

Systems Biology Graphical Notation: Activity Flow language Level 1 Version 1.2

Huaiyu Mi^{1*}, Falk Schreiber², Stuart Moodie³, Tobias Czauderna⁴, Emek Demir⁵, Robin Haw⁶, Augustin Luna⁵, Nicolas Le Novère⁷, Anatoly Sorokin⁸ and Alice Villéger⁹

¹University of Southern California, USA

²Monash University, Australia and MLU Halle, Germany

³Eight Pillars Ltd, UK

⁴Monash University, Australia

⁵Memorial Sloan-Kettering Institute, USA

⁶Ontario Institute for Cancer Research, Canada

⁷Babraham Institute, UK

⁸Institute of Cell Biophysics RAS, RU

⁹Freelance IT Consultant, UK

Summary

The Systems Biological Graphical Notation (SBGN) is an international community effort for standardized graphical representations of biological pathways and networks. The goal of SBGN is to provide unambiguous pathway and network maps for readers with different scientific backgrounds as well as to support efficient and accurate exchange of biological knowledge between different research communities, industry, and other players in systems biology. Three SBGN languages, Process Description (PD), Entity Relationship (ER) and Activity Flow (AF), allow for the representation of different aspects of biological and biochemical systems at different levels of detail.

The SBGN Activity Flow language represents the influences of activities among various entities within a network. Unlike SBGN PD and ER that focus on the entities and their relationships with others, SBGN AF puts the emphasis on the functions (or activities) performed by the entities, and their effects to the functions of the same or other entities. The nodes (elements) describe the biological activities of the entities, such as protein kinase activity, binding activity or receptor activity, which can be easily mapped to Gene Ontology molecular function terms. The edges (connections) provide descriptions of relationships (or influences) between the activities, e.g., positive influence and negative influence. Among all three languages of SBGN, AF is the closest to signaling pathways in biological literature and textbooks, but its well-defined semantics offer a superior precision in expressing biological knowledge.

*To whom correspondence should be addressed. Email: sbgn-editors@lists.sourceforge.net

Systems Biology Graphical Notation: Activity Flow language Level 1

Version 1.2

Date: July 27, 2015

Editors:

Huaiyu Mi
Falk Schreiber
Stuart Moodie
Tobias Czauderna
Emek Demir
Robin Haw
Augustin Luna
Nicolas Le Novère
Anatoly Sorokin
Alice Villéger

University of Southern California, USA
Monash University, Australia & MLU Halle, Germany
Eight Pillars Ltd, UK
Monash University, Australia
Memorial Sloan-Kettering Institute, USA
Ontario Institute for Cancer Research, Canada
Memorial Sloan-Kettering Institute, USA
Babraham Institute, UK
Institute of Cell Biophysics RAS, RU
Freelance IT Consultant, UK

To discuss any aspect of SBGN, please send your messages to the mailing list sbgn-discuss@caltech.edu. To get subscribed to the mailing list or to contact us directly, please write to sbgn-editors@lists.sourceforge.net. Bug reports and specific comments about the specification should be entered in the issue tracker <http://sf.net/p/sbgn/sbgn-af-11/>.



