June 12, 2007
Draft by Michael Blinov
Comments by Oliver Ruebenacker, Ion Moraru

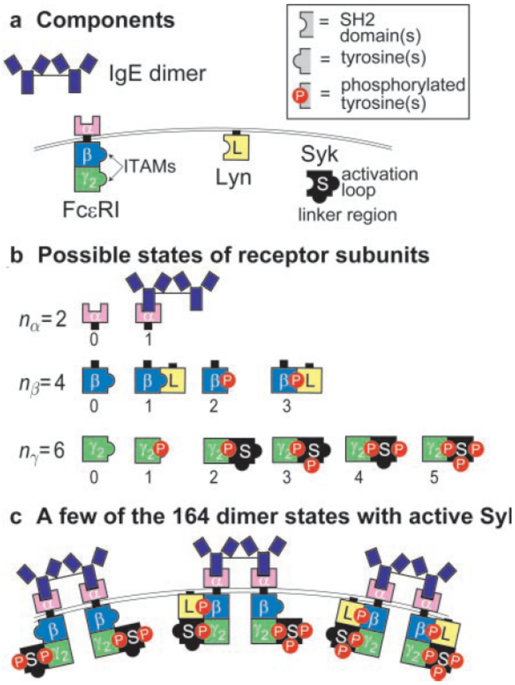
SBML L3: features necessary for description of rule-based modeling

1 Introduction

This document describes proposed features for inclusion in Systems Biology Markup Language (SBML) Level 3 that will enable the description of (1) complex chemical entities that are composed from other chemical entities, and (2) rule-based operations on such entities.

This document is not a complete proposal of SBML Level 3, but it presents features that are necessary for a rule-based modeling specification and that should be incorporated into SBML Level 3, namely multi-state multi-component species located in several compartments that undergo modifications of different components and bind multiple binding partners in multiple compartments. We will discuss how to encode in SBML:

(1) Multi-state multi-component species expressed as attributed graphs with explicit representation of the connectivity of species' components.
(2) Reactions in the form of graph transformations.
(3) Patterns that select groups of species having user-specified properties.
(4) Rules that define transformations of sets of species selected by some pattern.



In what follows we will consider a model for the early events in Fc_RI signaling induced by a bivalent ligand (Goldstein et al, 2002 and Faeder et al., 2003) *(a)* The bivalent ligand shown is a covalently cross-linked IgE dimer. The tetrameric form of FceRI, the high affinity receptor for IgE, consists of an alpha subunit that binds IgE, a beta subunit, and a disulfide-linked dimer of gamma subunits. The alpha and beta subunits contain ITAMs that become phosphorylated on tyrosine residues. The membrane-associated kinase Lyn can associate with the unphosphorylated or phosphorylated beta subunit. The cytosolic kinase Syk associates with a doubly phosphorylated gamma ITAM. On the representations of the kinases, notches indicate SH2 domains. The tandem SH2 domains of Syk are lumped together in the representation. On beta, gamma, and Syk, bumps represent groups of tyrosines that are lumped in the model as single targets for phosphorylation. On Syk, the model distinguishes tyrosines in the linker region, which are phosphorylated by Lyn, and those in the activation loop, which are phosphorylated by Syk. (*b*) The alpha subunit can be bound to ligand or unbound (two states of alpha). The beta subunit can be unphosphorylated or phosphorylated, with or without associated Lyn (four states of beta). The gamma dimer can be unphosphorylated or phosphorylated, and the phosphorylated form can be bound to Syk in any of four states of phosphorylation (six states of gamma). (*c*) There are 300 possible states for crosslinked FceRI dimers, of which 164 contain autophosphorylated Syk. Three of these states containing active Syk are shown.

2 New data needed for rule-based specification of multi-state multi-component species

Below we specify main features in SBML specification that we propose to add or modify in order to specify a rule-based model of FceRI (Faeder et al., 2003).

Here we follow the paper by Blinov et al. (2006), where species are introduced as graphs. So below we will use graph terminology.

