ram: An annotation standard for SBML Level 3 Specification Documents

Henning Lindhorst
henning.lindhorst@ovgu.de
Institute for Automation Engineering
Otto-von-Guericke University Magdeburg
Germany

Alexandra Reimers
alexandra.reimers@fu-berlin.de
Department of Mathematics and Computer Science
Freie Universität Berlin
Germany

Steffen Waldherr
steffen.waldherr@kuleuven.be
Department of Chemical Engineering
KU Leuven
Belgium

Alexander Bockmayr
alexander.bockmayr@fu-berlin.de
Department of Mathematics and Computer Science
Freie Universität Berlin
Germany

Version 1.0.1

21 September 2017

This is a working draft of release 1 of the specification for the annotation standard "ram" and not a normative document. Please submit your feedback via the issue tracker at https://bitbucket.org/hlindhor/sbml-ram-specifications

Contents

1 Introduction & Motivation

In the recent years the systems biology of metabolism has moved more and more from classical metabolic network study towards the study of growth as a result of an optimized cellular economy. This idea of studying growth strategies using resource allocation models has been initiated by Molenaar et al. (2009) in 2009. In their article, Molenaar et al. used a small dynamic model of a self-replicating system to explain how overflow metabolism arises by means of tradeoffs between different growth strategies. Further on, Goelzer et al. (2011) introduced resource balance analysis (RBA), as a means of predicting the cell composition of bacteria in a specific (constant) environment through a convex optimization problem taking the bioernergetic cost of running a pathway into account. More or less in parallel, Lerman et al. (2012) introduced the idea of an integrated model of metabolism and gene expression (ME model) as a means to explore the relationship between genotype and phenotype using biochemical representations of transcription and translation processes. Their research group then continued with an ME model of *Escherichia coli* (O'Brien et al., 2013). Also experimental studies focused on relating absolute protein abundances to how metabolic pathways balance production costs and activity requirements (Li et al., 2014).

These formalisms have then been taken a step forward, towards understanding how resources are distributed in a dynamically changing environment by means of a dynamic enzyme-cost flux balance analysis (deFBA) (Waldherr et al., 2015) and conditional flux balance analysis (cFBA) (Rügen et al., 2015). This has then been taken to the genome scale by studying the optimal glycogen and metabolite partitioning dynamics under a day-night cycle in a cyanobacterium using a dynamic resource allocation model (Reimers et al., ress).

Such dynamic resource allocation models have a wide area of application. One such an example is the study of microorganisms growing in industry-scale bioreactors. There, the organism has to balance resources not only in order to be able to grow optimally, but also in order to withstand transitions through local heterogeneities of the reactor. The ability to take such transitions into account within metabolism has been shown to be crucial for survival (van Heerden et al., 2014). Moreover, an extension of the deFBA formalism has been developed in order to predict the optimal resource allocation in an environment where such uncertainties are present (Lindhorst et al., 2016).

Given all these recent developments, we believe that there is a need to establish a new standard to share these models via SBML. At this point the 'ram' standard is build to share deFBA models and we base this first version of the standard solely on this model type. However, we hope to make it compatible to all types of resource allocation models with the help of the community. For all resource allocation models ('ram'), we need to include information which is not integrated in SBML right now. While the Level 3 package *Flux Balance Constraints* ('fbc') provides the possibility to include gene associations for each reaction in an elegant way, we are missing fields for essential constants.

Thus, we present in this document a standard to encode these via **Annotation** in the species and the reactions. Furthermore, we clearly define the formats of the model in detail to ensure easy exchange between different groups and implementations. In Section 1.1 we give a brief introduction to the deFBA to clarify the used nomenclature. The steps necessary for priming the model are described in Section 2. For the description of the annotations please see Section 3.

1.1 Mathematical description of the deFBA

The dynamic enzyme-cost flux balance analysis models a metabolic reaction network coupled with gene expression as a dynamic optimization problem. By assuming the system to be self-optimizing for growth, (mostly unknown) regulatory features of the network are neglected and the reaction fluxes are used as decision variables for the optimization problem. We present very shortly the mathematical description of the deFBA, so that the reader can understand the problems we face, when building these models.

We will use sets of indices to denote sub-matrices and sub-vectors. For instance, $S_{\mathscr{X},*}$ denotes the sub-matrix of S corresponding to the rows in $\mathscr{X}$ and all columns, while $v_{\mathscr{R}_x}$ denotes the sub-vector of v with the entries at the indices in $\mathscr{R}_x$.

