Systems Biology Graphical Notation: Process Description language Level 1

Date: November 20, 2014

User Manual

Chapter 1

Introduction

With the rise of systems and synthetic biology, the use of graphical representations of pathways and networks to describe biological systems has become pervasive. It was therefore important to use a consistent notation that would allow people to interpret those maps easily and quickly, without the need of extensive legends. Furthermore, distributed investigation of biological systems in different labs as well as activities like synthetic biology, that reconstruct biological systems, need to exchange their descriptions unambiguously, as engineers exchange circuit diagrams.

The goal of the Systems Biology Graphical Notation (SBGN) is to standardize the graphical/visual representation of biochemical and cellular processes. SBGN defines comprehensive sets of symbols with precise semantics, together with detailed syntactic rules defining their use. It also describes the manner in which such graphical information should be interpreted. SBGN is made up of three different and complementary languages [**?**]. This document presents the graphical elements composing the *Process Description language* of SBGN. It is not a normative description, but rather a document aimed at end users. It will provide examples of biological processes and there representation using SBGN Process Description language, and will give a general overview of the expressiveness of this language. People, such as software developers, looking for a normative description of SBGN Process Descriptions should rather read the technical specification of the language [**?**].

1.1 Overview of SBGN Process Descriptions

To quickly describe what SBGN Process Description language is about, let's give a brief overview of some of the relevant concepts with the help of an example shown in Figure **??**. It is a simple map for part of a mitogen-activated protein kinase (MAPK) cascade. The larger nodes in the figure (some of which are in the shape of rounded rectangles and others in the shape of circles) represent biological materials—things like macromolecules and simple chemicals (NB: the nodes representing physical entities (or proxies to physical entities) will always be colored in yellow in this document. Color is not part of the SBGN specification though). The biological materials are altered via processes (colored in green in this document), which are indicated in Process Description language by lines with arrows and other decorations. In this particular map, all of the processes happen to be the same: processes catalyzed by biochemical entities. The directions of the arrows indicate the direction of the processes; for example, unphosphorylated RAF kinase proceeds to phosphorylated RAF kinase via a process catalyzed by RAS. Although ATP and ADP are shown as incidental to the phosphorylations on this particular graph, they are involved in the same process as the proteins getting phosphorylated. The small circles on the nodes for RAF and other entity pools represent state variables (in this case, phosphorylation sites).

The essence of the Process Descriptions is *change*: it shows how different entities in the system process from one form to another. The entities themselves can be many different things. In the example of Figure **??**, they are either pools of macromolecules or pools of simple chemicals, but as will become clear later in this chapter, they can be other conceptual and material constructs as well. Note also that we speak of *entity pools* rather than individuals; this is because in biochemical network

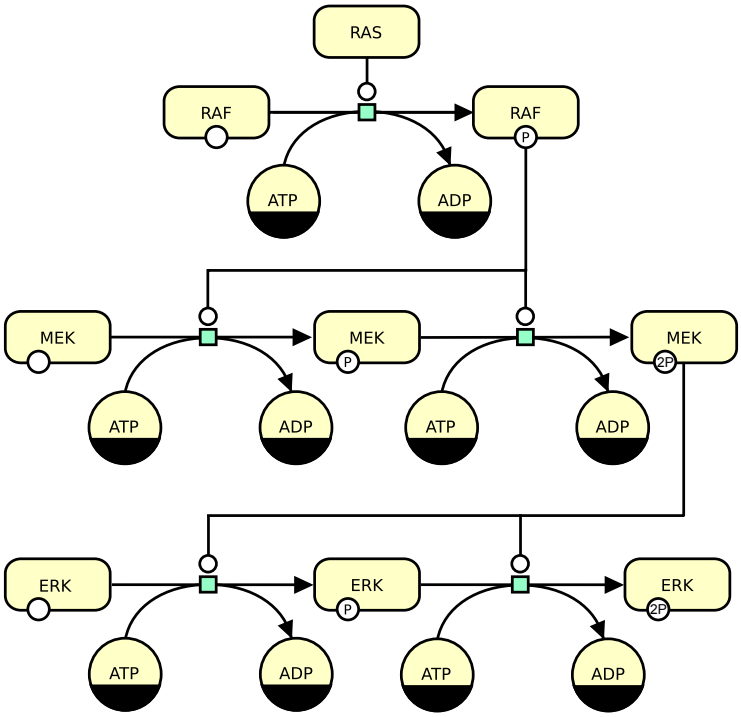


Figure 1.1: *This example of a Process Description uses two kinds of entity pool nodes: one for pools of different macromolecules (Section* **??***) and another for pools of simple chemicals (Section* **??***). Most macromolecule nodes in this map are adorned with state variables (Section* **??***) representing phosphorylation states. This map uses one type of process node, the process node (Section* **??***), and three kind of connecting arc, consumption (Section* **??***), production (Section* **??***) and catalysis (Section* **??***). Finally, some entity pool nodes have dark bands along their bottoms; these are clone markers (Section* **??***) indicating that the same pool nodes appear multiple times in the map.*

models, one does not focus on single molecules, but rather collections of molecules of the same kind. The molecules in a given pool are considered indistinguishable from each other. The way in which one type of entity is transformed into another is conveyed by a *process node* and arcs between entity pool nodes and process nodes indicate an influence by the entities on the processes. In the case of Figure **??**, those arcs describe consumption (Section **??**), production (Section **??**) and catalysis (Section **??**), but others are possible. Finally, nodes in Process Descriptions are usually not repeated; if they do need to be repeated, they are marked with *clone markers*—specific modifications to the appearance of the node (Section **??**). The details of this and other aspects of Process Description notation are explained in the rest of this chapter.

A reference card depicting all the symbols of SBGN Process Description Level 1 is present at the end of this document.

Lets look at a few additional examples which show typical biological processes and their SBGN Process Description representation. In Figure **??** a reversible reaction with two substrates and one product is shown. The enzyme E catalyzes an irreversible (metabolic) process which consumes two substrates (S1 and S2) and produces one product (P1). The enzyme is a protein, therefore represented as a *macromolecule*. Substrates and product of the biochemical reaction are represented by *simple chemicals*. The consumption of S1 and S2 is represented by the *consumption arcs*. The *production arc* represents the synthesis of P1.

In Figure **??** the formation of a complex is shown. Two *macromolecule* entities X and Y form the *complex* X_Y. Complex formation is represented using the *association* process node with ingoing *consumption* and outgoing *production* arcs. The *complex* glyph surrounds subunits X and Y.

In Figure **??** the regulation of a target gene by a transcription factor without knowledge about the promoter binding is shown. A transcription factor (TF) protein together with a target gene promoter X triggers the*process* of transcription. Direct binding of the TF to the target gene promoter has not been experimentally verified, therefore the *logical operator* AND is used to describe the yet unspeci-

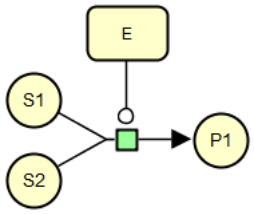


Figure 1.2: *This example of a Process Description shows an irreversible catalysis with 2 substrates and 1 products.*

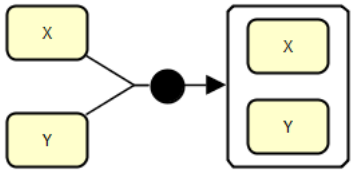


Figure 1.3: *This example of a Process Description shows an irreversible catalysis with 2 substrates and 1 products.*

fied interaction between TF and target gene. The TF protein is a *macromolecule* of the *material type* 'protein' (mt:prot) whereas the gene promoter is given as a *nucleic acid feature* with the *conceptual type* 'gene' (ct:gene). The connecting arc *necessary stimulation* is applied to indicate that the stimulation by both regulator and target is necessary for the transcription process to take place. The target gene messenger as a product of the transcription process is represented by a *nucleic acid feature* with the *conceptual type* 'mRNA' (ct:mRNA). The *unspecified source* symbol is used to represent the large number of substrates of a transcription process (i.e. trinucleotides).

A last example is show in Figure **??**, which shows passive transport or diffusion of a molecule. The *macromolecule* X in the cytosol serves as the substrate of a process leading to the production of the *macromolecule* X in the nucleus. This process describes the passive transport of X from one *compartment* to the other. The two macromolecules X do not carry the clone marker because the containing compartment is part of their identity.

More examples can be found in a list of so called SBGN bricks [**?**], which are building blocks representing basic biological patterns. These bricks can be used for assembly into different kinds of biological networks such as metabolic and regulatory networks.

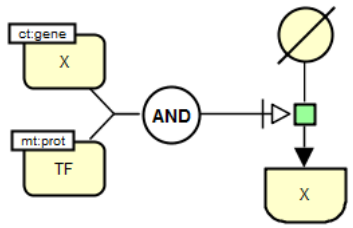


Figure 1.4: *This example of a Process Description shows a regulation of a target gene by a transcription factor without knowledge about the promoter binding.*

1.2 SBGN levels and versions

It was clear at the outset of SBGN development that it would be impossible to design a perfect and complete notation right from the beginning. Apart from the prescience this would require (which, sadly, none of the authors possess), it also would likely need a vast language that most newcomers would shun as being too complex. Thus, the SBGN community followed an idea used in the devel-

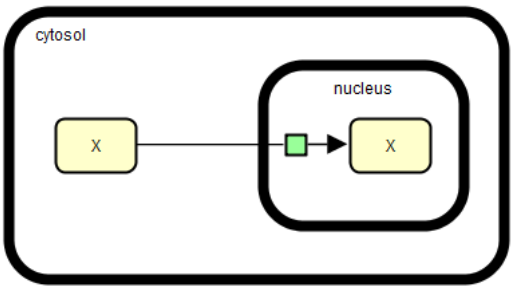


Figure 1.5: *This example of a Process Description shows a passive transport of a molecule.*

opment of other standards, i.e. stratify language development into levels.

A *level* of one of the SBGN languages represents a set of features deemed to fit together cohesively, constituting a usable set of functionality that the user community agrees is sufficient for a reasonable set of tasks and goals. Within *levels*, *versions* represent small evolution of a language, that may involve new glyphs, refined semantics, but no fundamental change of the way maps are to be generated and interpreted. In addition new versions should be backwards compatible, i.e., Process Description maps that conform to an earlier version of the Process Description language within the same level should still be valid. This does not apply to new levels.