- **speciesTypes – modified**. In addition to current use for relating species located in multiple compartments, we suggest to use speciesType to declare physical entities in all potential modification forms. Examples of speciesType include:
    - A tyrosine residue that can be phosphorylated or unphosphorylated.
    - Protein that is comprised of five tyrosine residues and in addition can be folded or unfolded.
    - Complex of two proteins connected via association of SH2 domain of one with the phosphorylated residue of the second can be optionally declared as a speciesType. In the last example, configuration space of this speciesType includes all modification of both proteins that do not break the bond between SH2 and phosphotyrosine.

  In graph-theoretical language, speciesType is a connected graph with vertices that may have different states. The simplest speciesType (such as a tyrosine residue) would be a single vertex that may have different states (such as phosphorylated and unphosphorylated). speciesType can be further constructed of other speciesTypes connected by bonds like a graph is constructed of vertices connected by edges.
- **Species – modified**. As before, species are uniquely defined chemical entities. Species have the same use as in SBML L2: specify initial concentrations and be referenced as reactants or products in reactions. The new feature is that each species is represented as a graph with each vertex (speciesType) being fully defined (taking a single state out of the potential set of states declared in speciesType for this vertex). Edges of this graph are chemical bonds between speciesTypes. Species can be of specific single speciesType, but can be also a graph of multiple speciesTypes connected by bonds.
- **SpeciesPattern – new**. Used as a pattern that can select any arbitrary user-specified sets of species. In graph-theoretical language, a pattern is a subgraph. Examples of speciesPatterns:
    - A single speciesType (vertex) in a specific state, e.g. tyrosine residue being phosphorylated. This speciesPattern selects all species that have this vertex in the given state, e.g. it selects all species with phosphotyrosines.
    - A speciesPattern can be declared as a speciesType with all but a single vertex having specified states, e.g. a folded protein with all but one of tyrosines being phosphorylated. This speciesPattern selects only as many species as there are different states in the speciesType declaration for this vertex, e.g. two species with a given tyrosine being phosphorylated or not.
- **Reactions – modified**. The main change is that SpeciesReference may be replaced with speciesPatternReference. In this case a reaction operates on the set of species selected by a speciesPattern and and effectively becomes a reaction rule. We propose no special tag for a reaction rule. Depending on whether reactants or products are uniquely defined species or sets of species, a reaction can be a regular reaction or a reaction rule.

3 speciesTypes

Modification of speciesType is a key feature of this SBML Level 3 proposal. It extends the speciesType of SBML L2 in that it declares chemical entities with specified shared features.

3.1 The simplest speciesType: component

The simplest speciesType is a component (such as a tyrosine residue) that is represented by vertex. This simplest speciesType provides a basic module that can be used to construct more complex chemical species, and is capable of assuming one of any number of user-enumerated states. The term component is an optional attribute class introduced for compatibility with biological ontologies and software tools like BioNetGen.

speciesType	Comments
id : SId	global
name : string {use="optional"}	
class: string {"component", "physicalEntity", …} {use="optional"}	for use with ontology and software
speciesTypeState: string [0..*] {use="optional"}	
compartment: Sid {use="optional"}	

Example 3.1.1. A prototypical phosphorylation site.

```
<speciesType id="p-site" class="components" name="site_of_phosphorylation">
    <listOfSpeciesTypeStates>
        <speciesTypeState value="u" name="unphosphorylated" default="true"/>
        <speciesTypeState value="p" name="phosphorylated"/>
    </listOfSpeciesTypeStates>
</speciesType>
```

The speciesType above, "p-site", is capable of assuming one of two states, an “unphosphorylated” state, and a "phosphorylated" state. Any species whose definition includes this component will have its state space doubled in that all other states of the species may occur with either the unphosphorylated or phosphorylated state.