The model consists of n species divided into four different groups:

- the set of *external species* $\mathscr{Y}$, present in the environment (e.g. carbon sources, oxygen, nitrogen), with corresponding molar amounts $y(t) \in \mathbb{R}_{\geq 0}^{|\mathscr{Y}|}$, $\forall\, t \geq 0$,
- the set of *internal metabolic species* $\mathscr{X}$ acting as precursors for the production of biomass (e.g. ATP, NADH, amino acids), with corresponding molar amounts $x(t) \in \mathbb{R}_{\geq 0}^{|\mathscr{X}|}$, $\forall\, t \geq 0$,
- the set of *storage species* $\mathscr{C}$, which save energy for later usage (e.g. starch, glycogen), with corresponding molar amounts $c(t) \in \mathbb{R}_{\geq 0}^{|\mathscr{C}|}$, $\forall\, t \geq 0$,
- the set of *macromolecules* $\mathscr{P}$, which are catalytic enzymes or necessary cellular building blocks, with corresponding molar amounts $p(t) \in \mathbb{R}_{\geq 0}^{|\mathscr{P}|}$, $\forall\, t \geq 0$,

with $n = |\mathscr{Y}| + |\mathscr{X}| + |\mathscr{C}| + |\mathscr{P}|$, $[y] = [x] = [c] = [p] = 1$ mol.

The deFBA model is a dynamic model and hence, all variables described above are considered as functions of time. As in most constraint-based modeling frameworks for metabolism, deFBA assumes that the cell has evolved to maximize its growth in the form of maximizing total biomass at each time point in the simulation period. Thus, we use the *objective weights* b_i, which are typically identical to the molecular weights w_i, $[b_i] = 1$ g/mmol, for all macromolecules $\mathscr{P}$ to define the *objective biomass* B_o until end-time T, $[T] = 1$h, as

$$\int_0^T B_o(t)\,\mathrm{d}t = \int_0^T b_{\mathscr{P}}^T p(t)\,\mathrm{d}t. \tag{1}$$

Additionally, we define the *total biomass* $B_t(t)$ by adding the weight of the storage

$$B_t(t) = B_o(t) + b_{\mathscr{C}}^T c(t). \tag{2}$$

The optimization problem is constructed with the assumption that reaction rates (fluxes) $v_i(t)$, $[v_i(t)] = 1$ mmol/h, which are also time-dependent, are chosen to maximize the biomass accumulation over the simulation time T given the initial macromolecule amounts p_0, and thus our objective function is

$$\max_{v(t)} B(p_0, T) = \max_{v(t)} \int_0^T b_{\mathscr{P}}^T p(t)\,\mathrm{d}t, \tag{3}$$

in which we use the objective biomass (1).

Note that, although in our formulation storage species are not part of the objective biomass, the deFBA formalism does not strictly prohibit this. This means that, if for the modeled organism the storage should be part of the total biomass, this can be incorporated. Furthermore, note that we allow some of the objective weights b_i to be zero, in order to account for the possibility that the modeled organism excretes enzymes that then catalyze external reactions.

As with the species we differentiate the r reactions into four groups

- the set of exchange and external reactions $\mathscr{R}_y$, which transport matter between the cell and the environment or convert external species into each other, with corresponding fluxes $v_{\mathscr{R}_y}(t) \in \mathbb{R}^{|\mathscr{R}_y|}$, $\forall\, t \geq 0$,
- the set of internal metabolic reactions $\mathscr{R}_x$, which convert internal metabolites into each other, with corresponding fluxes $v_{\mathscr{R}_x}(t) \in \mathbb{R}^{|\mathscr{R}_x|}$, $\forall\, t \geq 0$,
- the set of storage reactions $\mathscr{R}_c$, which convert between internal metabolites and storage, with corresponding fluxes $v_{\mathscr{R}_c}(t) \in \mathbb{R}^{|\mathscr{R}_c|}$, $\forall\, t \geq 0$,
- the set of biomass reactions $\mathscr{R}_p$, which synthesize macromolecules from internal metabolite precursors, with corresponding fluxes $v_{\mathscr{R}_p}(t) \in \mathbb{R}^{|\mathscr{R}_p|}$, $\forall\, t \geq 0$,

where $r = |\mathscr{R}_y| + |\mathscr{R}_x| + |\mathscr{R}_c| + |\mathscr{R}_p|$, and the set of all reactions $\mathscr{R}$ is given by $\mathscr{R} = \mathscr{R}_y \cup \mathscr{R}_x \cup \mathscr{R}_c \cup \mathscr{R}_p$.

The differential equations describing the dynamics of the species are given by the stoichiometric matrix $S \in \mathbb{R}^{n \times r}$ as

