the reactions participating in these pathways are more or less the same in all cells, while others are highly dependent on the species, the type of cell, or even on the individual that the cell belongs to.

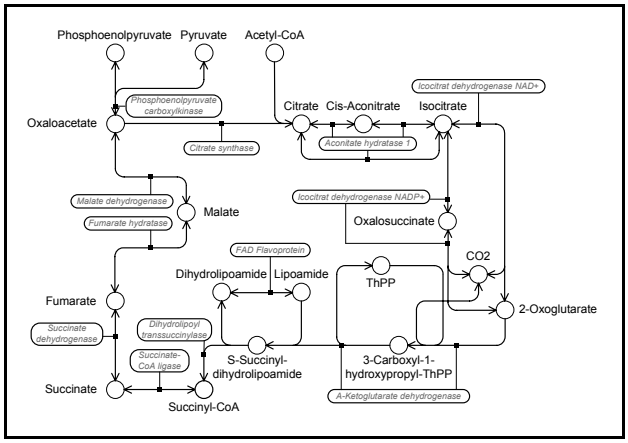

Figure 1. The metabolic pathway Citrate cycle for Bakers yeast (*Saccharomyces cerevisiae*). The enzymes that control the metabolic reactions are connected to the reaction arrows and shown in italic. The circles represent substances that participate in the pathway.

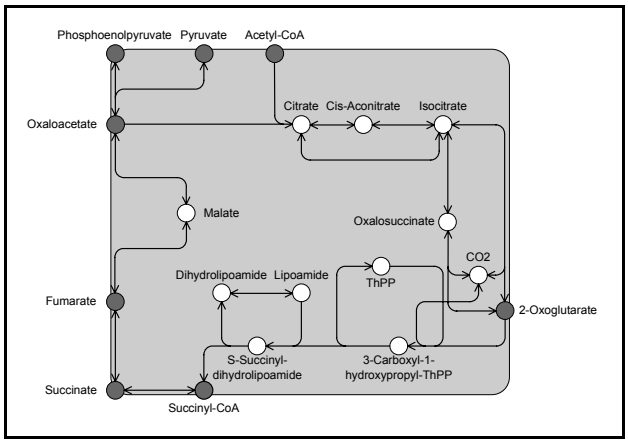

Figure 2. The metabolic pathway Citrate cycle for Bakers yeast (*Saccharomyces cerevisiae*) seen as a subsystem. The dark circles represent substances that are connection points to other metabolic pathways while the light circles represent substances that are internal with respect to the metabolic pathway. (Compare to Figure 1.)

Most of the reactions in these pathways are, in one way or another, controlled by enzymes, i.e., proteins. Proteins are the result of the transcription of DNA into RNA, and the subsequent translation of RNA into amino acid sequences. Enzymes (Figure 1) can either activate or inhibit the reaction in question and the amount of a protein in the cell is controlled by the expression of the gene that codes for that specific protein. One of the greatest challenges in the area right now is to figure out which proteins interact with which reactions and then try to find the corresponding coding gene in the DNA for these proteins.

Some of the reactions in these metabolic pathways are already well-known as well as mathematically defined. Other parts of these pathways are more or less undetermined, ranging from not being fully mathematically defined to not being fully discovered yet.

Each metabolic pathway is highly compartmentalized with a few in-flows and/or out-flows that can be connected to preceding and following metabolic pathways, e.g. the Starch and sucrose metabolism is a preceding pathway and the Citrate cycle is a following pathway of the Glycolysis while the Gluconeogenesis is both a preceding and following pathway of the Glycolysis (Figure 3).