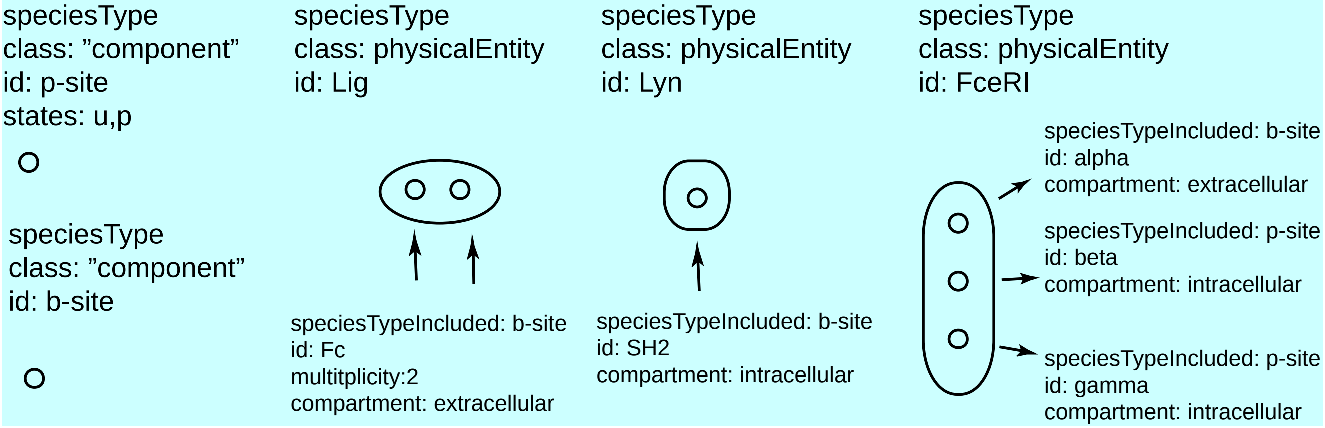
For the model of interest, the list of speciesTypes of the class component looks as follows:

```
<listOfSpeciesTypes class="component" >
  <speciesType id="b-site" name="site_of_binding"/>
  <speciesType id="p-site" name="site_of_phosphorylation">
     <listOfSpeciesTypeStates>
        <speciesTypeState value="u" name="unphosphorylated" default="true"/>
        <speciesTypeState value="p" name="phosphorylated" />
      </listOfSpeciesTypeStates>
  </speciesType>
</listOfSpeciesTypes>
```

3.2 *Example of speciesTypes: physicalEntities*

The next level in hierarchy of speciesTypes is physicalEntity, constructed of other speciesTypes (components) that are declared as speciesTypeIncluded:

speciesTypeIncluded	Comments
id : SId	
name : string {use="optional"}	
multiplicity: int { minInclusive="0" use="optional" default="1"}	The number of identical speciesTypeIncluded within SpeciesType
maxExternalBonds: int { minInclusive="0" use="optional" default="1"}	Maximal number of external bonds. The default value means that each speciesTypeIncluded can be connected to at most a single speciesType outside the enclosing speciesType.
maxInternalBonds: int { minInclusive="0" use="optional" default="0"}	Number of internal bonds. The default value means that there are no explicit internal bonds.


speciesType
class: ”component”
id: p-site
states: u,p
speciesType
class: ”component”
id: b-site
speciesType
class: physicalEntity
id: Lig
speciesTypeIncluded: b-site
id: Fc
multitplicity:2
compartment: extracellular
speciesType
class: physicalEntity
id: Lyn
speciesTypeIncluded: b-site
id: SH2
compartment: intracellular
speciesType
class: physicalEntity
id: FceRI
speciesTypeIncluded: b-site
id: alpha
compartment: extracellular
speciesTypeIncluded: p-site
id: beta
compartment: intracellular
speciesTypeIncluded: p-site
id: gamma
compartment: intracellular

```
<speciesType class="physicalEntity" id="Lig">
   <listOfSpeciesTypesIncluded>
     <speciesTypeIncluded id="Fc" speciesType="b-site" multiplicity="2" compartment="extracellular"/>
   </listOfSpeciesTypesIncluded>
</speciesType>

<speciesType class="physicalEntity" id="Lin">
   <listOfSpeciesTypesIncluded>
       <speciesTypeIncluded id="SH2" speciesType="p-site" compartment="intracellular"/>
   </listOfSpeciesTypesIncluded>
</speciesType>

<speciesType class="physicalEntity" id="FceRI">
   <listOfSpeciesTypesIncluded>
     <speciesTypeIncluded id="alpha" speciesType="b-site" compartment="extracellular"/>
     <speciesTypeIncluded id="beta" speciesType="p-site" compartment="intracellular"/>
     <speciesTypeIncluded id="gamma" speciesType="p-site" compartment="intracellular"/>
```

```
  </listOfSpeciesTypesIncluded>
</speciesType>
```