Capabilities and features that cannot be agreed upon and are judged insufficiently critical to require inclusion in a given level, are postponed to a higher level or version. In this way, the development of SBGN languages is envisioned to proceed in stages, with each higher levels adding richness compared to the levels below it.

1.3 How to get more information

This user manual will present the various symbols used by SBGN Process Description Level 1 (Chapter **??**), and provide guidance to design SBGN Process Description maps (Chapter **??**). The authors tried to keep the presentation simple, and to avoid being too technical.

The normative description of the language is the technical specification [**?**]. It is available from the SBGN website (`http://sbgn.org/`). This website is a portal for all things to the notation. In addition to the specifications, there are examples of maps, FAQs, and informations on part and forthcoming meetings.

The easiest and best way to get involved in SBGN discussions is to join the sbgn-discuss@caltech.edu mailing list. If you only want the announcements of meetings and new specifications, you can join the very low flux mailing list sbgn-announce@lists.sf.net instead.

Chapter 2

Symbols used in SBGN Process Descriptions

An SBGN Process Description map is mainly a bipartite graph, i.e. it is made up of two types of nodes that connect in an alternate way (some exceptions are described below, e.g. when *logical operators* or *tag* are used). The two types of nodes are the *process nodes* and the *entity pools nodes*, the later representing the things that are modified by processes. These nodes are connected by arcs. In addition, the *entity pools nodes* can be contained in *compartments*.

2.1 Entity pool nodes

An entity pool is a population of entities that cannot be distinguished from each other, when it comes to the SBGN Process Description Level 1 map. For instance all the molecular entities that fulfill the same role in a given process form an entity pool. As a result, an entity pool can represent different granularity levels, such as all the proteins, all the instances of a given protein, only certain forms of a given protein. It really depends on what we want to represent. To belong to different compartments is sufficient to belong to different entity pools. Calcium ions in the endoplasmic reticulum and calcium ions in the cytosol belong to different entity pools when it comes to representing calcium release from the endoplasmic reticulum.

The Process Description language contains six glyphs representing classes of material entities: *unspecified entity* (Section **??**), *simple chemical* (Section **??**), *macromolecule* (Section **??**), *nucleic acid feature* (Section **??**), and *complex* (Section **??**). (Specific types of macromolecules, such as protein, RNA, DNA, polysaccharide, and specific simple chemicals are not defined by Process Description but may be part of future levels of SBGN). In addition to the material entities, the Process Description language represents two conceptual entities: An absorbing pool, called *source and sink* (Section **??**), and a *perturbing agent* (Section **??**). Material and conceptual entities can optionally carry auxiliary units such as *units of information* (Section **??**), *state variables* (Section **??**) and *clone markers* (Section **??**).

2.1.1 Glyph: *Unspecified entity*

The simplest type of EPN is the *unspecified entity*: one which type is unknown or simply not relevant to the purposes of the map. This arises, for example, when the existence of the entity has been inferred indirectly, or when the entity is merely a construct introduced for the needs of a map, without direct biological equivalent. For cases where the identity of the entities composing the pool *is* known, there exist other, more specific glyphs described below in the manual.

An *unspecified entity* is represented by an elliptic container, as shown in Figure **??**. Note that this must remain an ellipse to avoid confusion with the Simple Chemical glyph, which is a circle (c.f. **??**).

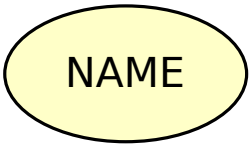


Figure 2.1: *The Process Description glyph for* unspecified entity.

2.1.2 Glyph: *Macromolecule*

Many biological processes involve *macromolecules*: biochemical substances that are built up from the covalent linking of pseudo-identical units. Examples of macromolecules include proteins, nucleic acids (RNA, DNA), and polysaccharides (glycogen, cellulose, starch, etc.). Attempting to define a separate glyph for all of these different molecules would lead to an explosion of symbols in SBGN, so instead, SBGN Process Description Level 1 defines only one glyph for all macromolecules. The same glyph is to be used for a protein, a nucleic acid, a complex sugar, and so on. The exact nature of a particular macromolecule in a map is then clarified using its label and decorations, as will become clear below. A *macromolecule* is represented by a rectangular container with rounded corners, as illustrated in Figure **??**.

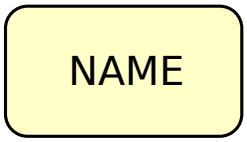


Figure 2.2: *The Process Description glyph for* macromolecule.

Examples of *macromolecules* are presented in Figure **??**.

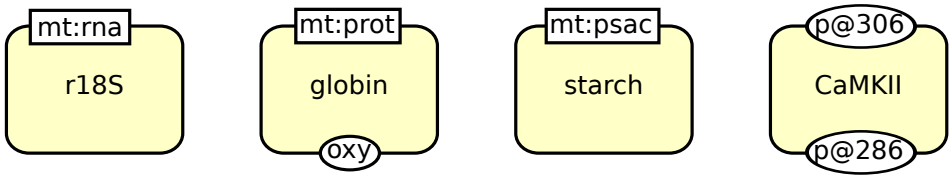


Figure 2.3: *Examples of* macromolecules. *From left to right: the macromolecule of 18S ribosomal RNA, globin (a protein) in the oxygenated state, a molecule of starch (polymer of glucose), calcium calmodulin kinase 2 phosphorylated on threonine 286 and 306.*

2.1.3 Glyph: *Simple chemical*

In SBGN Process Descriptions, a simple chemical is defined as the opposite of a macromolecule (Section **??**): it is a chemical compound that is *not* formed by the covalent linking of pseudo-identical residues. Examples of simple chemicals are an atom, a monoatomic ion, a salt, a radical, a solid metal, a crystal, etc. A *simple chemical* is represented by a circular container, as depicted in Figure **??**. To avoid confusion with the Unspecified Entity (**??**), this glyph must remain a circle and cannot be deformed into an ellipse.

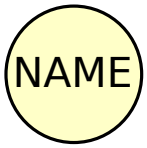


Figure 2.4: *The Process Description glyph for* simple chemical.

Examples of *simple chemicals* are presented in Figure **??**.

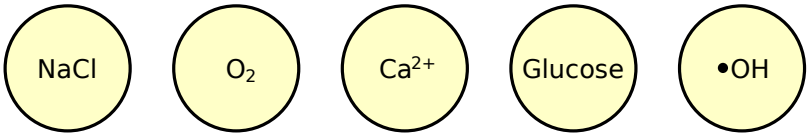


Figure 2.5: *Examples of* simple chemicals. *From left to right: sodium chloride (a salt), dioxygene (a elemental molecule), calcium ion, glucose (an heteroatomics molecule), hydroxyl radical.*

2.1.4 Glyph: *Nucleic acid feature*

The *Nucleic acid feature* construct in SBGN is meant to represent a fragment of a macromolecule carrying genetic information. A common use for this construct is to represent a gene or a transcript. The label of this EPN and its *units of information* (see Section **??**) are often important for making the purpose clear to the reader of a map. A *nucleic acid feature* is represented by a rectangular container whose bottom half has rounded corners, as shown in Figure **??**. This design reminds that we are fundamentally dealing with a unit of information, but this information is carried by a macromolecule.

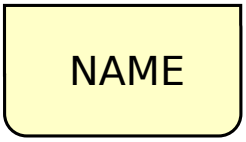


Figure 2.6: *The Process Description glyph for* nucleic acid feature.

Examples of *nucleic acid features* are presented in Figure **??**.

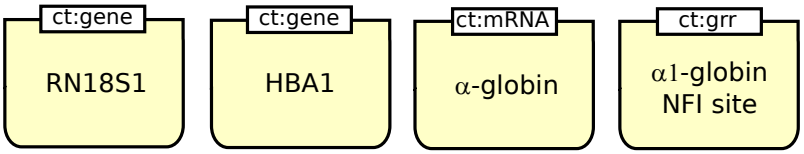


Figure 2.7: *Examples of* nucleic acid features. *From left to right: gene coding for the 18S ribosomal RNA, gene coding for α1-globin, messenger RNA coding for α-globin, nuclear factor 1 binding site on the promoter of α1-globin gene.*

2.1.5 Glyph: *Complex*

A *complex* node represents a biochemical entity composed of other biochemical entities, whether macromolecules, simple chemicals, multimers, or other complexes. The resulting entity may have its own identity, properties and function in an SBGN map. A *complex* possesses its own container box surrounding the juxtaposed container boxes of its components. This container box is a rectangle with cut-corners (an octagonal box with sides of two different lengths). The size of the cut-corners

are adjusted so that there is no overlap between the container and the components. The container boxes of the components must not overlap.

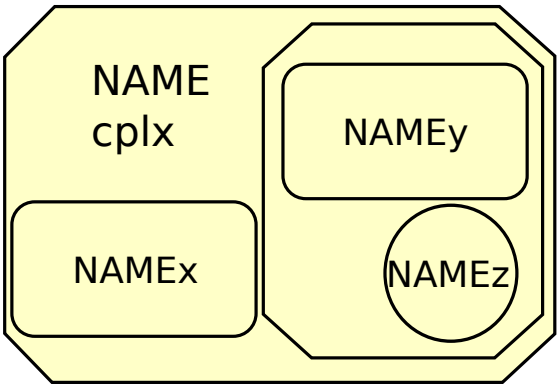


Figure 2.8: *The Process Description glyph for* complex.

Examples of *complexes* are presented in Figure **??**.