Contents

1 Introduction	1		
1.0.1 What are the languages?	1	2.9.2 Glyph: <i>Or</i>	19
1.0.2 Nomenclature	1	2.9.3 Glyph: <i>Not</i>	20
1.1 SBGN levels and versions	2	2.9.4 Glyph: <i>delay</i>	20
1.2 Developments, discussions, and notifications of updates	2		
2 Activity Flow glyphs	3	3 Activity Flow language grammar	22
2.1 Overview	3	3.1 Overview	22
2.2 Controlled vocabularies used in SBGN Activity Flow Level 1	4	3.2 Concepts	22
2.2.1 Unit of information material types	4	3.3 Syntax	22
2.2.2 Unit of information conceptual types	5	3.3.1 Activity Nodes connectivity definition	23
2.2.3 Physical characteristics of compartments	5	3.3.2 Containment definition	23
2.3 Activity nodes	6	3.3.3 Syntactic rules	23
2.3.1 Glyph: <i>Biological activity</i>	6	3.4 Semantic rules	23
2.3.2 Glyph: <i>Phenotype</i>	7	3.4.1 Activity node and Compartment	23
2.4 Auxiliary units	8	3.4.2 Modulation	24
2.4.1 Glyph: <i>Unit of information for Biological activity</i>	8	3.4.3 Submaps	24
2.4.2 Glyph: <i>Unit of information for Compartment</i>	10	4 Layout Guidelines for an Activity Flow Maps	25
2.5 Glyph: <i>Annotation</i>	10	4.1 Introduction	25
2.6 Container nodes	11	4.2 Layout guidelines	26
2.6.1 Glyph: <i>Compartment</i>	11	4.2.1 Requirements	26
2.7 Glyph: <i>Submap</i>	13	4.2.2 Recommendations	27
2.8 Modulation arcs	15	4.2.3 Additional suggestions	28
2.8.1 Glyph: <i>Positive influence</i>	15	5 Acknowledgments	29
2.8.2 Glyph: <i>Negative influence</i>	16	5.1 Level 1 Release 1.0	29
2.8.3 Glyph: <i>Unknown influence</i>	16	5.2 Level 1 Release 1.1	29
2.8.4 Glyph: <i>Necessary stimulation</i>	17	5.3 Level 1 Release 1.2	29
2.8.5 Glyph: <i>Logic arc</i>	17	5.4 Comprehensive list of acknowledgements	29
2.8.6 Glyph: <i>Equivalence arc</i>	18	5.5 Financial Support	30
2.9 Logical operators	19	A Examples	31
2.9.1 Glyph: <i>And</i>	19	B Reference card	35
		C Revision History	37
		C.1 Version 1.0 to Version 1.1	37
		C.2 Version 1.1 to Version 1.2	37

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 [1]. This document defines the *Activity Flow* visual language of SBGN. Activity Flows are one of three views of a biological process offered by SBGN. It is the product of many hours of discussion and development by many individuals and groups.

1.0.1 What are the languages?

PD is a language that permits the description of all the processes taking place in a biological system. The ensemble of all these processes constitute a Description. **ER** is a language that permits the description of all the relations involving the entities of a biological system. The ensemble of all these relations constitute a Relationship. **AF** is a language that permits the description of the flow of activity in a biological system.

1.0.2 Nomenclature

The three languages of SBGN should be referred to as:

- the Process Description language.
- the Entity Relationship language.
- the Activity Flow language.

Abbreviated as:

- the PD language.
- the ER language.
- the AF language.

A specific representation of a biological system in one of the SBGN languages should be referred to as:

- a Process Description map.
- an Entity Relationship map.
- an Activity Flow map.

Abbreviated as:

- a PD map.
- an ER map.
- an AF map.

The corpus of all SBGN representations should be referred to as:

- Process Descriptions.
- Entity Relationships.
- Activity Flows.

The capitalization is important. PD, ER and AF are names of languages. As such they must be capitalized in English. This is not the case of the accompanying noun (language or map).

1.1 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 development 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 evolutions 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., Activity Flow maps that conform to an earlier version of the Activity Flow language within the same level should still be valid. This does not apply to a new level.

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 level adding richness compared to the levels below it.

1.2 Developments, discussions, and notifications of updates

The SBGN website (<http://sbgn.org/>) is a portal for all things related to SBGN. It provides a web forum interface to the SBGN discussion list (sbgn-discuss@caltech.edu) and information about how anyone may subscribe to it. The easiest and best way to get involved in SBGN discussions is to join the mailing list and participate.

Face-to-face meetings of the SBGN community are announced on the website as well as the mailing list. Although no set schedule currently exists for workshops and other meetings, we envision holding at least one public workshop per year. As with other similar efforts, the workshops are likely to be held as satellite workshops of larger conferences, enabling attendees to use their international travel time and money more efficiently.

Notifications of updates to the SBGN specification are also broadcast on the mailing list and announced on the SBGN website.