3.3 Use of compartment attribute

Each speciesType may have an optional attribute “compartment” - restricting binding partners of this speciesType to only those located within this compartment. It does not restrict binding partners of speciesTypesIncluded! Namely, speciesType FceRI (physicalEntity) can have a compartment membrane. Meanwhile, FceRI consists of three speciesTypesIncluded (components): alpha of speciesType b-site with compartment extracellular, beta of speciesType p-site with compartment intracellular, and gamma of speciesType p-site with compartment intracellular. In this construct, if one specified direct binding of speciesType FceRI to something, this something must be in the membrane compartment. But speciesTypes alpha, beta and gamma can interact only with specicesTypes within corresponding compartments. If some of speciesTypes have no compartment specified, then the compartment attribute of higher hierarchical speciesType is applied.

3.4 Use of bonds

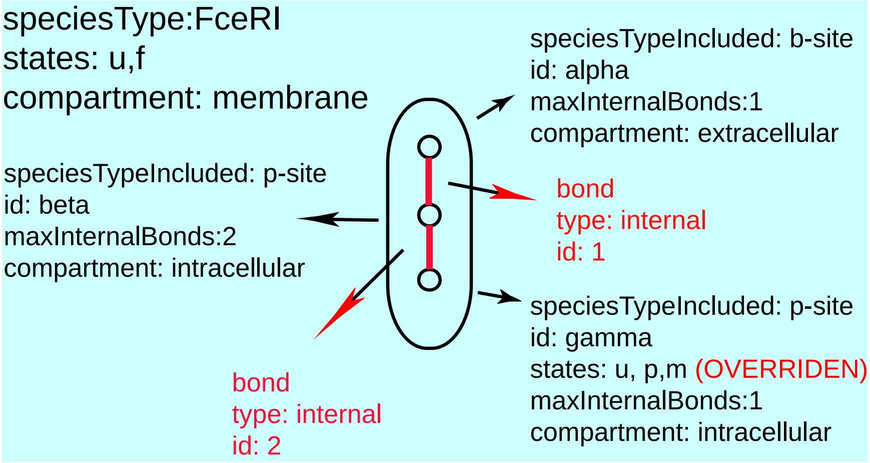
bond	Comments

id : SId
name : string {use="optional"}
bondType: string {"external" , "internal"}
	External bond connects speciesTypesIncluded to other speciesTypes outside of the enclosing speciesType.
	Internal bond connects speciesTypesIncluded to other speciesTypeIncluded within the enclosing speciesType.

Before we specified species, we can illustrate the use of internal bonds only. In the example below, maxInternalBonds is specified for each component, and components within physical entity are connected. Below is an example of speciesType with:

- multiple speciesTypesIncluded;
- Internal bonds between speciesTypesIncluded specified.

In addition, some states of speciesTypesIncluded are overridden.


speciesType:FceRI
states: u,f
compartment: membrane
speciesTypeIncluded: b-site
id: alpha
maxInternalBonds:1
compartment: extracellular
speciesTypeIncluded: p-site
id: beta
maxInternalBonds:2
compartment: intracellular
bond
type: internal
id: 1
speciesTypeIncluded: p-site
id: gamma
states: u, p,m (OVERRIDEN)
maxInternalBonds:1
compartment: intracellular
bond
type: internal
id: 2

```
<speciesType class="physicalEntity" id="FceRI" compartment="membrane">
  <listOfSpeciesTypeStates>
      <speciesTypeState value="u" name="unfolded" />
      <speciesTypeState value="f" name="folded" />
   </listOfSpeciesTypeStates>
   <listOfSpeciesTypesIncluded>
     <speciesTypeIncluded id="alpha" speciesType="b-site" compartment="extracellular"
maxInternalBonds ="1">
          <listOfBondReferences>
              <bondReference bond="1"/>
          </listOfBondReferences>
     <speciesTypeIncluded/>
     <speciesTypeIncluded id="beta" speciesType="p-site" compartment="intracellular" maxInternalBonds
="2">
          <listOfBondReferences>
              <bondReference bond="1"/>
              <bond Reference bond="2"/>
          </listOfBondReferences>
     <speciesTypeIncluded/>
     <speciesTypeIncluded id="gamma" speciesType="p-site" compartment="intracellular"
maxInternalBonds ="1">
         <listOfSpeciesTypeStates>
            <speciesTypeState id="u" name="unphosphorylated" default="true"/>
            <speciesTypeState id="p" name="phosphorylated"/>
            <speciesTypeState id="m" name="modified"/>
         </listOfSpeciesTypeStates>
          <listOfBondReferences>
              <bondReference bond="2"/>
          </listOfBondReferences>
      <speciesTypeIncluded/>
   </listOfSpeciesTypesIncluded>
   <listOfBonds>
       <bond id="1" bondType="internal"/>
       <bond id="2" bondType="internal"/>
   </listOfBonds>
</speciesType>
```