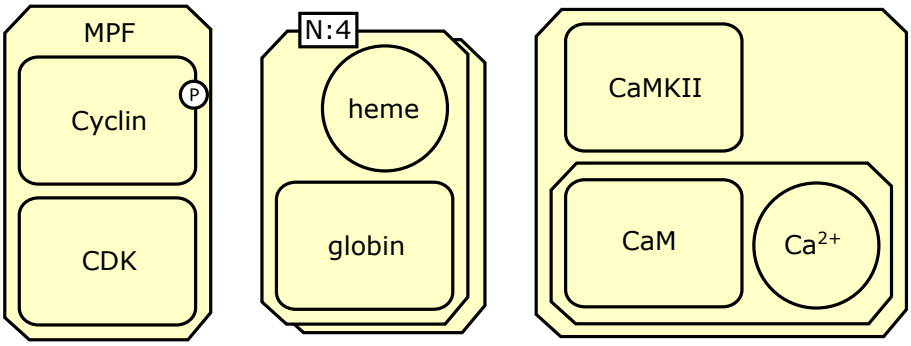


Figure 2.9: *Examples of* complexes. *From left to right: complex between phosphorylated cyclin and CDK2 forming the maturation promoting factor in yeast, a tetramer of complexes between globin and heme, and a complex between calcium-calmodulin kinase II and another complex, itself formed of calmodulin and calcium.*

2.1.6 Glyph: *Source* and *Sink*

It is useful to have the ability to represent the creation of an entity or a state from an unspecified source, that is, from something that one does not need or wish to make precise. For instance, in a

model where the production of a protein is represented, it may not be desirable to represent all of the amino acids, sugars and other metabolites used, or the energy involved in the protein's creation. Similarly, we may not wish to bother representing the details of the destruction or decomposition of some biochemical species into a large number of more primitive entities, preferring instead to simply say that the species "disappears into a sink". Yet another example is that one may need to represent an input (respectively, output) into (resp. from) a compartment without explicitly representing a transport process from a source (resp. to a target).

For these and other situations, SBGN defines a glyph for explicitly representing the involvement of an unspecified source or sink. A *source* or *sink* is represented by the mathematical symbol for "empty set", that is, a circle crossed by a bar linking the upper-right and lower-left corners of an invisible square drawn around the circle (∅). Figure **??** illustrates this. Each source or sink node should be linked to one and only one arc in a map. The symbol used in SBGN is borrowed from the mathematical symbol for "empty set", but it is important to note that it does not actually represent a true absence of everything or a physical void—it represents the absence of the corresponding structures in the model, that is, the fact that these sources or sinks are conceptually outside the scope of the map.

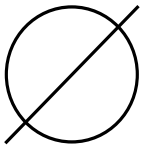


Figure 2.10: *The* source *and* sink *glyphs.*

2.1.7 Glyph: *Perturbing agent*

Biochemical networks can be affected by external influences. Those influences can be the effect of well-defined physical perturbing agents, such as a light pulse or a change in temperature; they can also be more complex and not well-defined phenomena, for instance the outcome of a biological process, an experimental setup, or a mutation. For these situations, SBGN provides the *perturbing agent* glyph. It is an EPN, and represents the amount to perturbing agent applied to a process. A *perturbing agent* is represented by a modified hexagon having two opposite concave faces, as illustrated in Figure **??**.

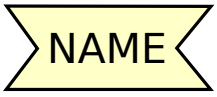


Figure 2.11: *The Process Description glyph for* perturbing agent.

2.1.8 Glyph: *Multimer*

As its name implies, a multimer is an aggregation of multiple identical or pseudo-identical entities held together by non-covalent bonds. Thus, they are distinguished from polymers by the fact that the latter involve covalent bonds, and should be represented by *macromolecules*. Here *pseudo-identical* refers to the possibility that the entities differ chemically but retain some common global characteristic, such as a structure or function, and so can be considered identical within the context of the SBGN Process Description. An example of this are the homologous subunits in a hetero-oligomeric receptor. SBGN Process Description accepts multimers of *simple chemical* (Section **??**), *macromolecule* (Section **??**), *nucleic acid feature* (Section **??**) or *complex* (Section **??**). A *multimer* is represented by two identical containers shifted horizontally and vertically and stacked one on top of the other as illustrated in Figure **??**.

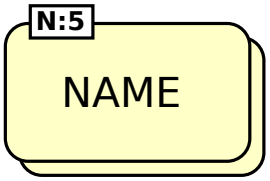


Figure 2.12: *The Process Description glyph for* multimer.

Examples of *multimers* are presented in Figure **??**.

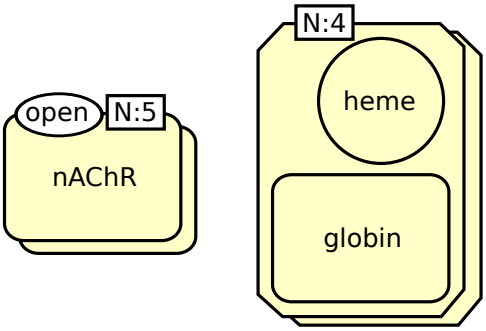


Figure 2.13: *Examples of* multimers. *From left to right: pentameric nicotinic receptor in the open state, tetramer of oxygenated globin.*

2.2 Decorations of the entity pool nodes

SBGN Process Description provides glyphs that decorate other glyphs, providing additional information that may be useful to the reader. These can provide annotation (*unit of information*), state information (*state variable*) or indicate duplication of entity pool nodes (*clone marker*).

2.2.1 Glyph: *Unit of information*

When representing biological entities, it is often necessary to convey some abstract information about the entity's function that is not related to its role in the map. The *unit of information* is a decoration that can be used in this situation to add information to an EPN. Some example uses include: characterizing a logical part of an entity such as a functional domain (a binding domain, a catalytic site, a promoter, etc.), or the information encoded in the entity (an exon, an open reading frame, etc.). A *unit of information* can also convey information about the physical environment, or the specific type of biological entity it is decorating. A *unit of information* is represented by a rectangle overlapping the border of the *EPN* being annotated.

The label carried by *unit of information* defines the information it carries. For certain predefined types of information having controlled vocabularies associated with them, SBGN defines specific prefixes that must be included in the label to indicate the type of information in question. The controlled vocabularies predefined in SBGN Process Description Level 1 are described in Section **??**.

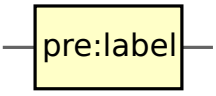


Figure 2.14: *The Process Description glyph for* unit of information.

2.2.2 Glyph: *State variable*

Many biological entities, such as molecules, can exist in different *states*, meaning different physical or informational configurations. These states can arise for a variety of reasons. For example, macromolecules can be subject to post-synthesis modifications, wherein residues of the macromolecules (amino acids, nucleosides, or glucid residues) are modified through covalent linkage to other chemicals. Other examples of states are alternative conformations as in the closed/open/desensitized conformations of a transmembrane channel, and the active/inactive forms of an enzyme.

SBGN provides a means of associating one or more *state variables* with an entity; each such variable can be used to represent a dimension along which the state of the overall entity can vary. When an entity can exist in different states, the state of the whole entity (i.e., the SBGN object) can be described by the current values of all its *state variables*, and the values of the *state variables* of all its possible components, recursively. A *state variable* is represented by an elliptical container overlapping the border of the *EPN* being annotated.

Figure 2.15: *The Process Description glyph for* state variable.

A *state variable* does not necessarily have to be Boolean-valued. For example, an ion channel can possess several conductance states; a receptor can be inactive, active and desensitized; and so on. As another example, a *state variable* "ubiquitin" could also carry numerical values corresponding to the number of ubiquitin molecules present in the tail. However, in all cases, a *state variable* on an EPN can only take *one* defined value. Further, an EPN's *state variable* should always be displayed and always set to a value. An "empty" *state variable* is a *state variable* that is set to the value "unset", it is not a *state variable* with no value. Note that the value "unset" is *not* synonymous to "any value" or "unknown value".

2.2.3 Glyph: *Clone marker*

It is sometimes necessary to represent the same *EPN* several times. Otherwise, the resulting graph is so tightly connected that the map becomes unreadable. An example would be the representation of currency molecules such as ATP. However, we must indicate the fact, so that a reader knows the processes involving this particular glyph are not the only processes involving the *EPN*. If an *EPN* is duplicated on a map, we therefore mark all its graphical reprensation with a *clone marker* auxiliary unit. This marker provides the reader with a visual indication that this node has been cloned, and that at least one other occurrence of the *EPN* can be found in the map (or in a submap; see Section **??**). The clone marker takes two forms, simple and labeled, depending on whether the node being cloned can carry state variables. Note that an *EPN* belongs to a single compartment. If two glyphs labelled "X" are located in two different compartments, such as ATP in cytosol and ATP in mitochondrial lumen, they represent different *EPNs*, and therefore do not need to be marked as cloned (and if they are, they are not part of the same clone).

The simple (unlabeled) *clone marker* is a portion of the surface of an *EPN* that has been modified visually through the use of a different shade, texture, or color. Figure **??** illustrates this. The *clone marker* occupies the lower part of the *EPN*. The filled area must be smaller than the unfilled one.

Figure 2.16: *The Process Description glyph for* simple clone marker *applied to a* simple chemical

Unlike the *simple clone marker*, the *labeled clone marker* includes (unsurprisingly, given its name) an identifying label that can be used to identify equivalent clones elsewhere in the map. This is particularly useful for stateful *EPNs*, because these can have a large number of state variables displayed and therefore may be difficult to visually identify as being identical. The filled area must be smaller than the unfilled one, but the be large enough to have a height larger than the *clone marker*'s label (cf below).

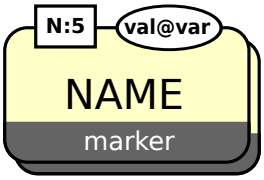


Figure 2.17: *The Process Description glyph for* labeled clone marker *applied to a* multimer *of* macromolecules.

2.2.4 Controlled vocabularies used in SBGN Process Description Level 1

Some glyphs in SBGN Process Descriptions can be enriched with particular kinds of textual annotations conveying information relevant to the purpose of the glyph. These annotations are *units of information* (Section **??**) or *state variables* (Section **??**). An example is the case of multimers, which can carry a *unit of information* conveying the number of monomers composing the multimer. Other cases are described throughout the rest of this chapter.

In the rest of this section, we describe the controlled vocabularies (CVs) used in SBGN Process Description Level 1. They cover the following categories of information: an EPN's material type, an EPN's conceptual type, covalent modifications on macromolecules, the physical characteristics of compartments, and cardinality (e.g., of multimers). In each case, some CV terms are predefined by SBGN. With the exception of covalent modifications, the controlled vocabulary terms contained in *units of information* or *state variables* must be prefixed to indicate the type of information being expressed. Authors may use other CV values not listed here, but in such cases, they should explain the term's meanings in a figure legend or other text accompanying the map.