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix} y(t) \\ x(t) \\ c(t) \\ p(t) \end{pmatrix} = S \begin{pmatrix} v_{\mathscr{R}_y}(t) \\ v_{\mathscr{R}_x}(t) \\ v_{\mathscr{R}_c}(t) \\ v_{\mathscr{R}_p}(t) \end{pmatrix} = \begin{pmatrix} S_{\mathscr{Y},\mathscr{R}_y} v_{\mathscr{R}_y}(t) \\ S_{\mathscr{X},\mathscr{R}_y} v_{\mathscr{R}_y}(t) + S_{\mathscr{X},\mathscr{R}_x} v_{\mathscr{R}_x}(t) + S_{\mathscr{X},\mathscr{R}_c} v_{\mathscr{R}_c}(t) + S_{\mathscr{X},\mathscr{R}_p} v_{\mathscr{R}_p}(t) \\ S_{\mathscr{C},\mathscr{R}_c} v_{\mathscr{R}_c}(t) \\ S_{\mathscr{P},\mathscr{R}_p} v_{\mathscr{R}_p}(t) \end{pmatrix}, \tag{4}$$

for all $t \geq 0$, where the entries $S_{i,j}$ describe the stoichiometry of species i in reaction j. The reactions producing the macromolecules $\mathscr{P}$ and the storage molecules $\mathscr{C}$ usually require a large number of the small metabolites $\mathscr{X}$, which may lead to ill-conditioned dynamics in (4) with large coefficients in $S_{\mathscr{X},\mathscr{R}_p}$, resp. $S_{\mathscr{X},\mathscr{R}_c}$. For effective numerical solution we apply a fixed scaling of macromolecules and storage, which is detailed in (Waldherr et al., 2015).

The numerical complexity of the problem is reduced using a quasi-steady-state assumption for the internal metabolites as

$$\frac{\mathrm{d}}{\mathrm{d}t}x(t) = 0 \Leftrightarrow S_{\mathscr{X},\mathscr{R}_y} v_{\mathscr{R}_y}(t) + S_{\mathscr{X},\mathscr{R}_x} v_{\mathscr{R}_x}(t) + S_{\mathscr{X},\mathscr{R}_c} v_{\mathscr{R}_c}(t) + S_{\mathscr{X},\mathscr{R}_p} v_{\mathscr{R}_p}(t) = 0, \forall\, t \geq 0. \tag{5}$$

Furthermore, flux constraints which are independent of enzymatic capacity can be added as

$$v_{\min} \leq v(t) \leq v_{\max}, \forall\, t \geq 0. \tag{6}$$

In flux balance analysis (FBA) (Orth et al., 2010; Varma and Palsson, 1994), where only the part of the system corresponding to internal and exchange reactions is modeled and a static biomass objective function is maximized, these box constraints are necessary to limit the growth yield, defined as the flux through the biomass reaction. For our application, the limiting factor for the growth rate is the capacity of the enzymes to catalyze the reactions, depending on the *catalytic constants* k_{cat}. These constraints are of the basic form

$$\frac{v}{k_{\text{cat}}} \leq p, \tag{7}$$

but due to possible catalysis of multiple reactions and reversible reactions we must use a slightly more complicated matrix formulation. Hence, we denote the set of reactions catalyzed by the enzyme $\mathscr{P}_i$ as

$$\text{cat}(\mathscr{P}_i) = \{\mathscr{R}_j \mid \mathscr{P}_i \text{ catalyzes } \mathscr{R}_j\}$$

and constrain the reactions fluxes via

$$\sum_{\mathscr{R}_j \in \text{cat}(\mathscr{P}_i)} \left| \frac{v_{\mathscr{R}_j}(t)}{k_{\text{cat}}^{\pm \mathscr{R}_j}} \right| \leq p_i(t), \forall\, t \geq 0,$$

with the forward constant $k_{\text{cat}}^{+\mathscr{R}_j}$ and the backward constant $[k_{\text{cat}}^{-\mathscr{R}_j}, k_{\text{cat}}^{\pm \mathscr{R}_j}] = 1/\text{h}$. Similarly, the amount of ribosome constrains the total rate of protein synthesis in the model. All these constraints can be formulated linearly as

$$H_C v(t) \leq H_E p(t), \ \forall\, t \geq 0, \tag{8}$$

with the *capacity matrix* H_C containing the catalytic constants and the filter matrix H_E containing exactly one non-zero entry per row. The *enzyme capacity constraint* (8) must be satisfied at all times. Assuming any pathway from nutrients to biomass contains at least one reaction limited by an enzyme, the rate of this reaction will be limiting and thus the growth rate will be finite at all times.

In addition to enzymes and ribosomes, deFBA models also include noncatalytic biomass. These are macromolecules of the cell that fulfill no immediate catalytic activity, such as the cell wall or the membrane, but are nevertheless crucial for reproduction and their synthesis consumes cellular resources. We impose a constraint to enforce the production of a certain quota noncatalytic biomass component (which we thus call *quota compound*) in a

proportional way with the catalytic biomass. Consider a quota macromolecule $\mathscr{P}_s$ and assume it must make up 20% of the total biomass B_t at any time point $t \geq 0$. We express this as

$$p_s(t) \geq \phi_s(b_{\mathscr{P}}^T p(t) + b_{\mathscr{C}}^T c(t)), \tag{9}$$

with $\phi_s = 0.2$. We call the according matrix formulation the *biomass composition constraint* and write

$$H_B \begin{pmatrix} c(t) \\ p(t) \end{pmatrix} \leq 0, \ \forall\, t \geq 0. \tag{10}$$