4 Specices and speciesPattern

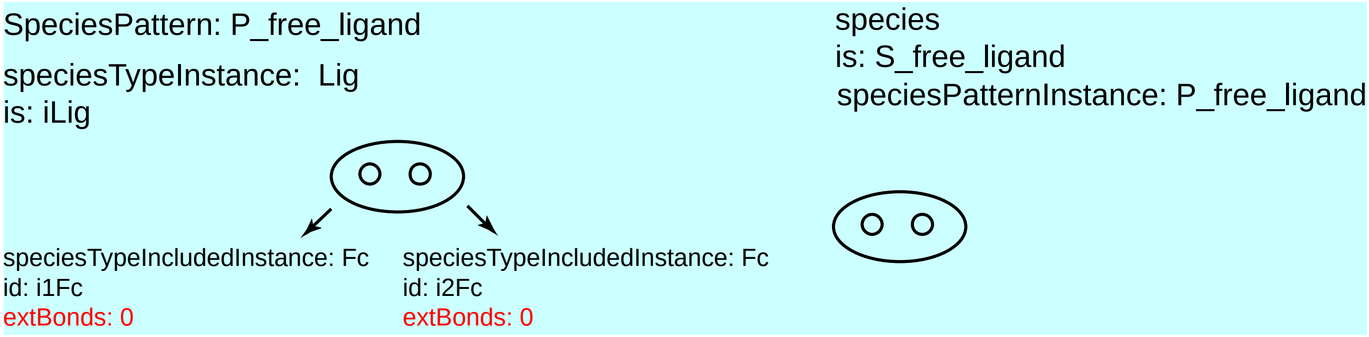
Species use is compatible to SBML L2. Species are uniquely defined chemical entities that may have compartment, initial amount, boundary condition etc. However, the new feature of species is an internal structure – a species consists of fully defined and connected species types declared by speciesTypesIncluded. Species can be viewed as graphs with all participating speciesTypes being fully defined. All unspecified states have default value.

speciesPattern is a key feature of this SBML Level 3 proposal. SpeciesPattern can be declared as sub-graph consisting of speciesTypes declared by speciesTypeInstances and speciesTypeIncludedInstances. speciesTypeInstance and speciesTypeIncludedInstances do not need to be fully defined – if some speciesTypesIncluded are omitted, or some speciesTypeStates are omitted – they assume all possible values.

speciesTypeInstance, speciesTypeIncludedInstance	Comments
id : SId	
name : string {use="optional"}	
multiplicity: int { minInclusive="0" use="optional" default="1"}	
extBonds: int { minInclusive="0" maxInclusive="maxExternalBonds" use="optional"}	Number of external bonds. If not used (default), any number of bonds up to maxExternalBonds can be used. It means that speciesTypeIncluded can be either connected or not connected to speciesTypes outside the enclosing speciesType. Value 1 with default maxExternalBonds 1 means that this speciesTypeInstance must be connected to something.
intBonds: int { minInclusive="0" maxInclusive="maxInternalBonds" use="optional"}	Number of internal bonds. If not used (default), any number of bonds up to maxExternalBonds can be used. It means that speciesTypeIncluded can be either connected or not connected to speciesTypes outside the enclosing speciesType.

4.1 Example: speciesPattern declared by a single speciesType with no connectivity

The speciesPattern “P_free_ligand” includes a species type Lig with all speciesTypesInstances being unbound.