Entity pool node material types

The material type of an EPN indicates its chemical structure. A list of common material types is shown in Table **??**, but others are possible. The values are to be taken from the Systems Biology Ontology ([**?**], http://www.ebi.ac.uk/sbo/), specifically from the branch *material entity* under *physical entity representation*. The labels are defined by SBGN Process Description Level 1.

Entity pool node conceptual types

An EPN's *conceptual type* indicates its function within the context of a given Process Description. In contrast to the *material types*, the *conceptual types* are not about physical composition, but about functional roles. For example, a strand of RNA is a physical artifact, but its use as messenger RNA is a role.

A list of common conceptual types is shown in Table **??**, but others are possible. The values are to be taken from the Systems Biology Ontology (http://www.ebi.ac.uk/sbo/), specifically from the branch *functional entity* under *physical entity representation*).

Name	Label	SBO term
Non-macromolecular ion	mt:ion	SBO:0000327
Non-macromolecular radical	mt:rad	SBO:0000328
Ribonucleic acid	mt:rna	SBO:0000250
Deoxribonucleic acid	mt:dna	SBO:0000251
Protein	mt:prot	SBO:0000297
Polysaccharide	mt:psac	SBO:0000249

Table 2.1: *A sample of values from the* material types *controlled vocabulary (Section* **??***).*

Name	Label	SBO term
Gene	ct:gene	SBO:0000243
Transcription start site	ct:tss	SBO:0000329
Gene coding region	ct:coding	SBO:0000335
Gene regulatory region	ct:grr	SBO:0000369
Messenger RNA	ct:mRNA	SBO:0000278

Table 2.2: *A sample of values from the* conceptual types *vocabulary (Section* **??***).*

Macromolecule covalent modifications

A common reason for the introduction of state variables (Section **??**) on an entity is to allow access to the configuration of possible covalent modification sites on that entity. For instance, a macromolecule may have one or more sites where a phosphate group may be attached; this change in the site's configuration (i.e., being either phosphorylated or not) may factor into whether, and how, the entity can participate in different processes. Being able to describe such modifications in a consistent fashion is the motivation for the existence of SBGN's covalent modifications controlled vocabulary.

Table **??** lists a number of common types of covalent modifications. The most common values are defined by the Systems Biology Ontology in the branch *addition of a chemical group*, under *occuring entity representation*. The labels shown in Table **??** are defined by SBGN Process Description Level 1; for all other kinds of modifications not listed here, the author of a Process Description must create a new label (and should also describe the meaning of the label in a legend or text accompanying the map).

Name	Label	SBO term
Acetylation	Ac	SBO:0000215
Glycosylation	G	SBO:0000217
Hydroxylation	OH	SBO:0000233
Methylation	Me	SBO:0000214
Myristoylation	My	SBO:0000219
Palmytoylation	Pa	SBO:0000218
Phosphorylation	P	SBO:0000216
Prenylation	Pr	SBO:0000221
Protonation	H	SBO:0000212
Sulfation	S	SBO:0000220
Ubiquitination	Ub	SBO:0000224

Table 2.3: *A sample of values from the* covalent modifications *vocabulary (Section* **??***).*

Physical characteristics

SBGN Process Description Level 1 defines a special unit of information for describing certain common physical characteristics. Table **??** lists the particular values defined by SBGN Process Description Level 1.

Name	Label	SBO term
Temperature	pc:T	SBO:0000147
Voltage	pc:V	SBO:0000259
pH	pc:pH	SBO:0000304

Table 2.4: *A sample of values from the* physical characteristics *vocabulary (Section* **??***).*

Cardinality

SBGN Process Description Level 1 defines a special unit of information usable on multimers for describing the number of monomers composing the multimer. Table **??** shows the way in which the values must be written. Note that the value is an positive non-zero integer, and not (for example) a range. There is at present no provision in SBGN Process Description Level 1 for specifying a range in this context because it leads to problems of entity identifiability.

Name	Label	SBO term
cardinality	N:#	SBO:0000364

Table 2.5: *The format of the possible values for the* cardinality *unit of information (Section* **??***). Here, # stands for the number; for example, "N:5".*

2.3 Process nodes

Process nodes represent processes that transform one or several entity pools into one or several entity pools, identical or different. SBGN Process Description Level 1 defines a generic *process* (Section **??**), as well as five more specific ones: the *omitted process* (Section **??**), the *uncertain process* (Section **??**), the *association* (Section **??**), the *dissociation* (Section **??**), and the *phenotype* (Section **??**).

2.3.1 Glyph: *Process*

A process is the basic process node in SBGN. It describes a process that transforms a given set of biochemical entities—macromolecules, simple chemicals or unspecified entities—into another set of biochemical entities. Such a transformation might imply modification of covalent bonds (conversion), modification of the relative position of constituents (conformational process) or movement from one compartment to another (translocation). A process transforms a set of entity pools (represented by *EPNs* in SBGN Process Description Level 1) into another set of entity pools. A *process* is represented by a square box linked to two connectors, small arcs attached to the centers of opposite sides. The consumption (Section **??**) and production (Section **??**) arcs are linked to the extremities of those connectors. The modulatory arcs (Section **??**) point to the other two sides of the box. A *process* connected to *production* arcs on opposite sides is a reversible process.

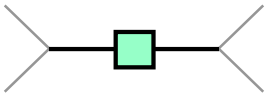


Figure 2.18: *The Process Description glyph for* process.

The example in Figure **??** illustrates the use of a *process* node to represent a reaction between two reactants that generates three products. The stoichiometry for each entity pool involved is 1, and therefore can be omitted.

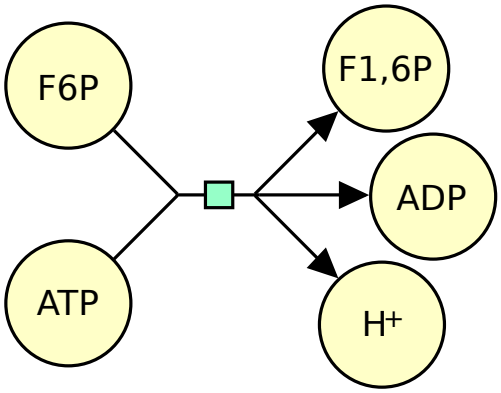


Figure 2.19: *Reaction between ATP and fructose-6-phosphate to produce fructose-1,6-biphosphate, ADP and a proton.*

The example in Figure **??** illustrates the use of a *process* node to represent the phosphorylation of a protein.

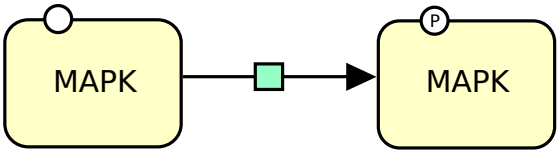


Figure 2.20: *Phosphorylation of the protein MAP kinase.*

The example in Figure **??** illustrates the use of a *process* node to represent a translocation. The large round-cornered rectangle represents a compartment border (see Section **??**).

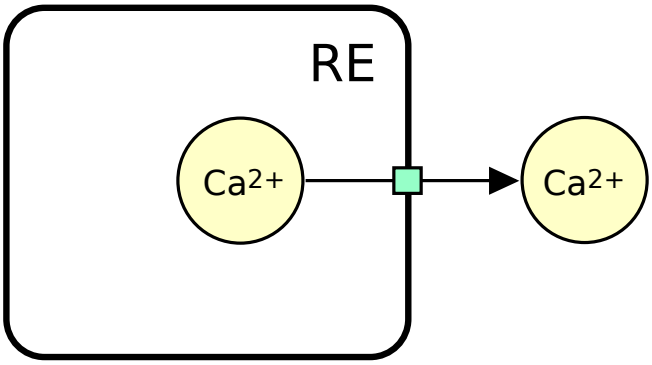


Figure 2.21: *Translocation of calcium ion out of the endoplasmic reticulum. Note that the* process *does not have to be located on the boundary of the* compartment. *A* process *is not attached to any* compartment.

The example in Figure **??** presents the conversion of two galactoses into a lactose. Galactoses are represented by only one *simple chemical*, the cardinality being carried by the *consumption* arc.

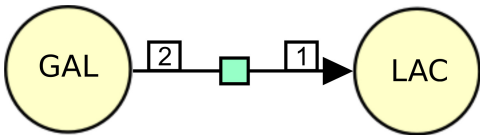


Figure 2.22: *Conversion of two galactoses into a lactose.*

2.3.2 Glyph: *Omitted process*

Omitted processes are processes that are known to exist, but are omitted from the map for the sake of clarity or parsimony. A single *omitted process* can represent any number of actual processes. For instance, one may want to represent a long chain of processes leading from one biochemical compound to another, without detailing all steps, but highlighting the fact that this is not a direct transformation. The *omitted process* is different from a *submap* (Section **??**). While a *submap* references to an explicit content, that is hidden in the main map, the *omitted process* does not "hide" anything within the context of the map, and cannot be "unfolded". An *omitted process* is represented by a *process* in which the square box contains a two parallel slanted lines oriented northwest-to-southeast and separated by an empty space.

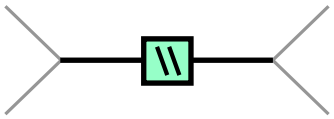


Figure 2.23: *The Process Description glyph for* omitted process.

2.3.3 Glyph: *Uncertain process*

Uncertain processes are processes that may not exist. A single *uncertain process* can represent any number of actual processes. *Uncertain process* would be used to represent hypothesis, reactions which existence is supported by weak evidence etc. An *uncertain process* is represented by a *process* which square box contains a question mark.

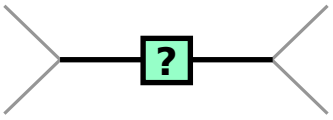


Figure 2.24: *The Process Description glyph for an* uncertain process.

2.3.4 Glyph: *Association*

The association between one or more *EPNs* represents the non-covalent binding of the biological objects represented by those *EPNs* into a larger complex. An *association* between several entities is represented by a filled disc linked to two connectors separated by 180 degrees. The consumption (Section **??**) and production (Section **??**) arcs are linked to the extremities of those connectors. An *association* is never reversible, the inverse process being represented by a *dissociation* (Section **??**).