Chapter 2

Activity Flow glyphs

This chapter provides a catalog of the graphical symbols available for representing entities in Activity Flows. In Chapter 3 beginning on page 22, we describe the rules for combining these glyphs into a legal SBGN Activity Flow map, and in Chapter 4 beginning on page 25, we describe requirements and guidelines for the way that Activity Flow maps are visually organized.

2.1 Overview

To set the stage for what follows in this chapter, we first give a brief overview of some of the concepts in the Activity Flow notation with the help of an example shown in Figure 2.1 on the following page.

The diagram illustrates the regulation of peroxisome proliferator-activated receptor delta (PPAR delta, a nuclear hormone receptor) on brown fat metabolism, a redraw from Fig 7E of Pan et. al [2]. The rectangle nodes represent *biological activities* - activities from biological materials. The type of material is indicated in the units of information decorating on the activity nodes (See Section 2.3.1). Each biological activity can influence, or be influenced by, other biological activities, and such relationships are represented in Activity Flow by lines with arrows and other decorations. It should be noted that the essence of Activity Flow is to show the flow of activities from one entity to another or within the same entity. For example, in the diagram, it shows that PPAR δ positively influences the Twist-1 gene expression. The underlying mechanisms of how the influence occurs may not be known and is not captured in the diagram. If the mechanism is known, the details should be described in a Process Description and/or Entity Relationship map.

Table 2.1 summarizes the different SBGN abstractions described in this chapter.

Component	Abbrev.	Role	Examples
Activity node	AN	A functional unit that can affect, or be affected by, another functional unit	Biological activity
Container node	CN	An encapsulation of one or more other SBGN constructs	Compartments
Modulating arc	MA	Links between different activities to indicate influences	Positive influence, Negative influence
Auxiliary units	AU	A decorating glyph to the AN to provide additional information of the node, such as the property where the activity is originated	Unit of information
Logical operators	—	Combines one or several inputs into one output	Boolean <i>and</i> , <i>or</i> , <i>not</i> , <i>delay</i>

Table 2.1: Summary of Activity Flow components and their roles.

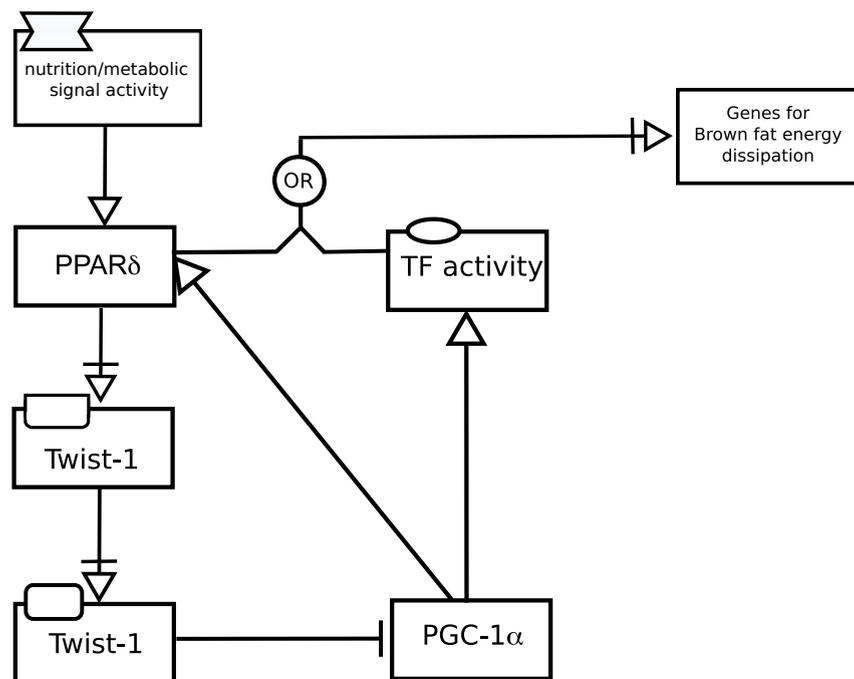


Figure 2.1: This example of Activity Flow is based on Figure 7E of Pan et. al. [2]. It depicts the effect of nutrients and metabolic signals on brown fat metabolism through PPAR δ . The signal, shown as a perturbation, positively influences the nuclear hormone receptor PPAR δ , which in turn stimulates the Twist 1 gene expression. Please note the different units of information on Twist-1 activity nodes that indicate the activity from different biological materials (gene and protein). The Twist-1 protein negatively influences the PGC-1 α activity, which positively influences PPAR δ and other unspecified transcription factor activity to stimulate the expression of genes for brown fat energy dissipation. Therefore, the Twist-1, induced by PPAR δ , serves as a negative feedback regulator of PGC-1 α in brown fat metabolism.