SpeciesPattern: P_free_ligand
speciesTypeInstance: Lig
is: iLig
speciesTypeIncludedInstance: Fc
id: i1Fc
extBonds: 0
speciesTypeIncludedInstance: Fc
id: i2Fc
extBonds: 0
species
is: S_free_ligand
speciesPatternInstance: P_free_ligand

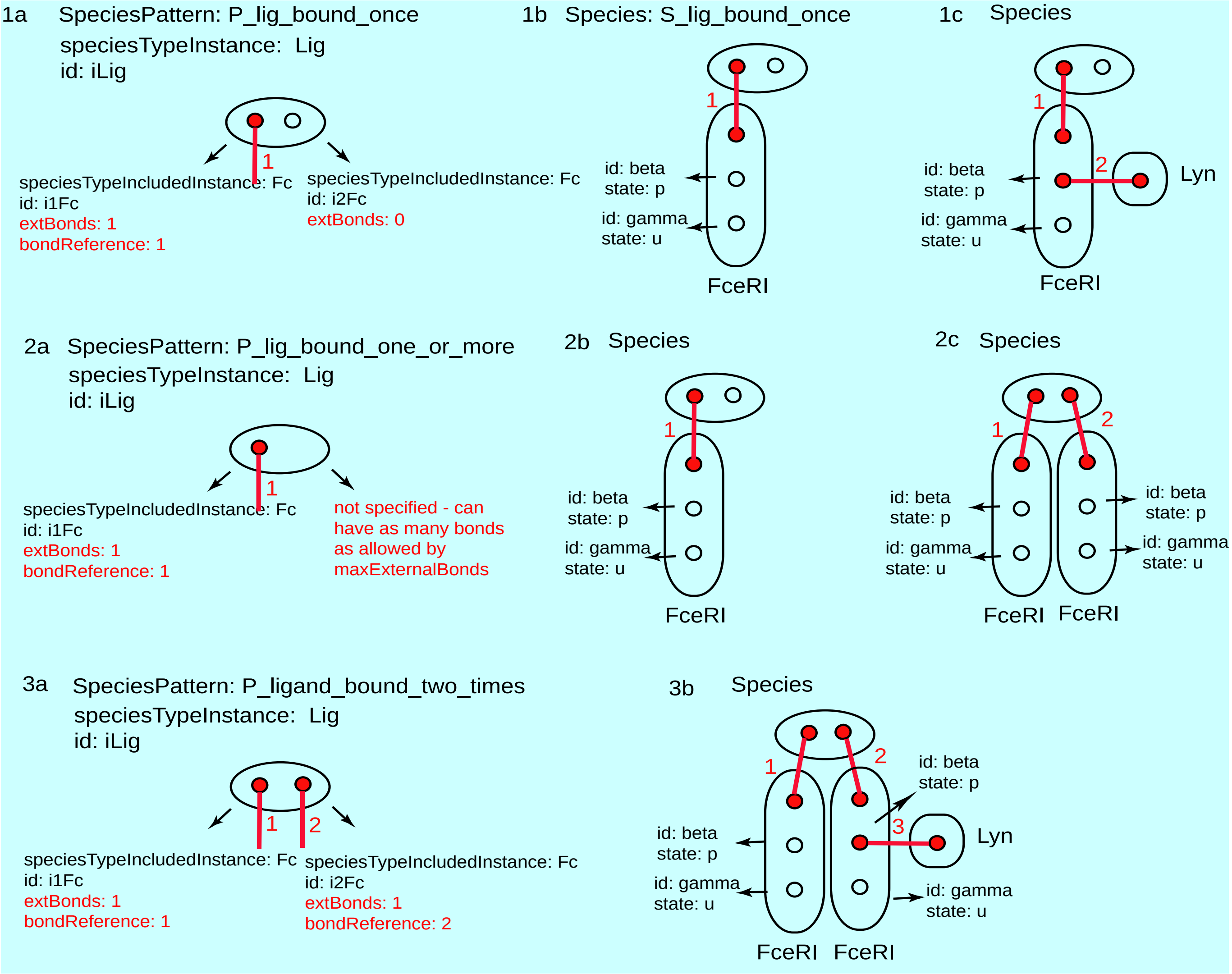
```
<speciesPattern id="P_free_ligand">
    <listOfSpeciesTypeInstances>
        <speciesTypeInstance speciesType="Lig" id="iLig">
           <listOfSpeciesTypeIncludedInstances>
                <speciesTypeIncludedInstance id="i1Fc" speciesTypeIncluded="Fc" extBonds="0"/>
                <speciesTypeIncludedInstance id="i2Fc" speciesTypeIncluded="Fc" extBonds="0"/>
            </listOfSpeciesTypeIncludedInstances>
        </speciesTypeInstance >
    </listOfSpeciesTypeInstances>
</speciesPattern>
```