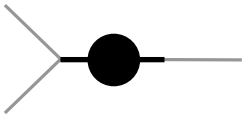


Figure 2.25: *The Process Description glyph for* association.

The example in Figure **??** illustrates the association of cyclin and CDC2 kinase into the Maturation Promoting Factor.

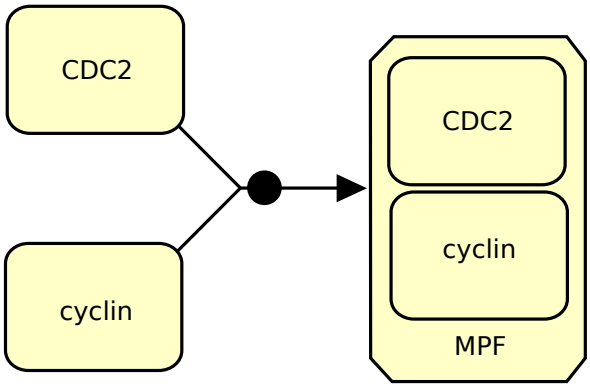


Figure 2.26: *Association of cyclin and CDC2 kinase into the Maturation Promoting Factor.*

An *association* does not necessarily involve components of the same nature. Figure **??** gives an example illustrating the association of a pentameric *macromolecule* (a nicotinic acetylcholine receptor) with a *simple chemical* (the local anesthetic chlorpromazin) in an unnamed complex.

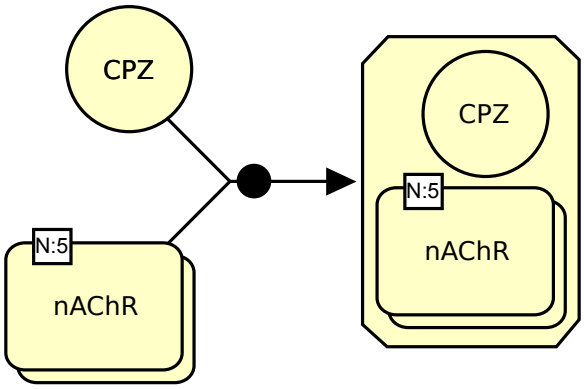


Figure 2.27: *The association of a pentameric macromolecule with a simple chemical in an unnamed complex.*

An association does not necessarily result in the formation of a *complex*; it can also produce a *multimer*. Figure **??** gives an example of using the successive formation of an hemoglobin monomer then a tetramer of the resulting complex.

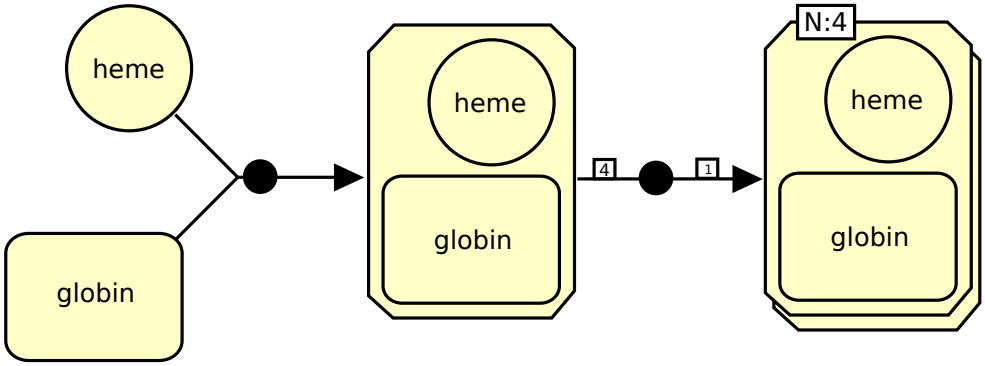


Figure 2.28: *Formation of hemoglobin.*

2.3.5 Glyph: *Dissociation*

The dissociation of an *EPN* into one or more *EPNs* represents the rupture of a non-covalent binding between the biological entities represented by those *EPNs*. A *dissociation* between several entities is represented by two concentric circles. A simple empty disc could be, in some cases, confused with the *catalysis* (section Section **??**). Moreover, the existence of two circles reminds the dissociation, by contrast with the filled disc of the *association* (Section **??**).

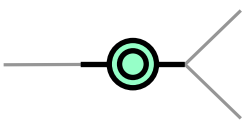


Figure 2.29: *The Process Description glyph for* dissociation.

2.3.6 Glyph: *Phenotype*

A biochemical network can generate phenotypes or affect biological processes. Such processes can take place at different levels and are independent of the biochemical network itself. To represent these processes in a map, SBGN defines the *phenotype* glyph, which describes a process consuming

nothing and producing nothing, but only modulated. A *phenotype* is represented by an hexagone, as illustrated in Figure **??**.

Figure 2.30: *The Process Description glyph for* phenotype.

The example in Figure **??** illustrates the use of a *phenotype* node to represent cell division, stimulated by the mono-phosphorylated form of the maturation promoting factor (see Section **??** for the meaning of the open arrowhead).

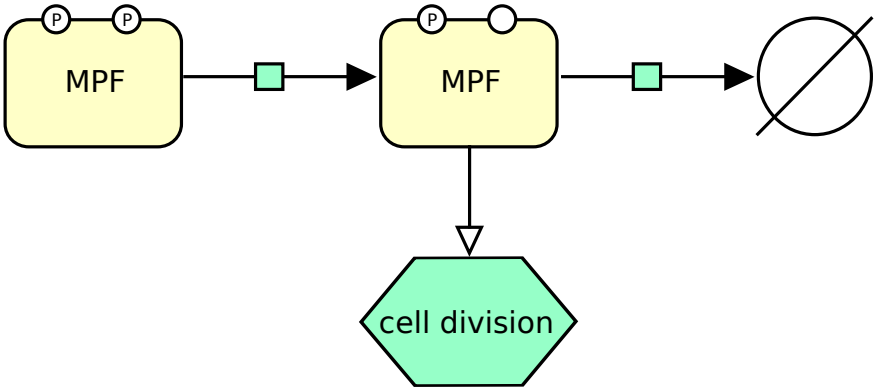


Figure 2.31: *Cell division stimulated by MPF.*

2.4 Arcs

Arcs are lines that link nodes of SBGN together. The symbols attached to their extremities indicate their meaning. SBGN Process Description Level 1 defines nine arcs. *consumption* (Section **??**), *production* (Section **??**), *modulation* (Section **??**), *stimulation* (Section **??**), *catalysis* (Section **??**), *inhibition* (Section **??**), and *necessary stimulation* (Section **??**) connect *EPNs* to *PDs*. *LogicArc* (Section **??**) link *EPNs* and *logic arcs*. *equivalenceArc* (Section **??**) link nodes to *tag*. Arcs can take any shape, and are not restricted to segments of straight lines.

2.4.1 Glyph: *Consumption*

Consumption is the arc used to represent the fact that an entity pool is consumed by a process, but is not produced by the process. A *consumption* is represented by a simple line without particular symbols at its extremities. A cardinality label may be associated with *consumption* (Section **??**) indicating the stoichiometry of the entity pool node for this process. This label is a number enclosed in a rectangle with one of the long sides adjacent to the consumption arc. Once assigned to one arc connecting to a process node, cardinality should be represented on all *consumption* and *production* arcs connected to that process node to avoid misinterpretation. In the case where the stoichiometry of some part of the process is not known, or undefined, a question mark (?) should be used within the cardinality label of the corresponding arcs.

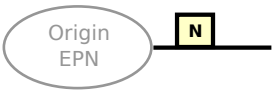


Figure 2.32: *The Process Description glyph for* consumption.

2.4.2 Glyph: *Production*

Production is the arc used to represent the fact that an entity pool is produced by a process. In the case of a reversible process, the *production* arc also acts as a *consumption* arc. The target extremity of a *production* carries a filled arrowhead. A cardinality label can be associated with a *production* arc indicating the stoichiometry of a process.

Figure 2.33: *The Process Description glyph for* production.

2.4.3 Glyph: *Modulation*

A modulation affects the flux of a process represented by the target process. Such a modulation can affect the process **positively or negatively**, or even both ways depending on the conditions, for instance the concentration of the intervening participants. A *modulation* can also be used when one does not know the precise direction of the effect. The target extremity of a *modulation* carries an empty diamond.

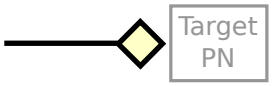


Figure 2.34: *The Process Description glyph for* modulation.

Figure **??** represents the effect of nicotine on the process converting closed and open states of a nicotinic acetylcholine receptor. High concentrations of nicotine open the receptor while low concentrations can desensitize it without opening.

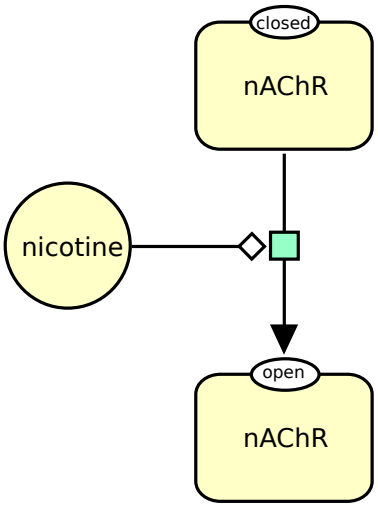


Figure 2.35: *Modulation of nicotinic receptor opening by nicotine.*

2.4.4 Glyph: *Stimulation*

A stimulation affects **positively** the flux of a process represented by the target process. This stimulation can be for instance a catalysis or a positive allosteric regulation. Note that *catalysis* exists independently in SBGN, see Section **??**. The target extremity of a *stimulation* carries an empty arrowhead.

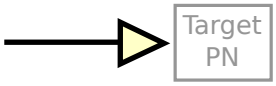


Figure 2.36: *The Process Description glyph for* stimulation.

The example in Figure **??** illustrates the use of two *stimulations* arcs to represent the opposite effects of agonists and inverse agonists on G-protein coupled receptor activity. Agonists stimulate the transition from inactive to active, while inverse agonists stimulate the transition inactive to active.