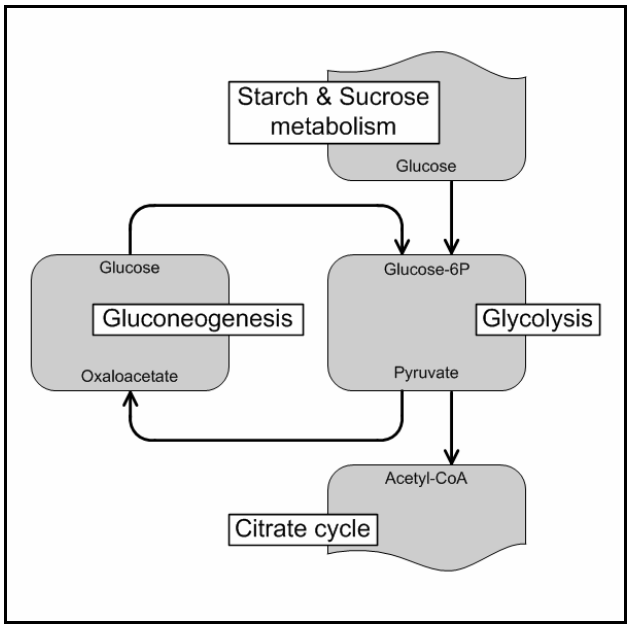

Figure 3. Interconnection of the four metabolic pathways, the Starch and sucrose metabolism, the Glycolysis, the Glucogenesis and the Citrate cycle. More pathways do connect to each one of the four pathways, but for simplicity these have been edited out.

2.4 Levels in the Whole-cell System and Multicellular Systems

The connection of all possible metabolic pathways for a cell will result in a fully functional system level in the whole-cell system, i.e., the metabolic level. But in order to understand and get a complete view of the entire whole-cell system one needs to look beyond the metabolic level. Apart from the metabolic level the whole-cell system also contains a gene-expression level. The latter level involves not only the transcription of DNA into RNA and the subsequent translation of the RNA into proteins, i.e., enzymes involved in metabolic reactions, but also all interactions in-between DNA, RNA, and proteins. Interactions in-between metabolites, i.e., substances taking part in the metabolic reactions, and DNA, RNA, and proteins are also considered to some extent at this level. Figure 4

provides a somewhat simplified view of the two levels of the whole-cell system.

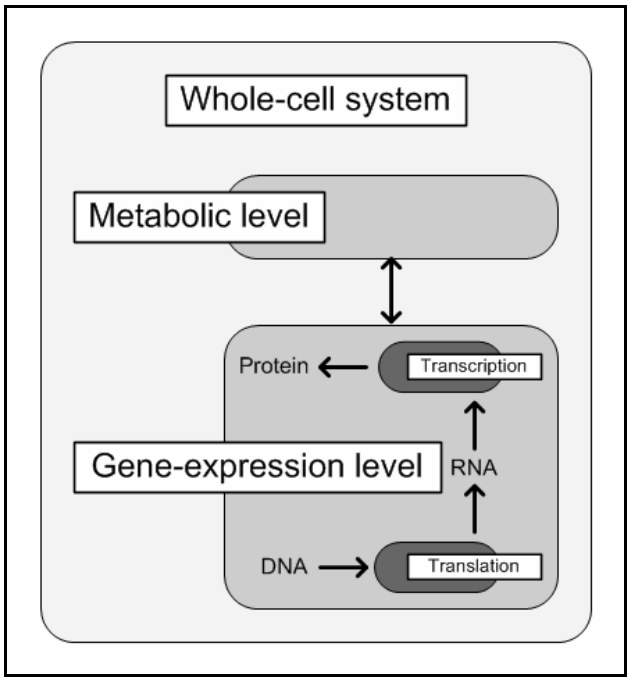

Figure 4. The whole-cell system with the gene-expression and the metabolic levels.

Beyond the whole-cell system there is a meta-level of different kinds of multi-cellular systems that all involves some kind of interchange of substances, and/or communication. Cellular specializations and/or differentiations are common in multi-cellular organisms and the assembly of them can be referred to as an organism level. Both uni-cellular and multi-cellular species can be part of large aggregated multi-species system, i.e., ecological systems.

3 Data Used for Modeling and Simulation of Metabolic pathways

Much of the data regarding metabolic pathways obtained through experiments and analysis is accessible in different public and commercial reference databases. In order to be able to model metabolic pathways one needs to know the participating substances and the reactions in-between them. The organization of entire blocks of metabolic pathways can be found in human-curated maps in public databases, i.e., KEGG [6] and BioCarta's "Proteomic Pathway Project" [7]. The equations specifying the reactions can, however not be found in those maps. This information can instead be retrieved from databases that provide data on individual enzymatic reactions, i.e., BRENDA [8] and EMP [9], and in databases that provide data on multi-step metabolic pathways, i.e., MPW [10] and EcoCyc/HumanCyc [11].