The speciesPattern “T_free_ligand” includes a single species Lig, because all potential binding sites are unbound. Please note that if we can specify direct binding to Lig, this speciesPattern may consist of more than one species. There are two ways to declare a species: as a reference to speciesPattern (by default all unspecified externalBonds are zeros) or as a reference to a speciesType:

```
<species id="S_free_ligand">
      <speciesPatternInstance speciesPattern="T_free_ligand" id="iLig"/>
</species>

<species id="S_free_ligand">
    <listOfSpeciesTypeInstances>
        <speciesTypeInstance speciesType="Lig" id="iLig">
    </listOfSpeciesTypeInstances>
</species>
```

4.2 *Example: speciesPattern declared by a single speciesType with multiple external bonds*



Example 1a A speciesPattern selecting species that contain a ligand bound via a single Fc domain. Effectively, this speciesPattern selects monomers.

```
<speciesPattern id="P_ lig_bound_once">
    <listOfSpeciesTypeInstances>
        <speciesTypeInstances speciesType="Lig" id="iLig">
           <listOfSpeciesTypesIncludedInstances>
             <speciesTypeIncludedInstance id="i1Fc" speciesTypeIncluded="Fc" extBonds="1">
                        <listOfBondReferences>
                            <bondReference bond="1"/>
                        <listOfBondsReferences>
             </speciesTypeIncludedInstance>
             <speciesTypeIncludedInstance id="i2Fc" speciesTypeIncluded="Fc" extBonds="0"/>
           <listOfSpeciesTypeIncludedInstances>
        </speciesTypesInstance>
    </listOfSpeciesTypeInstances>
     <listOfBonds>
         <bond id="1" bondType="external"/>
     </listOfBonds>
</speciesPattern>
```

Example 1b A species matched by speciesPattern 1a.

```
<species id="S_ lig_bound_once">
    <listOfSpeciesTypeInstances>
        <speciesTypeInstance speciesType="Lig" id="iLig">
           <listOfSpeciesTypeIncludedInstances>
                 <speciesTypeIncludedInstance id="i1Fc" speciesTypeIncluded="Fc" extBonds="1">
                        <listOfBondReferences>
                            <bondReference bond="1"/>
                        <listOfBondsReferences>
                 </speciesTypeIncludedInstance>
                <speciesTypeIncludedInstance id="i2Fc" speciesTypeIncluded="Fc" extBonds="0"/>
           </listOfSpeciesTypeIncludedInstances>
        </speciesTypeInstance>
        <speciesTypesInstance speciesType="FceRI" id="iFceRI">
            <listOfSpeciesTypeIncludedInstances>
                <speciesTypeIncludedInstance id="ialpha" speciesType="alpha" extBonds="1">
                        <listOfBondReferences>
                            <bondReference bond="1"/>
                        <listOfBondsReferences>
                 </speciesTypeIncludedInstance>
                <speciesTypeIncludedInstance id="ibeta" speciesType="beta" speciesTypeState="p"
extBonds="1"/>
            </listOfSpeciesTypeIncludedInstances>
        </speciesTypeInstance>
    </listOfSpeciesTypeInstances>
     <listOfBonds>
         <bond id="1" bondType="external"/>
     </listOfBonds>
```

```
</speciesPattern>
```