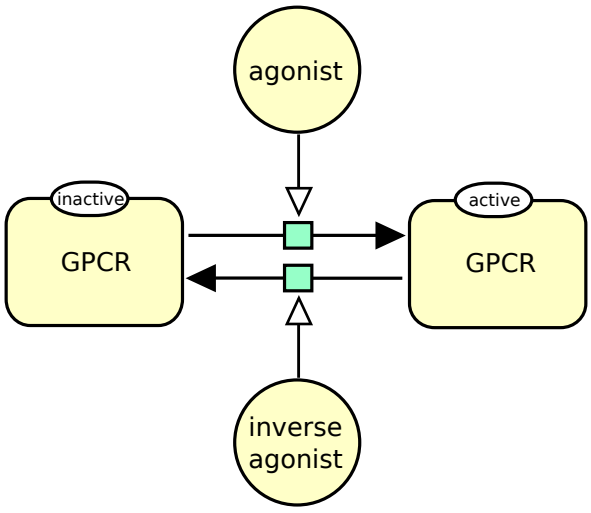


Figure 2.37: *Opposite effects of agonists and inverse agonists on GPCRs.*

2.4.5 Glyph: *Catalysis*

A catalysis is a particular case of stimulation, where the effector affects positively the flux of a process represented by the target process. The positive effect on the process is due to the lowering of the activation energy of a reaction. The target extremity of a *catalysis* carries an empty circle.

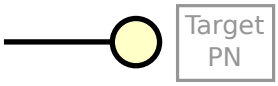


Figure 2.38: *The Process Description glyph for* catalysis.

The example in Figure **??** illustrates the use of *catalysis* arc to represent the effect of MAPKK on the phophorylation of MAPK.

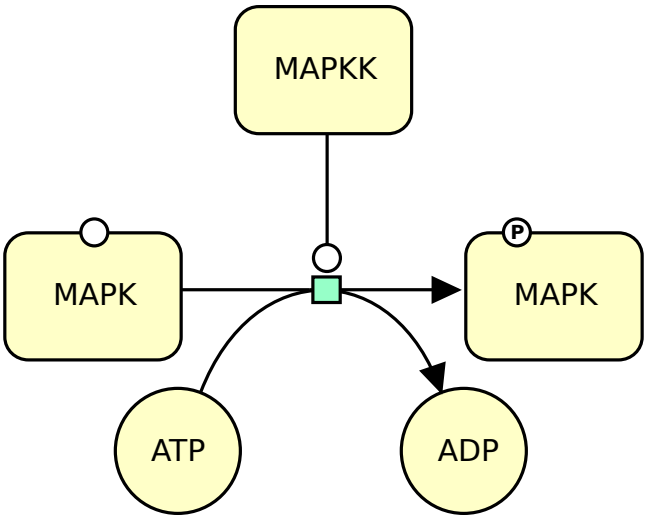


Figure 2.39: *MAPKK catalyses the phosphorylation of MAPK.*

2.4.6 Glyph: *Inhibition*

An inhibition **negatively** affects the flux of a process represented by the target process. This inhibition can be for instance a competitive inhibition or an allosteric inhibition. The target extremity of an *inhibition* carries a bar perpendicular to the arc.

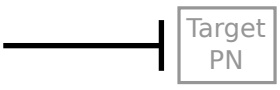


Figure 2.40: *The Process Description glyph for* inhibition.

2.4.7 Glyph: *Necessary stimulation*

A necessary stimulation, is one that is necessary for a process to take place. A process modulated by a necessary stimulation can only occur when this necessary stimulation is active. The target extremity of a *necessary stimulation* carries an open arrow (to remind that it is a *stimulation*) coming after a larger vertical bar.

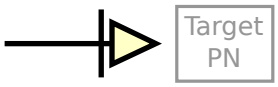


Figure 2.41: *The Process Description glyph for* Necessary Stimulation.

The example in Figure **??** below describes the transcription of a gene X, that is the creation of a messenger RNA X triggered by the gene X. The creation of the protein X is then triggered by the mRNA X. (Note that the same example could be represented using the gene as reactant and product, although it is semantically different.)

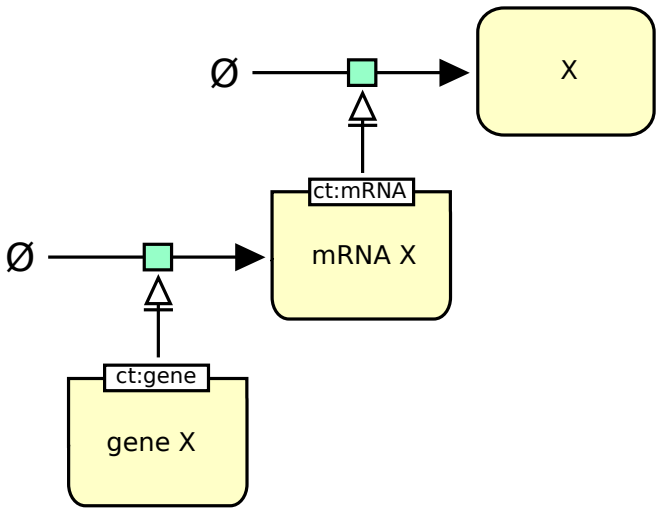


Figure 2.42: *The creation of a messenger RNA X triggered by the gene X.*

The example in Figure **??** below describes the transport of calcium ions out of the endoplasmic reticulum. Without IP3 receptor, there is not calcium flux, therefore, one cannot use a *stimulation*. The Necessary Stimulation instead represents this absolute stimulation.

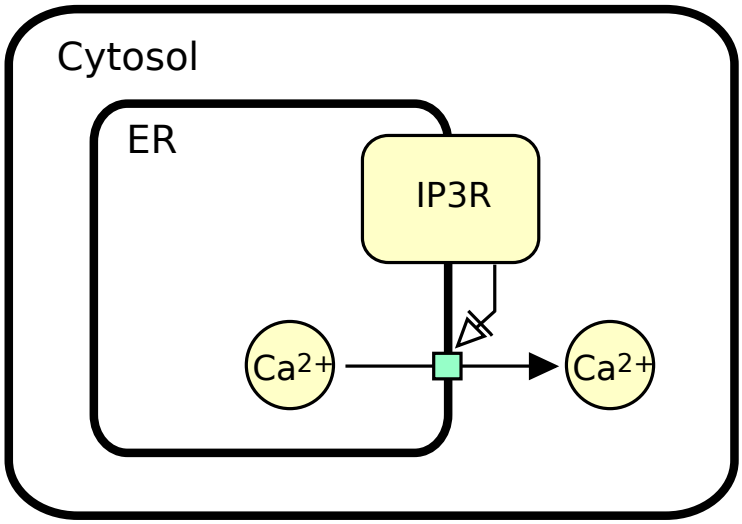


Figure 2.43: *The transport of calcium ions out of the endoplasmic reticulum into the cytosol. Note that IP3R crosses both compartment boundaries. This is allowed, but the Macromolecule should only belong to one of the compartments.*

2.4.8 Glyph: *Logic arc*

Logic arc is used to represent the fact that an entity influences the outcome of a logic operator. A *logic arc* is represented by a simple line without particular symbols at its extremities.

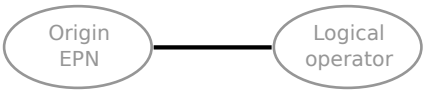


Figure 2.44: *The Process Description glyph for* logic arc.

2.4.9 Glyph: *Equivalence arc*

Equivalence arc is the arc used to represent the fact that all entities marked by a *tag* are equivalent. An *equivalence arc* is represented by a simple line without particular symbols at its extremities.

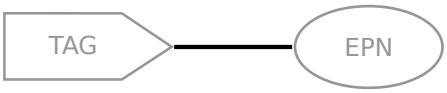


Figure 2.45: *The Process Description glyph for* Equivalence arc.

2.5 Logical operators

2.5.1 Glyph: *And*

The glyph *and* is used to denote that all the *EPNs* linked as input are necessary to produce the output. For instance a modulator A *and* a modulator B, when both present modulate the flux of a process. *And* is represented by a circle carrying the word "AND".

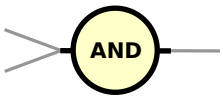


Figure 2.46: *The Process Description glyph for* and. *Only two inputs are represented, but more would be allowed.*

2.5.2 Glyph: *Or*

The glyph *or* is used to denote that any of the *EPNs* linked as input is sufficient to produce the output. *Or* is represented by a circle carrying the word "OR".

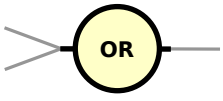


Figure 2.47: *The Process Description glyph for* or. *Only two inputs are represented, but more would be allowed.*

2.5.3 Glyph: *Not*

The glyph *not* is used to denote that the *EPN* linked as input cannot produce the output. *Not* is represented by a circle carrying the word "NOT".

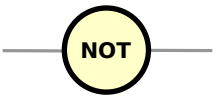


Figure 2.48: *The Process Description glyph for* not.

2.6 Glyph: *Compartment*

A compartment is a logical or physical structure that contains entity pool nodes. An *EPN* can only belong to one compartment. Therefore, the "same" biochemical species located in two different compartments are in fact two different "pools" and should be represented by two *EPNs*. A compartment is represented by a surface enclosed in a continuous border or located between continuous borders. These borders should be noticeably thicker than the borders of the *EPNs*. A compartment can take **any** geometry. A compartment must always be entirely enclosed.

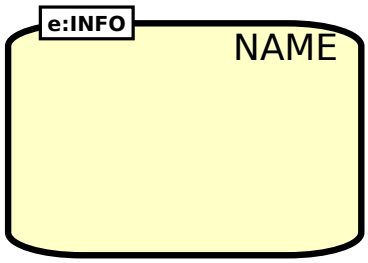


Figure 2.49: *The Process Description glyph for* compartment.

To allow more aesthetically pleasing and understandable maps, compartments are allowed to overlap each other visually, but it must be kept in mind that this does not mean the top compartment contains part of the bottom compartment.

2.6.1 Glyph: *Submap*

A *submap* is used to encapsulate processes (including all types of nodes and edges) within one glyph. The *submap* hides its content to the users, and display only input terminals (or ports), linked to *EPNs* (Section **??**). A *submap* is not equivalent to an *omitted process* (see Section **??**). In the case of an SBGN description that is made available through a software tool, the content of a *submap* may be available to the tool. A user could then ask the tool to expand the *submap*, for instance by clicking on the icon representing the *submap*. The tool might then expand and show the *submap* within the same map (on the same canvas), or it might open it in a different canvas. In the case of an SBGN description made available in a book or a website, the content of the *submap* may be available on another page, possibly accessible via an hyperlink on the *submap*.