Finally, since deFBA models do not include all resource and energy consuming processes in the cell, an ATP-maintenance reaction may be used to tune the model-derived growth rate and represent additional unmodeled energy sinks. An ATP-maintenance reaction hydrolyzes ATP as

$$\text{ATP} \rightarrow \text{ADP} + \text{Pi}.$$

These reactions are typically enforced proportionally to the total biomass. Thus we assign each maintenance reaction $\mathscr{R}_m$ a scaling factor ψ_m, $[\psi_m] = 1\text{mmol}/(\text{g}\cdot\text{h})$, and write

$$v_{\mathscr{R}_m}(t) \geq \psi_m B_t(t) \Leftrightarrow v(t) \geq H_M \begin{pmatrix} c(t) \\ p(t) \end{pmatrix}, \ \forall\, t \geq 0. \tag{11}$$

We do not include the maintenance reactions as an individual class of reactions as we are usually only handling very few of them in comparison to other reactions.

To formulate the dynamic optimization problem we need to choose initial conditions for the external species y_0, storage species c_0, and the macromolecules p_0. In many cases, one can assume that cells are adapted to achieve maximum growth rate in a certain medium in which they have been cultured before the start of the process modeled by deFBA. To obtain the biomass composition in these cases, a good strategy is to solve an RBA problem (Goelzer et al., 2011) with extracellular species amounts y_0 based on the preculture medium, yielding storage and macromolecule amounts $c_0(y_0)$ and $p_0(y_0)$ for optimal growth in this medium. The initial values are then set as

$$y(0) = y_0, \ c(0) = c_0(y_0), \ p(0) = p_0(y_0). \tag{12}$$

The metabolites $x(t)$ operate in quasi-steady-state (see equation (5)) and thus do not need initial values. The complete deFBA problem then reads

$$\max_{v(t)} \int_0^T B_o(t)\, \mathrm{d}t = \int_0^T b_{\mathscr{P}}^T p(t)\, \mathrm{d}t \tag{3}$$

$$\text{s.t. } \frac{\mathrm{d}}{\mathrm{d}t} \begin{pmatrix} y(t) \\ x(t) \\ c(t) \\ p(t) \end{pmatrix} = S \begin{pmatrix} v_{\mathscr{R}_y}(t) \\ v_{\mathscr{R}_x}(t) \\ v_{\mathscr{R}_c}(t) \\ v_{\mathscr{R}_p}(t) \end{pmatrix}, \qquad \forall\, t \geq 0 \tag{4}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} x(t) = 0, \qquad \forall\, t \geq 0 \tag{5}$$

$$v_{\min} \leq v(t) \leq v_{\max}, \qquad \forall\, t \geq 0 \tag{6}$$

$$H_C v(t) \leq H_E p(t), \qquad \forall\, t \geq 0 \tag{8}$$

$$H_B \begin{pmatrix} c(t) \\ p(t) \end{pmatrix} \leq 0, \qquad \forall\, t \geq 0 \tag{10}$$

$$v(t) \geq H_M \begin{pmatrix} c(t) \\ p(t) \end{pmatrix}, \qquad \forall\, t \geq 0 \tag{11}$$

$$y(0) = y_0, \ c(0) = c_0, \ p(0) = p_0 \tag{12}$$

$$y(t), \ c(t), \ p(t) \geq 0, \qquad \forall\, t \geq 0 \tag{13}$$

To solve this dynamic optimization problem, time is discretized using a collocation method and the whole problem is cast into a linear program (LP) as described in (Waldherr et al., 2015).

2 Usage of 'fbc'-package

2.1 Using 'fbc' for genetic information

To keep the deFBA model unique in terms of the SBML representation, we need to discuss the handling of enzymatic catalysis of the reactions. As a first step, we eliminate dependencies of single reactions to multiple catalyzing enzymes. As example consider the following reaction with genome information in the notes in COBRA format and in the GeneProductAssociation from the 'fbc'-package.

```
<reaction id="reaction_1" reversible="true" fast="false">
  <notes>
    <body xmlns="http://www.w3.org/1999/xhtml">
      <p>GENE_ASSOCIATION:(YGR143W OR YPR159W)</p>
    </body>
  </notes>
  <fbc:geneProductAssociation fbc:id="reaction_1">
    <fbc:or>
      <fbc:geneProductRef fbc:geneProduct="YGR143W" />
      <fbc:geneProductRef fbc:geneProduct="YPR159W" />
    </fbc:or>
  </fbc:geneProductAssociation>
</reaction>
```

While it is possible to use the COBRA format with the 'ram' standard, we only support the usage of the 'fbc' package for genetic information and biomass independent flux bounds. When constructing a deFBA model containing a reaction with multiple catalysts, we interpret it as two distinct reactions, which are independent from each other. To enforce this interpretation, we split the reaction already in the SBML file.