Example 2a A speciesPattern selecting species that contain a ligand bound via at least a single Fc component (but may be two), effectively selecting all ligands bound to receptors:

```
<speciesPattern id="P_ lig_bound_one_or_more">
    <listOfSpeciesTypeInstances>
        <speciesTypeInstance speciesType="Lig" id="iLig">
            <listOfSpeciesTypeIncludedInstances>
              <speciesTypeIncludedInstance id="i1Fc" speciesTypeIncluded="Fc" externalConnectivity="1">
                          <listOfBondReferences>
                               <bondReference bond="1"/>
                          <listOfBondsReferences>
              </speciesTypeIncludedInstance >
              <speciesTypeIncludedInstance id="i2Fc" speciesTypeIncluded="Fc"/>
            </listOfSpeciesTypeIncludedInstances>
        <speciesTypeInstance>
    </listOfSpeciesTypeInstances>
     <listOfBonds>
          <bond id="1" bondType="external"/>
     </listOfBonds>
</speciesPattern>
```

Example 3a A speciesPattern selecting species that contain a ligand bound via both Fc component (effectively selecting dimers):

```
<speciesPattern id="P_ lig_bound_twice">
    <listOfSpeciesTypeInstances>
        <speciesTypeInstance speciesType="Lig" id="iLig">
            <listOfSpeciesTypeIncludedInstances>
              <speciesTypeIncludedInstance id="i1Fc" speciesTypeIncluded="Fc" externalConnectivity="1">
                          <listOfBondReferences>
                               <bondReference bond="1"/>
                          <listOfBondsReferences>
              </speciesTypeIncludedInstance >
              <speciesTypeIncludedInstance id="i2Fc" speciesTypeIncluded="Fc" externalConnectivity="1">
                          <listOfBondReferences>
                               <bondReference bond="1"/>
                          <listOfBondsReferences>
              </speciesTypeIncludedInstance >
            </listOfSpeciesTypeIncludedInstances>
          </speciesTypeInstance>
        </listOfSpeciesTypeInstances>
     <listOfBonds>
          <bond id="1" type="external"/>
     </listOfBonds>
</speciesPattern>
```

5 Specification of observables

Sometimes we need several speciesPatterns to select multiple species corresponding to a specific experimental observable. It is done through observable:

```
  <observable id="Receptor-phosph">
    <listOfSpecies>
      <speciesReference species="S1 "/>
      <speciesReference species="S2"/>
    </listOfSpecie>
    <listOfSpeciesPatterns>
      <speciesPatternReference speciesPattern="Receptor-beta-phosph"/>
      <speciesPatternReference speciesPattern="Receptor-gamma-phosph"/>
    </listOfSpeciesPatterns>
  </observable>
```

6 Specification of reaction and reaction rules

Reaction rules are used to generate chemical reactions from a list of chemical species by identifying sets of reactants and products.

Applying a reaction to a set of chemical species consists of the following steps:

- identify the group of species corresponding to each reactant in speciesType;
- for each combination of reactant species drawn from these groups, the rule is applied by replacing states, connectivities and bonds in reactant species with the corresponding states, connectivities and bonds in product species to define the products. In carrying out this replacement, states, connectivities and bonds that are not specified are inherited in product species from reactant species.
- the product species are then checked against the current list of chemical species and added to the list if they are not already present. The generated reaction can also be checked against the list of previously generated reactions to prevent duplication of reactions or to identify overlap between rules.

The main change: reactants and products are selected not as speciesReference, but as speciesPatternReference. Thus, if a reaction includes speciesPatternReference in place of speciesReference, it is automatically a rule. Note that if a speciesPatternReference selects a single species, then the containing reaction is as in SBML L2.

If speciesPatterns P_monomer, P_free_ligand" and P_lig_monomer are previously defined, ligand-binding reaction may be written in the following form (for clarity of XML examples, we describe only listOfReactants and listOfProducts; kineticLaw is omitted and will be considered later):

```
<reaction id="Ligand_bind" reversible="true">
    <listOfReactants>
        <speciesPatternReference speciesPattern="P_monomer"/>
        <speciesPatternReference speciesPattern="P_free_ligand"/>
    </listOfReactants>
    <listOfProducts>
        <speciesPatternReference speciesType="P_lig_monomer"/>
    </listOfProducts>
```

</reactionRule>

However, often we need to specify speciesPatterns within reaction rule, as below.