The *submap* is represented as a square box to remind the viewer that it is fundamentally a process. A *submap* carries labeled terminals. When the *submap* is represented folded, those terminals are linked to external *EPNs* (Section **??**). In the unfolded view, exposing the internal structure of the *submap*, a set of *tags* point to the corresponding internal *EPNs* Section **??**. A *tag* is represented by a rectangle fused to an empty arrowhead. The symbol should be linked to one and only one edge (i.e., it should reference only one EPN or compartment).

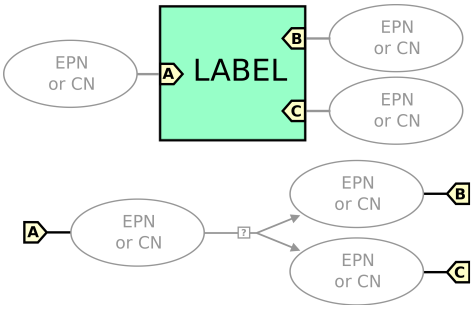


Figure 2.50: *The Process Description glyph for* submap. *(Upper part) folded submap. (Lower part) content of the submap. The* uncertain process *represents the content that is not available outside the submap.*

The left part of Figure **??** represents a *submap* that transforms glucose into fructose-6-phosphate. The *submap* carries five terminals, four linked to EPNs and one linked to a *compartment*. The latter is particularly important in the case of EPNs present only in a *compartment* enclosed in a *submap*, and that are not linked to terminals themselves. Note that the terminals do not define a "direction", such as input or output. The flux of the reactions is determined by the context.

The map on the right of Figure **??** represents an unfolded version of the *submap*. Here, anything outside the *submap* has disappeared, and the internal *tags* are not linked to the corresponding external *terminals*. The yellow nodes are also present in the parent map, while the salmon nodes are specific to the submap. Note the tag 5, linking the compartment "mito" of the *submap* to the compartment "mito" outside the *submap*. The compartment containing Glu6P is implicitly defined as the same as the compartment containing Glu and Fru6P. There is no ambiguity because if Glu and Fru6P were in different compartments, one of them should have been defined within the *submap*.

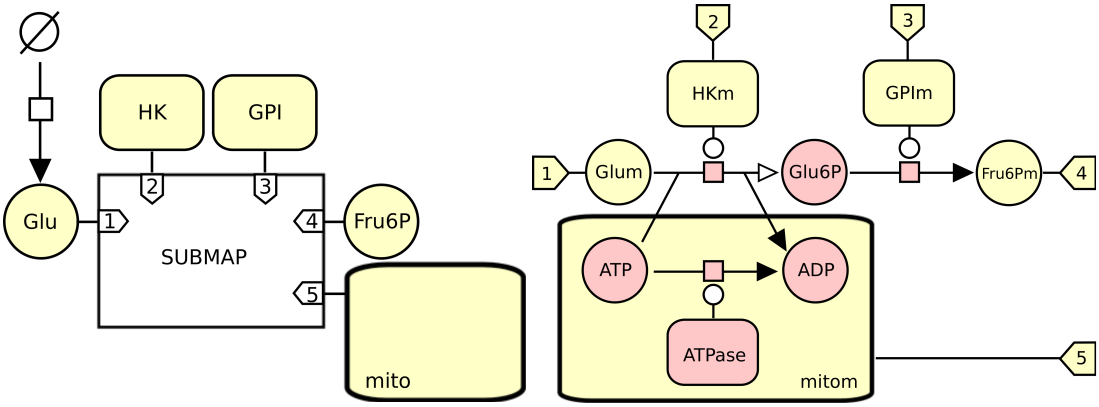


Figure 2.51: *Example of a submap with contents elided.*

Chapter 3

Building a SBGN Process Description map

Now that the various symbols used by the SBGN Process Description language have been introduced, some guidance on building map will be provided.

3.1 How to choose the symbols to use?

It is important to realise that there are in general more than one way to represent a system in SBGN Process Description. The choice of concepts and symbols often depend on the granularity of information available, and the message the authors of the map wish to convey to the readers of the map.

As a first example of variable information granularity, let's take will take the example of MAP kinase phosphorylation (ERK). A very simple representation would be to encode the state of phosphorylation in *entity pools node* with different names, ERK, ERK-P, ERK-PP (Figure **??**).

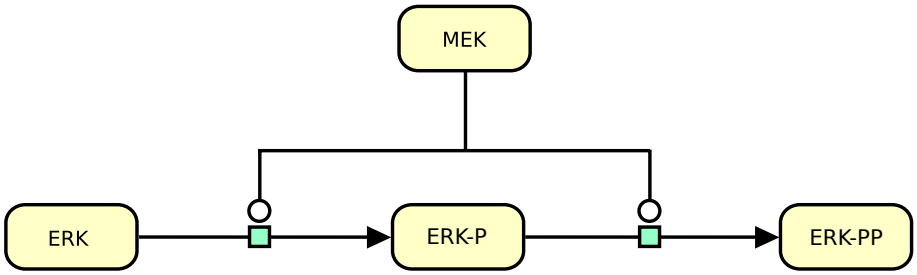


Figure 3.1: *Phosphorylation of ERK by MEK, where ERK is represented by three different macromolecules.*

This kind of representation would be obtained if the SBGN Process Description map was generated from a model encoded in SBML core [**?**]. One of the problems with this representation is the difficulty for the reader to understand that the effect of MEK is to catalyse the phosphorylation of ERK. A scientist familiar with signalling pathways would probably imediately make the connection. A biologist in general could be more cautious. "P" could represent anything (peptide? proline?). A computer would require a special algorithm to parse the names, and this algorithm would easily fail. Instead of ERK-PP, we could have used PP-ERK, ERK_PP, ERKP1P2, ERKTPYP etc. But in fact, there is nothing in the the map Figure **??** indicating that the reactions catalysed add covalent modifications to ERK. Thore reactions could be anything, such as aggregation, cleavage etc.

In order to overcome those issues, one can use state variables. On Figure **??**, one uses a single state variables to represent the number of phosphorylations on ERK.

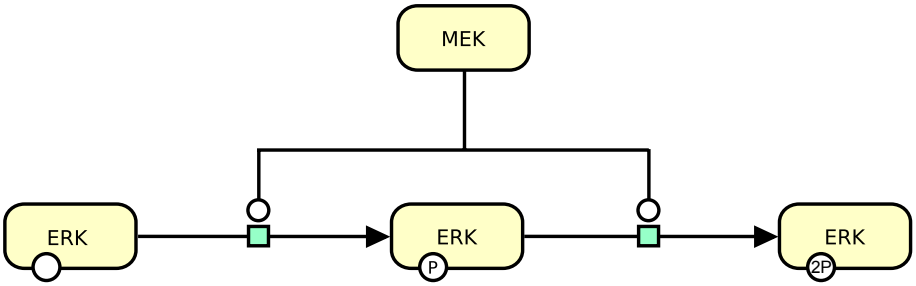


Figure 3.2: *Phosphorylation of ERK by MEK, where ERK is represented by three different states, non-phosphorylated, mono-phosphorylated and bi-phosphorylated.*

Because 'P' is a reserved symbol of the covalent modifications vocabulary, there is no ambiguities. We know that each reaction add a phosphate to ERK. That would be the representation to favour if we only know the number of phophorylation (e.g. by western blot with non-specific antibodies), or if we do not care which site is phosphorylated. Note that the leftmost ERK carries an empty state variable, that is equivalent to "0P". The state variable is not ommited. This rule of SBGN Process Description Level 1 is called "once a variable, always a variable" (OVAV). If we want to, or can, be more specific about distinct phosphorylations, one can create two state variables, one for each site (Figure **??**).

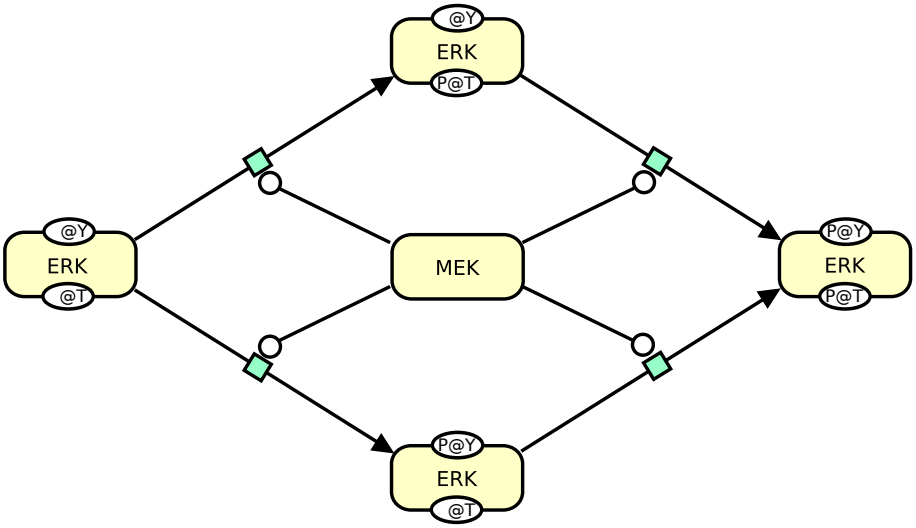


Figure 3.3: *Phosphorylation of ERK by MEK, where each phosphorylated form of ERK is represented.*

In this representation, we have all the information related to the two phosphorylation sites, the threonine and the tyrosine. They are represented by the variable symbols T and Y in the figure. But one could have chosen X and Y or 1 and 2. The important issue is to distinguish them. Note that the creation of an extra *entity pool node* is unavoidable. SBGN Process Description Level 1 does not currently allow logical expressions in the state variables. Therefore if only one *entity pool node* was to be used to represent the single-phosphorylated form, a choice between T or Y should have been made, and the resulting map would not have carried the same information than (Figure **??**).