Furthermore, we handle reactions catalyzed by an enzyme complex, which is built from multiple **fbc:geneProduct**, by introduction of a new catalytic compound. Consider the following reaction.

```
<reaction id="reaction_2" reversible="false" fast="false">
  <fbc:geneProductAssociation fbc:id="reaction2">
    <fbc:and>
      <fbc:geneProductRef fbc:geneProduct="YVR173" />
      <fbc:geneProductRef fbc:geneProduct="YKT009W" />
    </fbc:and>
  </fbc:geneProductAssociation>
</reaction>
```

Instead of constructing the two gene products individually, we only introduce a new "super"-enzyme, which we call *complex*, and use as the catalytic compound. The individual components YKT009W and YVR173 are then are replaced with the complex in the file. We suggest to save the composition of the complex is saved in the **fbc:label** field in the **fbc:geneProduct**, see also Section 3.3.

Thus, each reaction has either exactly one **fbc:geneProductRef** or it is not constrained by the enzymatic constraints (8). In deFBA all macromolecules, especially the enzymes, are included as dynamic variables and each **fbc:geneProduct** has a corresponding **Species**. We encode this information in the **fbc:associatedSpecies** attribute of the **fbc:geneProduct** elements.

2.1.1 Example

Consider the two reactions from the previous Section. The correct inclusion of the genetic information can look like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<sbml xmlns="http://www.sbml.org/sbml/level3/version1/core" level="3" version="1"
 xmlns:fbc="http://www.sbml.org/sbml/level3/version1/fbc/version2" fbc:required="false">
 <model id="genetic information" fbc:strict="false">
   <listOfCompartments>
     <compartment id="bio" name="macromolecules" constant="true"/>
   </listOfCompartments>
   <listOfSpecies>
     <species id="A_bio" compartment="bio" constant="false"/>
     <species id="B_bio" compartment="bio" constant="false"/>
     <species id="C_bio" compartment="bio" constant="false"/>
   </listOfSpecies>
   <fbc:listOfGeneProducts>
     <fbc:geneProduct fbc:id="YGR143W" fbc:associatedSpecies="A_bio" fbc:label="1*YGR143W"/>
     <fbc:geneProduct fbc:id="YPR159W" fbc:associatedSpecies="B_bio" fbc:label="1*YPR159W"/>
     <fbc:geneProduct fbc:id="Complex_1" fbc:associatedSpecies="C_bio"
      fbc:label="1*YVR173 AND 1*YKT009W"/>
   </fbc:listOfGeneProducts>
   <listOfReactions>
     <reaction id="reaction_1_1" reversible="false" fast="false">
       <fbc:geneProductAssociation fbc:id="reaction_1_1">
         <fbc:geneProductRef fbc:geneProduct="YGR143W" />
       </fbc:geneProductAssociation>
     </reaction>
     <reaction id="reaction_1_2" reversible="false" fast="false">
       <fbc:geneProductAssociation fbc:id="reaction_1_2">
         <fbc:geneProductRef fbc:geneProduct="YPR159W" />
       </fbc:geneProductAssociation>
     </reaction>
     <reaction id="reaction_2" reversible="false" fast="false">
       <fbc:geneProductAssociation fbc:id="reaction_2">
         <fbc:geneProductRef fbc:geneProduct="Complex_1" />
       </fbc:geneProductAssociation>
     </reaction>
   </listOfReactions>
 </model>
</sbml>
```

3 The 'ram' specifications

3.1 Compartments

We define no special usuages for the **Compartment** elements and should be used as the SBML specifications intend. This means the biomass products we add to the original metabolic network should be placed according to their physical location. E.g. enzymes should be placed where the reaction they are catalyzing is happening.

If this should not be possible, 'ram' allows for an arbitrary choice of compartments for all enzymes, quota or storage species. For large scale systems we suggest to incorporate the id of the **Compartment** into the id of the **Species**. More on that topic in the Section 3.2.1.

3.2 Species

Each **Species** element in the **listOfSpecies** must contain the following attributes.

- **id**
  - string. As we do not use **metaids**, this id must be unique among the species. Other elements, such as **fbc:geneProduct**, may share ids with **Species**-elements. We suggest to add the **id** of the **Compartment** to avoid duplicates, if the same species occurs in multiple compartments. We elaborate on this problem later in this section.
- **Compartment**
  - string. Points to the **id** of a **Compartment** in the **listOfCompartments**.
- **constant**
  - boolean. Specifies whether the species is regarded as a fixed species. If this set to "true" the amount of the species can not change after the initial assignment.
- **boundaryCondition**
  - boolean. Specifies, whether a species is at the systems boundary. The amounts of a boundary species can not be changed by the metbaolic reactions, but by rules, e.g. **AssignmentRule**, **RateRule**. Nonlimiting external species, marked with **boundaryCondition**="true" and **constant**="true" are substituted in the deFBA model with the empty set.
- **hasOnlySubstanceUnits**
  - boolean. Always set to "true" because the deFBA models are using molar amounts as unit for the species.

Additionally, each species element can have the following optional attributes:

- **name**
  - string. Additional information about species.
- **initialAmount**
  - double. If the model provides initial values, these can be set here. The attribute is only accepted if all non-limiting external species and biomass species are equipped with the attribute. Constant species do not need the attribute.

3.2.1 Guideline to ensure uniqueness of macromolecule ids

There are some enzymes that can act in different compartments of the cell. An example is fumarase, which catalyzes reactions both in the cytosol and in the mitochondrion in yeast. While we include the respective compartments for the species in their description, a common error is to give both enzymes the same **id**. Hence, we suggest to name enzymes in a specific pattern combining e.g. their name and their respective location.

- If the enzyme is acting in only one compartment we choose its **id** in the format "Main_id_[acting-compartment]".
- If the enzyme is a transporter between two compartments we choose "Main_id_[compartment1]_[compartment2]".
- The main **id** is simply chosen as "E_reaction_id".

Following these naming conventions also allows the user to read alot of the enzymes function by just looking at the respective **id**.

Of course, the user can choose these **id**s freely, but following these suggestions can help with easier evaluation of the model.

3.2.2 Species and ram:species

The 'ram' standard defines a format for a new xml-node inside the **Annotation** environment, defined as **ram:species**. The **ram:species**-nodes are mandatory for all species as they are used to classify the species depending on their function. Attributes of the nodes are

- **ram:speciesType**
  - string. Must be either 'extracellular', 'metabolite', 'enzyme', 'storage' or 'quota' to identify the function of the species in the deFBA model.
- **ram:molecularWeight**
  - string. Pointing at a **parameter id** containing the molecular weight, cf. w in Section 1.1. External species or metabolites do not need a molecular weight. Thus, their attributes can be left empty, which is interpretated as a value of zero.
- **ram:objectiveWeight**
  - string. Pointing at a **parameter id** containing the objective weight, cf. b in Section 1.1. If the species should not be included in the objective function set this to "zero". Because of the importance of the objective weight, empty string lead to a parsing error.
- **ram:biomassPercentage**
  - string. Pointing to a **parameterid** containing the biomass percentage, cf. H_B matrix (10). If the species should not be included in the biomass composition constraint (10) set this to "zero".

Example for a **Species** element