Although all the above resources together represent a good general reference in the work of modeling and simulation of metabolic pathways, they also have significant limitations. The usually non species-specific information causes many errors and inconsistencies, and in many cases the amount of data that can be found for a pathway is not enough for building accurate pathway models [12]. Yet another problem with these databases is that the data contained in different databases might be inconsistent. But even with the mentioned limitation it is still possible to perform modeling and simulation of metabolic pathways with the information provided by the above resources.

4 Benefits of Using Modelica for Biological and Biochemical Systems

Biological and biochemical systems can often easily be described using mathematical relations and expressions. This makes the equation-based Modelica [13] a suitable programming and modeling language for modeling of such systems. First of all, Modelica classes are acausal, i.e., can adapt to more than one data flow context [14], which is a great benefit when dealing with chemical reactions where the flow of matter can move in two directions.

The complexity of biological and biochemical models can be rather high, containing several hundreds of items. However, this will not be a problem since Modelica's strength as a modeling language for complex technical systems is well proven [15].

Moreover, Modelica's strong software component model also makes it ideal as an architectural description language for complex systems [15], e.g. metabolic pathway webs. It is also possible to model both discrete and continuous systems, as well as hybrids thereof [14]. Especially hybrid systems are quite common in the subject area of biology and biochemistry.

Finally, since the complexity of the biological and biochemical models can be rather high. Since Modelica is an object-oriented language the realization of the several hundreds of items within a metabolic pathway will be greatly facilitated through instantiating only a few basic components.

5 Development of the Libraries

5.1 Development Environment

The `BioChem` and `Metabolic` libraries have been developed using the MathModelica [16, 17] environment that consist of the Dymola kernel [18], the Mathematica notebook environment [19], and the graphical Model Editor.

In the MathModelica environment the Modelica code along with the documentation for each library is integrated in Mathematica notebooks. This does not only make it easier for non-computer science users to navigate the code, it also facilitates for these users to write their own Modelica classes. The Model Editor is a graphical drag-and-drop interface currently based on Microsoft Visio [20]. The user creates models in the graphical environment by dragging and dropping components from existing model libraries onto the diagram area and then connecting them in a suitable manner. Models can also be created in the Mathematica notebook textual environment, but the models must then first be transferred to the Model Editor in order to get a graphical view of the model.

Once a model has been created it can either be transferred to a notebook for further processing and documentation or simulated in the simulation environment provided by MathModelica. The Dymola kernel handles the simulations by receiving, compiling, and executing the model. The result from the simulation can then be presented with different types of diagram. The parameters and the initial values of the model can also be altered in-between simulations.

5.2 Basic Idea of Library Design

The design idea behind the `BioChem` library is to create a general purpose Modelica library for modeling and simulation of biological and biochemical systems (Figure 5).

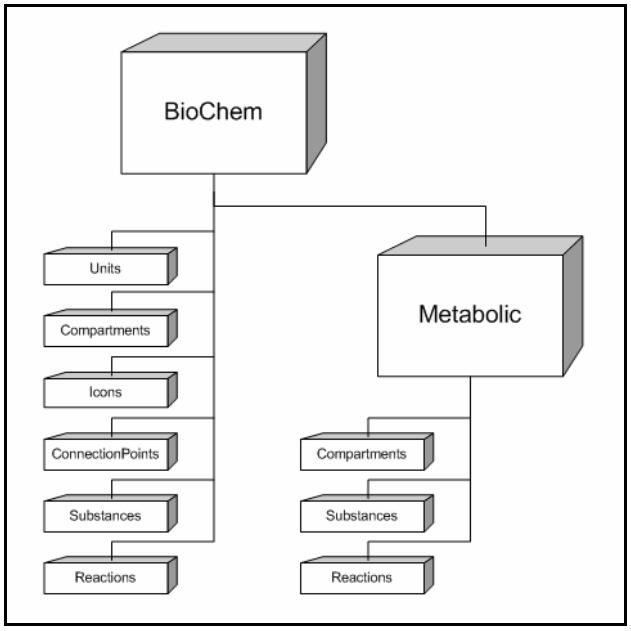

Figure 5. Simplified view of the structure of the `BioChem` and the `Metabolic` libraries.