As a second example, we will consider the oxygenation of hemoglobin. If one wants to convey the message that 4 oxygene molecules bind to a molecule of hemoglobin, it is sufficient to create *macromolecules* for hemoglobin and oxy-hemoglobin.

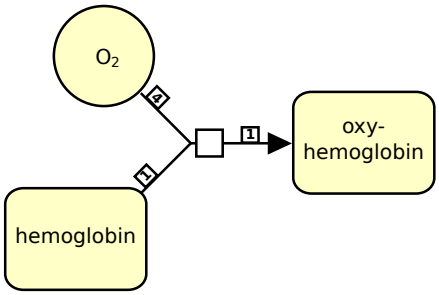


Figure 3.4: *Hemoglobin oxygenation using* macromolecules.

If conveying the fact that hemoglobin is a multimer is important (for instance to suggest cooperativity), one can use *multimers* instead. In addition, in Figure **??**, the concept of oxygenation is represented with a state variable rather than being embedded in the name of the nodes.

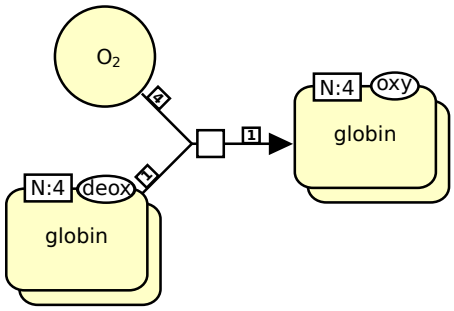


Figure 3.5: *legend*

An additional layer of complexity is needed if we want to mention the α and β subunits. A *complex* can then be used. In addition Figure **??** explicitely represent the complexes between globin and oxygene instead of using state variables.

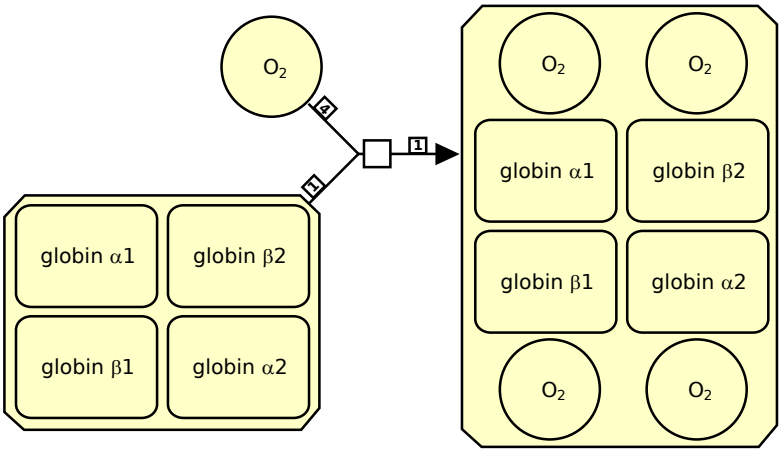
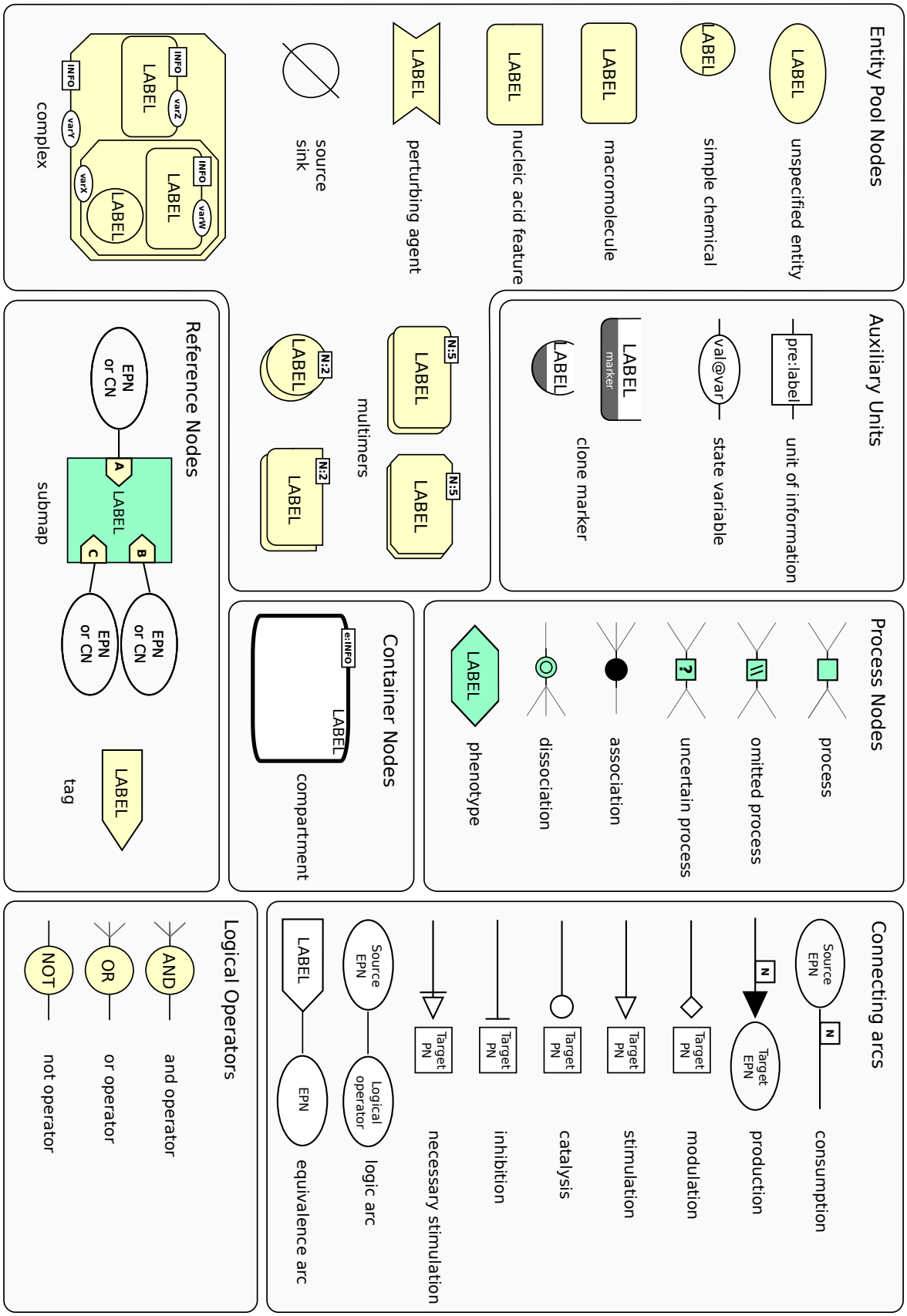


Figure 3.6: *legend*

In conclusion, one can see that many different choice are offered to represent an idea in SBGN, and the map writers make the choice. By doing so, they acknowledge that more or less information can be extracted from the resulting map. The important issue is that anyone reading the map interpret it the same way. That is the topic of the following section.


Entity Pool Nodes
unspecified entity
simple chemical
macromolecule
nucleic acid feature
perturbing agent
source
sink
complex
Auxiliary Units
unit of information
state variable
clone marker
multimers
Process Nodes
process
omitted process
uncertain process
association
dissociation
phenotype
Container Nodes
compartment
Reference Nodes
submap
tag
Connecting arcs
consumption
production
modulation
stimulation
catalysis
inhibition
necessary stimulation
logic arc
equivalence arc
Logical Operators
and operator
or operator
not operator

Bibliography

[1] N Le Novère, Michael Hucka, Huaiyu Mi, Stuart Moodie, Falk Schreiber, Anatoly Sorokin, Emek Demir, Katja Wegner, Mirit I Aladjem, M Sarala Wimalaratne, Frank T Bergman, Ralph Gauges, Peter Ghazal, Hideya Kawaji, Lu Li, Yukiko Matsuoka, Alice Villeger, Sarah E Boyd, Laurence Calzone, Mélanie Courtot, Ugur Dogrusoz, Tom C Freeman, Akira Funahashi, Samik Ghosh, Akiya Jouraku, Sohyoung Kim, Fedor Kolpakov, Augustin Luna, Sven Sahle, Esther Schmidt, Steven Watterson, Guanming Wu, Igor Goryanin, Douglas B Kell, Chris Sander, Herbert Sauro, Jacky L Snoep, Kurt Kohn, and Hiroaki Kitano. The systems biology graphical notation. *Nat Biotechnol*, 27(8):735–741, 2009. 10.1038/nbt.1558.

[2] Stuart Moodie, Nicolas Le Novère, Emek Demir, Huaiyu Mi, and Alice Villéger. Systems biology graphical notation: Process description language level 1. *Nature Precedings*, 2011.

[3] Astrid Junker, Anatoly Sorokin, Tobias Czauderna, Falk Schreiber, and Alexander Mazein. Wiring diagrams in biology: towards the standardized representation of biological information. *Trends in Biotechnology*, 30(11):555–557, 2012.

[4] M. Courtot, N. Juty, C. Knüpfer, D. Waltemath, A. Zhukova, A. Dräger, M. Dumontier, A. Finney, M. Golebiewski, J. Hastings, et al. Controlled vocabularies and semantics in systems biology. *Molecular systems biology*, 7(1), 2011.

[5] M. Hucka, A. Finney, H. M. Sauro, H. Bolouri, J. C. Doyle, H. Kitano, A. P. Arkin, B. J. Bornstein, D. Bray, A. Cornish-Bowden, A. A. Cuellar, S. Dronov, E. D. Gilles, M. Ginkel, V. Gor, I. I. Goryanin, W. J. Hedley, T. C. Hodgman, J.-H. Hofmeyr, P. J. Hunter, N. S. Juty, J. L. Kasberger, A. Kremling, U. Kummer, N. Le Novère, L. M. Loew, D. Lucio, P. Mendes, E. Minch, E. D. Mjolsness, Y. Nakayama, M. R. Nelson, P. F. Nielsen, T. Sakurada, J. C. Schaff, B. E. Shapiro, T. S. Shimizu, H. D. Spence, J. Stelling, K. Takahashi, M. Tomita, J. Wagner, and J. Wang. The Systems Biology Markup Language (SBML): A medium for representation and exchange of biochemical network models. *Bioinformatics*, 19(4):524–531, 2003.