2.2 Controlled vocabularies used in SBGN Activity Flow Level 1

Some glyphs in SBGN Activity Flow can contain particular kinds of textual annotations conveying information relevant to the purpose of the glyph. These annotations are *units of information* (Section 2.4.1). An example is in the case of a *biological activity*, which can have a *unit of information* conveying the type of entity the activity is from.

The text that appears as the *unit of information* decorating a Container Node (CN) must in most cases be prefixed with a controlled vocabulary term indicating the type of information being expressed.

In the rest of this section, we describe the controlled vocabularies (CVs) used in SBGN Activity Flow Level 1. Some CV terms are predefined by SBGN, but unless otherwise noted, they are not the only terms permitted. 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.

2.2.1 Unit of information material types

The material type of an AN can be visualized in the *unit of information* glyph to indicate its chemical structure. A list of common material types is shown in Table 2.2 on the following page, 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 having identifier **SBO:0000240** (*material entity* under *entity*). It is optional to use them in the text to label the *unit of information*.

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

Table 2.2: A sample of values from the material types controlled vocabulary (Section 2.2.1).

The material types are in contrast to the *conceptual types* (see below). The distinction is that material types are about physical composition, while conceptual types are about roles. For example, a strand of RNA is a physical artifact, but its use as messenger RNA is a role.

2.2.2 Unit of information conceptual types

An *conceptual type* indicates the function within the context of a given Activity Flow. A list of common conceptual types is shown in Table 2.3, 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 having identifier SBO:0000241 (*conceptual entity* under *entity*). It is optional to use them in the text to label the *unit of information*.

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 2.3: A sample of values from the conceptual types vocabulary (Section 2.2.2).

2.2.3 Physical characteristics of compartments

SBGN Activity Flow Level 1 defines a special unit of information for describing certain common physical characteristics of compartments. Table 2.4 lists the particular values defined by SBGN Activity Flow Level 1. The values correspond to the Systems Biology Ontology branch with the identifier SBO:0000002 (*quantitative parameter*).

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 2.2.3).

2.3 Activity nodes

An *Activity node* (AN) represents the activity of an entity or an entity pool, but not the entities themselves. For instance, multiple activity nodes can be used to represent different activities of a particular entity, while one activity node can be used to represent the activity of a complex multimer. In addition to activities of material entities, SBGN Activity Flow Level 1 represents activity from a conceptual entity: *phenotype*. Auxiliary units, such as *units of information*, can be used to indicate the material property of the activity source. Each activity is displayed only once in a particular compartment.

2.3.1 Glyph: *Biological activity*

SBGN Activity Flow Level 1 uses one glyph to represent molecular activities of all types of biological entities, collectively they are called *biological activity*. The nature of the molecule that the activity comes from, e.g., simple chemical or macromolecule, can be encoded in the *units of information* (Section 2.4.1).

It should be noted that the *biological activity* is not equivalent to a biological entity per se. A biological activity can come from one biological entity, a part of an entity, or a combination of them. It is up to the users to determine how to represent it in their diagram. For example, a protein kinase receptor such as an EGF receptor, has two activities, the binding activity that allows the extracellular part of the receptor to bind to the ligand, and the kinase activity that is capable of phosphorylating the downstream protein and initiating the intracellular signaling. The user can choose to use two nodes to represent each activity, or to use one node to represent the overall "EGF receptor activity" (Figure 2.3 on the next page). More examples can be found in examples in Appendix A.

SBO Term:

SBO:0000412 ! biological activity

Container:

A *biological activity* is represented by a rectangle, as shown in Figure 2.2 on the following page.