```
<species id="Emetab1" name="Generic metabolic enzyme" compartment="bio" initialAmount="1.1"
constant="false" hasOnlySubstanceUnits="true" boundaryCondition="false">
  <annotation>
    <ram:RAM xmlns:ram="https://www.fairdomhub.org/sops/304">
      <ram:species ram:molecularWeight="weight_2" ram:objectiveWeight="weight_12"
      ram:biomassPercentage="zero" ram:speciesType="enzyme"/>
    </ram:RAM>
  </annotation>
</species>
```

3.3 Reactions

Each **Reaction** element in the **ListOfReactions** must contain the following

- **id**
  - string. As we do not use **metaids**, this id must be unique among the reactions. We recommend to start all macromolecule producing reactions with 'synth_' to distinguish them by id.
- **reversible**
  - boolean. Decides whether the reaction is reversible. We handle this attribute as strict, so all reactions must be formulated s.t. the non-reversible reactions are happening in the positive direction with $V_{\min} = 0$ (6).
- **fast**
  - boolean. The **fast** attribute is deprecated and will be removed completely in SBML Level 3 Version 2 (L3S2). We will also delete it then but keep it right now as a strictly "false" attribute to avoid compatibility issues.
- **listOfReactants** (may be empty)
- **listOfProducts** (may be empty)

Additionally, each **Reaction** element can have:

- **name**
  - string. Additional information about reaction.
- **fbc:geneProductAssociation**
  - cf. Section 2.1. States the enzyme catalyzing this reaction. We only allow each reaction to be catalyzed by one enzyme. Hence, it is necessary to create copies of the same reaction, if multiple enzymes can act as catalyst.

3.3.1 Reactions and ram:reaction

The 'ram' standard defines a format for a new xml-node inside the **Annotation** environment, defined as **ram:reaction**. The **ram:reaction**-node is only mandatory for a **Reaction**, which either has a **fbc:geneAssociation** or is a maintenance reaction (11). Attributes of the nodes are

- **ram:maintenanceScaling**
  - string. Pointing at a **parameter** containing the double value for the construction of the matrix H_M (cf. (11)). If the **Reaction** is no maintenance reaction this must be set to "zero".
- **ram:kcatForward**
  - string. Pointing to a **parameter** containing the value for the $k_{\text{cat}}^{+\mathscr{R}_j}$, cf. H_C matrix (8). Reactions without a **fbc:geneProductAssociation** can leave this attribute empty.
- **ram:kcatBackward**
  - string. Pointing to a **parameter** containing the value for the $k_{\text{cat}}^{-\mathscr{R}_j}$, cf. H_C matrix (8). If the **Reaction** is reversible but has no**fbc:geneProductAssociation** this attribute can be left empty. If the **Reaction** is irreversible (**reversible**="false"), this value must be zero.

Example for a **Reaction** element:

```
<reaction id="PMetab1" reversible="false" fast="false">
  <annotation>
    <ram:RAM xmlns:ram="https://www.fairdomhub.org/sops/304">
      <ram:reaction ram:kcatForward="kcatR3" ram:kcatBackward="zero" ram:maintenanceScaling="zero"/>
    </ram:RAM>
  </annotation>
  <fbc:geneProductAssociation fbc:id="Ribosome">
      <fbc:geneProductRef fbc:geneProduct="R" />
  </fbc:geneProductAssociation>
  <listOfReactants>
    <speciesReference species="AA" stoichiometry="200" constant="true"/>
    <speciesReference species="ATP" stoichiometry="800" constant="true"/>
  </listOfReactants>
  <listOfProducts>
    <speciesReference species="Emetab1" stoichiometry="1" constant="true"/>
  </listOfProducts>
</reaction>
```

4 Full example

A full working example is the following resource allocation model. This model contains only some aspects of the 'ram' standard as it poses as a minimal example. And this example and more detailed examples utilizing all aspects of 'ram' can be downloaded at https://hlindhor@bitbucket.org/hlindhor/sbml-ram-specifications.

```
<?xml version="1.0" encoding="UTF-8"?>
<sbml xmlns="http://www.sbml.org/sbml/level3/version1/core" level="3" version="1"
xmlns:fbc="http://www.sbml.org/sbml/level3/version1/fbc/version2" fbc:required="false">
  <model id="enzymatic_growth" name="enzymatic_growth" fbc:strict="false">

    <listOfCompartments>
      <compartment id="external" name="extracellular compartment. nutrients, waste, etc."
      spatialDimensions="3" size="1" constant="true"/>
      <compartment id="cytosol" name="cytosol. Collecting all non external components"
      spatialDimensions="3" size="1" constant="true"/>
    </listOfCompartments>

    <listOfSpecies>
      <species id="N"   compartment="external"    initialAmount="2000"  constant="false"
      boundaryCondition="true" hasOnlySubstanceUnits="true">
        <annotation>
    <ram:RAM xmlns:ram="https://www.fairdomhub.org/sops/304">
      <ram:species ram:molecularWeight="zero" ram:objectiveWeight="zero"
      ram:biomassPercentage="zero" ram:speciesType="extracellular"/>
    </ram:RAM>
  </annotation>
      <species id="A"   compartment="cytosol"     initialAmount="0"     constant="false"
      boundaryCondition="false" hasOnlySubstanceUnits="true">
        <annotation>
    <ram:RAM xmlns:ram="https://www.fairdomhub.org/sops/304">
      <ram:species ram:molecularWeight="zero" ram:objectiveWeight="zero"
      ram:biomassPercentage="zero" ram:speciesType="metabolite"/>
    </ram:RAM>
  </annotation>
      <species id="M"   compartment="cytosol"     initialAmount="0.1"   constant="false"
      boundaryCondition="false" hasOnlySubstanceUnits="true">
        <annotation>
          <ram:RAM xmlns:ram="https://www.fairdomhub.org/sops/304">
            <ram:species ram:molecularWeight="weighM" ram:objectiveWeight="oWeightM"
            ram:biomassPercentage="zero" ram:speciesType="storage"/>
          </ram:RAM>
        </annotation>
      </species>
      <species id="E"   compartment="cytosol"     initialAmount="0.1"   constant="false"
      boundaryCondition="false" hasOnlySubstanceUnits="true">
        <annotation>
          <ram:RAM xmlns:ram="https://www.fairdomhub.org/sops/304">
            <ram:species ram:molecularWeight="weighE" ram:objectiveWeight="oWeightE"
            ram:biomassPercentage="zero" ram:speciesType="enzyme"/>
          </ram:RAM>
```

```
    </annotation>
  </species>
</listOfSpecies>

<listOfParameters>
  <parameter constant="true" id="zero"     value="0"    />
  <parameter constant="true" id="weighM"   value="150"  />
  <parameter constant="true" id="weighE"   value="100"  />
  <parameter constant="true" id="oWeightM"  value="150"  />
  <parameter constant="true" id="oWeightE"  value="100"  />
  <parameter constant="true" id="kcatA"    value="150"  />
  <parameter constant="true" id="kcatE"    value="1"    />
  <parameter constant="true" id="kcatM"    value="2"    />
</listOfParameters>

<fbc:listOfGeneProducts>
  <fbc:geneProduct fbc:id="E" fbc:label="enzymes" fbc:associatedSpecies="E"/>
</fbc:listOfGeneProducts>