The `BioChem` library is not intended, nor designed to be used directly for creating models and running simulations, but rather to provide some common basic behaviors, attributes, and environmental properties to be used in special-purpose biological and biochemical libraries. With the basic features provided in `BioChem` it is easy to create new special-purpose libraries without extensive addition of new code.

So far the Modelica library `Metabolic` is the only library to use the features provided by `BioChem` (Figure 5). The design idea behind `Metabolic` was to create a special-purpose Modelica library for modeling, simulation, and visualization of metabolic pathways, i.e., modeling, simulation, and visualization of the metabolic level in cells. The classes implemented in `Metabolic` describe substances and reactions that can take place in-between these substances in a diverse number of metabolic pathways.

5.3 The `BioChem` library

Most substances and reactions, respectively, have some common basic features. For instance, all substances must have a concentration and all reactions must have at least one substrate and one product. The design objective behind the `BioChem` library is to collect these basic features of substances and reactions along with units, compartment properties, and other attributes that are commonly used in these kinds of systems in a general-purpose biological and biochemical Modelica library.

```
package BioChem
  package Units
    "Units used in sub-packages of BioChem"
  end Units;
  package CompartmentProperties
    "Properties for compartments used in sub-libraries"
  end CompartmentProperties;
  package Icons
    "Icons used in the package"
  end Icons;
  package ConnectionPoints
    "Connector interfaces used in sub-libraries"
  end ConnectionPoints;
  package ReactionNodes
    "Reaction nodes"
    package Basics
      "Basic components for reaction nodes in the package"
    end Basics;
    package Substances
      "Partial models for substances in sub-libraries"
    end Substances;
  end ReactionNodes;
  package Reactions
    "Reaction edges"
    package Basics
      "Basic compnents for reaction in the package "
    end Basics;
    package ReactionTypes
      "Reaction types for reactions in sub-libraries"
    end ReactionTypes;
  end Reactions;
  package Metabolic
    "Package for metabolic cellular reactions"
  end Metabolic;
end BioChem;
```

Figure 6. Structure of the `BioChem` library.

In order to avoid recreating model code for the basic features of substances and reactions for each new Modelica library for biological or biochemical systems these features can instead be collected in one library. Along with substances and reactions it is also practical to define a default environmental container in which the substances are contained and where the reactions can occur. From the visualization's point of view it is also practical to define some default interfaces and

icons which later might be replaced in each sub-library. Not only the icons and interfaces are designed to be easily changed and/or replaced, most of the classes in BioChem are designed in such a way that they easily can be extended, and some parameters can also be replaced. The structure of the package is shown in Figure 6.

Due to the design of BioChem some restrictions on the types of systems that BioChem can be used for arise. The systems that the classes in BioChem can be used for are only those biological and biochemical systems that contain chemical reactions. Only for those systems fully functional models that can be used for simulation can be specified.

```
within BioChem;
package Metabolic
  "Package for metabolic cellular reactions"
  package Units
    "Units used in the package"
  end Units;
  package Compartments
    "Different types of compartments used in the package"
  end Compartments;
  package Substances
    "Reaction nodes"
  end Substances;
  package Reactions
    "Reaction edges"
    package Kinetics
      "Kinetic reactions"
      package UniUni
        "A->B kinetic reactions"
      end UniUni;
      package UniBi
        "A->B+C kinetic reactions"
      end UniBi;
      package UniTri
        "A->B+C+D kinetic reactions"
      end UniTri;
      package BiUni
        "A+B->C kinetic reactions"
      end BiUni;
      package BiBi
        "A+B->C+D kinetic reactions"
      end BiBi;
      package BiTri
        "A+B->C+D+E kinetic reactions"
      end BiTri;
      package TriUni
        "A+B+C->D kinetic reactions"
      end TriUni;
      package TriBi
        "A+B+C->D+E kinetic reactions"
      end TriBi;
      package TriTri
        "A+B+C->D+E+F kinetic reactions"
      end TriTri;
    end Kinetics;
    package SBML
      "Reactions pre-defined in SBML"
      package MichaelisMenten
        "Michaelis-Menten kinetics reactions"
      end MichaelisMenten;
      package Hill
        "Hill kinetics reactions"
      end Hill;
      package Activation
        "Activation kinetics reactions"
      end Activation;
      package Inhibition
        "Inhibition kinetics reactions"
      end Inhibition;
      package Modifier
        "Modifier kinetics reactions"
      end Modifier;
      package Misc
        "Miscellaneous SBML-defined reactions"
      end Misc;
    end SBML;
  end Reactions;
end Metabolic;
```