Label:

A *biological activity* is identified by a label placed in an unbordered box containing a string of characters. The characters can be distributed on several lines to improve readability, although this is not mandatory. The label box must be attached to the center of the container. The label may spill outside of the container.

Auxiliary items:

A *biological activity* can carry a *unit of information* (Section 2.4.1), which can provide information such as the nature of the entity from which the activity originated. Specific glyphs are used to represent different types of entities (Section 2.4.1). The center of the bounding box of a *unit of information* is located on the mid-line of the border of the *biological activity* node. The label in the *unit of information*, which is optional, indicates the name of the molecule where the activity comes from, as shown in Figure 2.3 on the next page.



Figure 2.2: *The Activity Flow glyph for biological activity.*

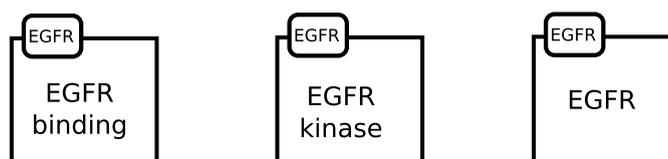


Figure 2.3: *An example of Activity Flow glyphs of EGFR activities. Since EGFR protein has both binding and kinase activities, each of those activities can be represented by different nodes, labeled as EGFR binding and EGFR kinase. One node can be used to represent the overall activity of EGFR. The label in the unit of information indicates the protein that the activities come from. In this example, all three activities come from the same EGFR protein*

2.3.2 Glyph: *Phenotype*

A phenotype is a type of biological process. In SBGN, *phenotype* is used to show the observable or measurable outcome of the network. It is usually the end-point(s) of the network, i.e., it can not be used as the start of an arc.

SBO Term:

SBO:0000358 ! phenotype

Container:

A *phenotype* is represented by an elongated hexagon, as illustrated in Figure 2.4.

Label:

An *phenotype* is identified by a label placed in an unbordered box containing a string of characters. The characters can be distributed on several lines to improve readability, although this is not mandatory. The label box must be attached to the center of the *phenotype* container. The label may spill outside of the container.



Figure 2.4: *The Activity Flow glyph for phenotype.*

2.4 Auxiliary units

2.4.1 Glyph: *Unit of information for Biological activity*

When representing biological activities, it is often useful to illustrate the nature of the entity where the activity is originated, e.g., whether the activity is from a macromolecule (protein or nucleic acid), or from a chemical compound. The SBGN Activity Flow Level 1 *unit of information* is used to add such information to a glyph. It represents the information in two ways. First, different symbols are used to represent the nature of the entity where the activity is from, e.g., macromolecule, nucleic acid feature, or complex. These symbols are identical to the *entity pool node* symbols in SBGN Process Description language. Second, names of the entity (gene names, protein names) are usually provided as labels in the *unit of information* container.

SBO Term:

Not applicable.

Container:

A unit of information is represented by containers of different shapes, depending on the nature of the entity where the biological activity is from. There are a total of six types of unit of information, as shown in Figure 2.5 on the next page. Below is a summary of the six glyphs.

- A.** macromolecule – Macromolecules are 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.). A unit of information of a macromolecule is represented by a rectangle with rounded corners, as illustrated in (A) of Figure 2.5 on the following page. This container is used to decorate a biological activity that is originated from a macromolecule, such as a protein, a nucleic acid, or a complex sugar.
- B.** nucleic acid feature – The Nucleic acid feature construct in SBGN is meant to represent a fragment of a macromolecule carrying genetic information. A unit of information of a nucleic acid feature is represented by a rectangle whose bottom half has rounded corners, as shown in (B) of Figure 2.5 on the next page.
- C.** simple chemical – A simple chemical 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 unit of information of a simple chemical is represented by a circular container, as shown in (C) of Figure 2.5 on the following page.
- D.** unspecified entity – An unspecified entity is used to represent the entity type that 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 relevance. A unit of information of an unspecified entity is represented by an elliptic container, as shown in (D) of Figure 2.5 on the next page. It is used to decorate a biological activity that is originated from an unspecified entity.
- E.** complex – A complex represents a biochemical entity composed of other biochemical entities, whether macromolecules, simple chemicals, or other complexes. The resulting entity may have its own identity, properties and function in an SBGN map. A unit of information of a complex is represented by an octagon as shown in (E) of Figure 2.5 on the following page. It is used to decorate a biological activity that is originated from a complex.
- F.** perturbation – Biochemical networks can be affected by external influences. Those influences can be well-defined physical perturbations, such as a light pulse or a change