<listOfReactions>
  <reaction id="VA" reversible="false" fast="false">
    <fbc:geneProductAssociation fbc:id="Enzymes">
        <fbc:geneProductRef fbc:geneProduct="E" />
    </fbc:geneProductAssociation>
    <listOfReactants>
      <speciesReference species="N" stoichiometry="1" constant="true"/>
    </listOfReactants>
    <listOfProducts>
      <speciesReference species="A" stoichiometry="1" constant="true"/>
    </listOfProducts>
    <annotation>
      <ram:RAM xmlns:ram="https://www.fairdomhub.org/sops/304">
        <ram:reaction ram:kcatForward="kcatA" ram:kcatBackward="zero"
        ram:maintenanceScaling="zero"/>
      </ram:RAM>
    </annotation>
  </reaction>
  <reaction id="VE" reversible="false" fast="false">
    <fbc:geneProductAssociation fbc:id="Enzymes">
        <fbc:geneProductRef fbc:geneProduct="E" />
    </fbc:geneProductAssociation>
    <listOfReactants>
      <speciesReference species="N" stoichiometry="100" constant="true"/>
      <speciesReference species="A" stoichiometry="100" constant="true"/>
    </listOfReactants>
    <listOfProducts>
      <speciesReference species="E" stoichiometry="1" constant="true"/>
    </listOfProducts>
    <annotation>
      <ram:RAM xmlns:ram="https://www.fairdomhub.org/sops/304">
        <ram:reaction ram:kcatForward="kcatE" ram:kcatBackward="zero"
        ram:maintenanceScaling="zero"/>
```

```
          </ram:RAM>
        </annotation>
      </reaction>
      <reaction id="VM" reversible="false" fast="false">
        <fbc:geneProductAssociation fbc:id="Enzymes">
            <fbc:geneProductRef fbc:geneProduct="E" />
        </fbc:geneProductAssociation>
        <listOfReactants>
          <speciesReference species="N" stoichiometry="100" constant="true"/>
          <speciesReference species="A" stoichiometry="100" constant="true"/>
        </listOfReactants>
        <listOfProducts>
          <speciesReference species="M" stoichiometry="1" constant="true"/>
        </listOfProducts>
  <annotation>
          <ram:RAM xmlns:ram="https://www.fairdomhub.org/sops/304">
            <ram:reaction ram:kcatForward="kcatM" ram:kcatBackward="zero"
            ram:maintenanceScaling="zero"/>
          </ram:RAM>
        </annotation>
      </reaction>
    </listOfReactions>
  </model>
</sbml>
```

5 Acknowledgments

This work was made possible by a grant from the German Federal Research Ministry (BMBF) Fkz. 031L0017A.

References

Goelzer, A., Fromion, V., and Scorletti, G. (2011). Cell design in bacteria as a convex optimization problem. *Automatica*, 47(6):1210–1218.

Lerman, J. A., Hyduke, D. R., Latif, H., Portnoy, V. A., et al. (2012). In silico method for modelling metabolism and gene product expression at genome scale. *Nature Communications*, 3.

Li, G.-W., Burkhardt, D., Gross, C., and Weissman, J. S. (2014). Quantifying absolute protein synthesis rates reveals principles underlying allocation of cellular resources. *Cell*, 157(3):624–635.

Lindhorst, H., Lucia, S., Findeisen, R., and Waldherr, S. (2016). Modeling metabolic networks including gene expression and uncertainties. *arXiv preprint arXiv:1609.08961*.

Molenaar, D., van Berlo, R., de Ridder, D., and Teusink, B. (2009). Shifts in growth strategies reflect tradeoffs in cellular economics. *Molecular Systems Biology*, 5(1).

O'Brien, E. J., Lerman, J. A., Chang, R. L., Hyduke, D. R., and Palsson, B. Ø. (2013). Genome-scale models of metabolism and gene expression extend and refine growth phenotype prediction. *Molecular Systems Biology*, 9(1):693.

Orth, J. D., Thiele, I., and Palsson, B. Ø. (2010). What is flux balance analysis? *Nature Biotechnology*, 28(3):245–248.

Reimers, A.-M., Knoop, H., Bockmayr, A., and Steuer, R. (2017 (in press)). Evaluating cellular trade-offs and optimal resource allocation during cyanobacterial diurnal growth. *Proceedings of the National Academy of Sciences*.

Rügen, M., Bockmayr, A., and Steuer, R. (2015). Elucidating temporal resource allocation and diurnal dynamics in phototrophic metabolism using conditional FBA. *Scientific Reports*, 5:15247.

van Heerden, J. H., Wortel, M. T., Bruggeman, F. J., Heijnen, J. J., Bollen, Y. J., Planqué, R., Hulshof, J., OâĂŹToole, T. G., Wahl, S. A., and Teusink, B. (2014). Lost in transition: start-up of glycolysis yields subpopulations of nongrowing cells. *Science*, 343(6174):1245114.

Varma, A. and Palsson, B. Ø. (1994). Metabolic Flux Balancing: Basic Concepts, Scientific and Practical Use. *Nature Biotechnology*, 12(10):994–998.

Waldherr, S., Oyarzún, D. A., and Bockmayr, A. (2015). Dynamic optimization of metabolic networks coupled with gene expression. *Journal of Theoretical Biology*, 365:469–485.