Figure 7. Structure of the Metabolic library.

5.4 The Metabolic library

Most classes in the Metabolic library extend one or more classes in the BioChem library. Generally the partial models specified in BioChem are extended, and with only a few additions, turned into fully functional models. The structure of the Metabolic package is shown in Figure 7.

As mentioned earlier, many of the reactions that occur in metabolic pathways are more or less the same in all cells no matter what species one look at. This is utilized in Metabolic to create a collection of partial models of different metabolic pathways that through small changes and/or additions are turned into fully functional species-specific metabolic pathways.

6 BioChem Sub-packages

Since the design objective for BioChem was to provide properties and attributes that are common in biological and biochemical systems the library contains several packages holding classes and partial models. The classes can be used as they are in sub-libraries to BioChem, while the partial models must be further extended to fully functional models.

6.1 BioChem.Units

A number of physical types are needed in order to be able to declare most parameters and variables in the BioChem package. Some of the types can be found in Modelica.SIunits and are here re-defined in order to avoid long name paths. The SI-types used in BioChem are volume (m^{-3}), amount of substance (mol), and concentration (mol m^{-3}).

Most of the other types in the package are non-SI types and thus need to be fully declared. In order for a reaction to actually transport something it has to have a flow of some kind. For a chemical reaction this flow is the volumetric reaction rate (mol m^{-3} s^{-1}). Together with the concentration, the molar flow rate of a substance (mol s^{-1}) is used in the interfaces between connected components.

6.2 BioChem.Compartments

In order to be able to control the environment of the reaction during a simulation a chemical reaction must take place in a restricted screened-off container. Within this container the basic physical properties, e.g. volume and temperature, are the same for all reactions that take place and all substances contained in that container.

In BioChem.Compartments this is solved using the inner-outer construct, i.e., a global variable. An inner volume is declared in the partial compartment model, giving all objects placed within an extension of the partial model the same surrounding volume. The objects that need to have knowledge of the global volume can use the declaration of an outer volume to reach it. The package so far only contains partial mod-

els for some different types of containers that can be found in cells.

6.3 BioChem.Icons

The package `BioChem.Icons` contains icons used in the drag-and-drop interface of the Model Editor in MathModelica. A substance is represented by a circle and the fill color is changed depending on the type of substance represented, i.e., substance in solution, fixed concentration, gaseous substance, etc. Since the substance only come in a few flavors there is one icon for each type of node.

The reactions on the other hand come in many different variations. A reaction is represented by an arrow with two or more ends. The number of ends an arrow can have is determined by the numbers of substrates and products that are involved in the reaction. Substrate-ends are, by convention, on the left side of the arrow, while product-ends are on the right side. Arrowheads indicate the direction of the reaction, i.e., irreversible reactions only have heads on the product-ends while reversible reactions have heads on both ends.

Instead of creating one icon for each type of reaction the final graphical interface for a reaction is built out of several partial icons. The reaction arrow is divided into three parts, substrates side (left part of the reaction arrow), middle, and products side (right part of the reaction arrow). The middle is the same for all reactions, while the two other parts differentiate depending on the number of substrates and products participating in the reaction.

Enzymes can affect reactions, which is represented by a small arrow and an enzyme sign. The sign represent the type of effect that the enzyme have on the reaction, i.e., inhibition, activation, or a combination of both, and are indicated with a **—**, **+**, and **M** respectively.

6.4 BioChem.ConnectionPoints

The package `BioChem.ConnectionPoints` contains the connector `SubstanceConnector` (Figure 8) that is used when connecting the different components in a model. In order to be able to make simulations using a connected model, the connector has to have a flow variable. For chemical reactions this flow variable is the molar flow rate of a substance (mol s^{-1}). There is also a non-flow variable in the connector, the concentration of a substance. The concentration is later on used in equations with relations to the reaction rate in reaction models.

The connector is used in several partial models in `BioChem.ConnectionPoints`. Each partial model relate to the graphical interface of at least on icon in `BioChem.Icons` (Not more than one icon at a time though.). For the reaction arrows, connectors are placed at each intended connectable end. For the enzymes regulating the reactions the connectors are placed at the enzyme signs. Finally for substances, eight connectors are placed on the rim of the circle that represents the node of substance.

```
within BioChem.ConnectionPoints;
connector SubstanceConnector
  "Connection point for substance transfer"
  extends Icons.SubstanceConnector;
  Units.Concentration c
    "Concentration of substance at the connection";
  flow Units.MolarFlowRate r
    "Molar flow rate of substance at the connection";
end SubstanceConnector;
```

Figure 8. `SubstanceConnector`, the connector used in `BioChem` and later on also in `Metabolic`.

6.5 BioChem.Substances

The package `BioChem.Substances` contains partial models of different kinds of nodes needed to represent substances in biological and biochemical systems. The basic attributes corresponding to the properties that are studied during simulations, i.e., the amount and the concentration of the substance, are declared in these partial models. All partial substance models also extend the partial model `BioChem.ConnectionPoints.Node`, which contains the connector interface.

6.6 BioChem.Reactions.Basics

All reactions need some basic components in order to work properly. In the package `BioChem.Reactions.Basics` these basic components are collected in a partial reaction model, `Reaction`.

`BioChem.Reactions.Basics` also contains components that are not needed in all types of reactions, but can rather be seen as roles assigned in some reactions while left vacant in others. Using the role-approach, the directions of a reaction can be seen as two roles. The role for a forward directed reaction is almost always appointed, while the role for a backward directed reaction only is assigned for reversible reactions.

The different types of enzymes that can affect a reaction can also be seen as a set of roles. When no enzymes affect the reaction, all enzyme roles are vacant. The different roles that are possible to assign are activator, inhibitor, and modifier. A modifier is a situation dependent enzyme that can react as either an inhibitor or an activator, depending on the environmental context. These roles are also directional, i.e., they can be appointed in both a forward and a backward context.

In `BioChem.Reactions.Basics` model for all the above roles are defined.

6.7 BioChem.Reactions.ReactionTypes

BioChem.Reactions.ReactionTypes contains a collection of partial models for different types of reactions that can take place in biological and biochemical systems. The reaction types are obtained by combining some different types of classes from other packages in BioChem. First, there is the combination of substrates and products. Then there is the appointment of the two reaction-direction roles. Finally, there is the possibility to appoint an enzyme role. At this point only three substrates and three products are allowed and only one of the enzyme roles can be appointed at a time.

Given the above restrictions four irreversible and seven reversible reaction types for each possible combination of substrates and products are generated, giveing 99 different reaction types to choose from in the sub-libraries.

Parts of the graphical interface for the MathModelica Model Editor are also defined in this package. Each partial model has a graphical representation in the form of a reaction arrow. If the role for the backward directed reaction is appointed, all the arrow-ends have heads, otherwise only the product-ends have heads. A small arrow perpendicular to the reaction arrow is used to indicate that there is an enzyme-role assigned in the reaction. An enzyme-arrow above the reaction arrow indicate that the enzyme is involved in the transformation of substrate into product, while an enzyme arrow below the reaction arrow indicate that the enzyme is involved in the reverse transformation.

Along with the graphical interface the partial models for connector interfaces in BioChem.ConnectionPoints are also extended. Since each of the connector interfaces have been defined in relation to an icon the extensions are quite straightforward.

7 Metabolic Sub-packages

The Metabolic library consists of several subpackages containing fully functional classes that can be used for building models and running simulations of metabolic systems.

7.1 Metabolic.Compartments

The Metabolic.Compartments package contains models for some of the different types of containers that can be found in cells when dealing with modeling and simulation of metabolic pathways. The partial compartment models in BioChem.Compartments are extended in order to obtain the basic properties of a compartment.

In order to be able to run a simulation of a model all substances, reactions, and other constructs in the model must be placed within a compartment model. Otherwise the global volume cannot be reached with the outer-declaration.

Reactions and substances that require different properties than the ones provided by the main-compartment can be placed in new compartments within or adjacent to the main-compartment.

7.2 Metabolic.Substances

The package Metabolic.Substances contains different types of nodes needed for representing a substance in a metabolic pathway. The substance models are specified by extending the partial models of substance nodes in BioChem.Substances and adding some additional attributes and equations. Thus both normal substance nodes and nodes with different types of restrictions, e.g. on the concentration of the substance, can be specified.

Typically the concentration in a substance node is allowed to change without restrictions during a simulation, while the total amount of substance in the node is conserved at all times. Some of the models have an assert statement that checks that the concentration never drop more than the tolerance below zero than. The tolerance is a parameter and can thus be changed for every node in a model as well as for each simulation run.

7.3 Metabolic.Reactions

Metabolic.Reactions contains a collection of models for different types of reactions that can take place in metabolic pathway systems. The reactions are obtained by extending at least one of the 99 reaction types in BioChem.Reactions.ReactionTypes and then adding an equation for the relation between the reaction rate and the participating substances, i.e., substrates, products, and interacting enzymes.

Using more or less all of the possible the reaction types in BioChem.Reactions.ReactionTypes four irreversible and sixteen reversible reaction types for each possible combination of substrates and products are generated, giving 180 different reaction models to choose from in the drag-and-drop interface in MathModelica.

The Systems Biology Markup Language (SBML) is a computer-readable format for representing models of biological and biochemical systems. SBML is, amongst other, applicable to metabolic pathways, cell-signaling pathways, and genomic regulatory networks [21, 22]. SBML has some predefine reactions, which are common in SBML-models of metabolic pathways. All these 32 SBML-reactions are also included in Metabolic.Reactions in order to facilitate the

translation of SBML-models into Modelica, and vice versa. The translation of models is performed with a two-way Modelica-SBML parser [23].

8 Conclusions

During the work with the `BioChem` and the `Metabolic` libraries some limitations of the Modelica language has forced us to re-design the libraries' structure at several points. The original BioChem library [24, 25] was at a point divided into two libraries, i.e., `BioChem` and the `Metabolic`, which made a significant improvement of the library design and hence the underlying library structure. The design that is presented in this paper is currently being extensively tested and has not shown any major shortcomings this far.

9 Future Work

The `BioChem` package will probably have few additions of classes and models in the future, while there will surely be more packages added. As mentioned before, the main purpose of `BioChem` is to serve as a general-purpose package for biological and biochemical Modelica-packages. Some work with ecological models in Modelica has been done with inspiration from the `BioChem` library [26]. These models can now easily be added as a sub-library under `BioChem`.

As for `Metabolic`, the limitations on the number of substrates and products for a reaction will be removed. The construction of a library with metabolic pathway templates will also continue. The idea is that these model templates can easily be extended and adapted to concrete models. The concrete models can then be used in standalone and connected simulations. For all of the above tasks, the data contained in the different resources mentioned in Section 3 will be useful.

Acknowledgments

The authors would like to thank MathCore Engineering AB for supplying the MathModelica tool and Andreas Idebrant for providing essential software support. Morgan Ericsson, Växjö universitet, has provided valuable feedback on the text. Emma Larsdotter Nilsson was in part funded by the Swedish National Graduate School in Computer Science (CUGS).

References

[1] Nicholson, J.K., et al., *Metabonomics: a platform for studying drug toxicity and gene function.* Nat. Rev. Drug. Discov., 2002. **1**(2): p. 153-161.

[2] Klopman, G., M. Dimayuga, and J. Talafous, *META. 1. A program for the evaluation of metabolic transformation of chemicals.* J Chem Inf Comput Sci, 1994. **34**(6): p. 1320-5.

[3] Talafous, J., et al., *META. 2. A dictionary model of mammalian xenobiotic metabolism.* J Chem Inf Comput Sci, 1994. **34**(6): p. 1326-33.

[4] Darvas, F. and G. Dormán, *High-throughput ADMETox estimation : in vitro and in silico approaches*. 2002, Westborough, MA: Eaton Pub. ; BioTechniques Press. x, 89.

[5] Greene, N., et al., *Knowledge-based expert systems for toxicity and metabolism prediction: DEREK, StAR and METEOR.* SAR QSAR Environ Res, 1999. **10**(2-3): p. 299-314.

[6] Kanehisa, M., *The KEGG database.* Novartis Found Symp, 2002. **247**: p. 91-101; discussion 101-3, 119-28, 244-52.

[7] BioCarta, *Proteomic Pathway Project, http://www.biocarta.com*. 2004.

[8] Schomburg, I., et al., *BRENDA, the enzyme database: updates and major new developments.* Nucleic Acids Res, 2004. **32 Database issue**: p. D431-3.

[9] Selkov, E., et al., *The metabolic pathway collection from EMP: the enzymes and metabolic pathways database.* Nucleic Acids Res, 1996. **24**(1): p. 26-8.

[10] Selkov, E., Jr., et al., *MPW: the Metabolic Pathways Database.* Nucleic Acids Res, 1998. **26**(1): p. 43-5.

[11] Karp, P.D., et al., *The EcoCyc and MetaCyc databases.* Nucleic Acids Res, 2000. **28**(1): p. 56-9.

[12] Bugrim, A., T. Nikolskaya, and Y. Nikolsky, *Early prediction of drug metabolism and toxicity: systems biology approach and modeling.* Drug Discov Today, 2004. **9**(3): p. 127-35.

[13] ModelicaAssociation, *Modelica webpage, www.modelica.org*.

[14] Fritzson, P., *Principles of Object-Oriented Modeling and Simulation with Modelica*. 2003: IEEE Press and Wiley.

[15] Fritzson, P. and P. Bunus. *Modelica - A General Object-Oriented Language for Continuous and Discrete-Event System Modeling and Simulation.* in *The 35th Annual Simulation Symposium*. 2002. San Diego, California, USA: IEEE.

[16] Fritzson, P., J. Gunnarsson, and M. Jirstrand. *MathModelica - An Extensible Modeling and Simulation Environment with Integrated Graphics and Literate Programming*. in *The 2nd International Modelica Conference*. 2002. Oberpfaffenhofen, Germany.

[17] MathCoreAB, *MathModelica website, www.mathcore.com*.

[18] DynasimAB, *Dymola website, www.dynasim.se/dymola.htm*.

[19] Wolfram, S., *The Mathematica Book*. 2003: Wolfram Media.

[20] Microsoft, *Visio website, http://office.microsoft.com/en-us/FX010857981033.aspx*.

[21] Hucka, M., et al., *The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models.* Bioinformatics, 2003. **19**(4): p. 524-31.

[22] Finney, A. and M. Hucka, *Systems biology markup language: Level 2 and beyond.* Biochem Soc Trans, 2003. **31**(Pt 6): p. 1472-3.

[23] Vollmar, D., *A two-way SBML translator and graphical icons for BioChem*, in *Deppartment of Computer and Information Science*. 2004, Linköpings universitet: Linköping.

[24] Larsdotter Nilsson, E. and P. Fritzson. *BioChem - A Biological and Chemical Library for Modelica.* in *The 3rd International Modelica Conference*. 2003. Linköping, Sweden.

[25] Larsdotter Nilsson, E. *Simulation of Biological Pathways using Modelica.* in *The Huntsville Simulation Conference 2003*. 2003. Huntsville (AL), USA.

[26] Edelfeldt, S., *Evaluation and comparison of ecological models simulating nitrogen processes in treatment wetlands, implemented in Modelica*, in *Department of Science and Technology*. 2005, Linköpings universitet: Norrköping.