in temperature; they can also be more complex and not well-defined phenomena, for instance, glucose deprivation, stress. A unit of information of a perturbation is represented by a modified hexagon having two opposite concave faces, as illustrated in F of Figure 2.5. It is used to decorate a biological activity when it is originated from a perturbation.

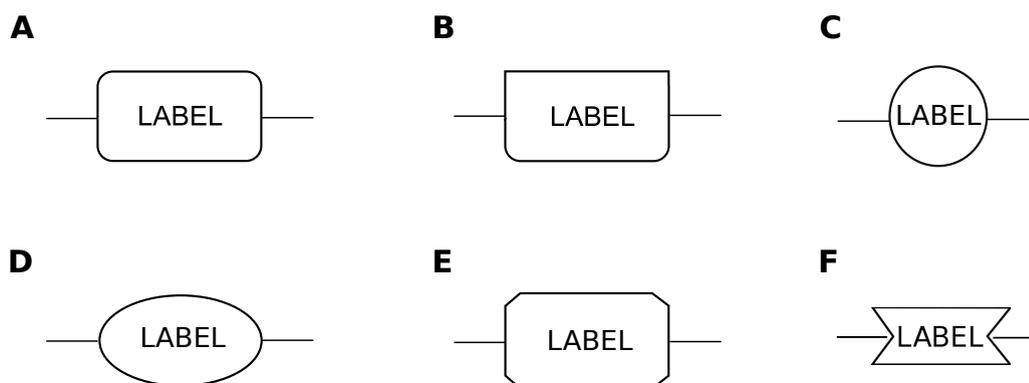


Figure 2.5: *The Activity Flow glyph for unit of information.*

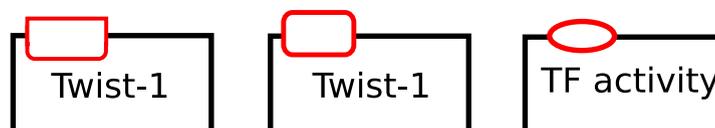


Figure 2.6: *Examples of unit of information used on biological activity node to indicate that the Twist-1 activity is from a nucleic acid feature or a macromolecule, or a transcription factor activity from an unspecified entity.*

Figure 2.6 shows examples taken from Figure 2.1 on page 4, where *units of information* is used on Activity Nodes to illustrate the properties of the entities that the activities are originated from.

The long side of the glyphs above (except for simple chemical) should be oriented parallel to the border of the AN being annotated by the *unit of information*. The center of the bounding box of a *state of information* should be located on the mid-line of the border of the AN.

Label:

A *unit of information* is not required to carry any label. If a label is desired, it should be placed in an unbordered box containing a string of characters. The characters can be distributed on several lines to improve readability, although this is not mandatory. The label box must be attached to the center of the container. The label may spill outside of the container. The label defines the information carried by the *unit of information*.

Auxiliary items:

A *unit of information* does not carry any auxiliary items.

2.4.2 Glyph: *Unit of information for Compartment*

A *unit of information* can be used to decorate a compartment to convey information about physical characteristics of the compartments (Section 2.2.3).

SBO Term:

Not applicable.

Container:

A *unit of information* for compartment is represented by a rectangle as shown in Figure 2.7. The long side of the rectangle should be oriented parallel to the border of the *compartment* being annotated by the *unit of information*. The center of the bounding box of a *unit of information* should be located on the mid-line of the border of the *compartment*.



Figure 2.7: *The Activity Flow glyph for unit of information.*

Label:

A *unit of information* is identified by a label placed in an unbordered box containing a string of characters. The characters can be distributed on several lines to improve readability, although this is not mandatory. The label box must be attached to the center of the container. The label may spill outside of the container.

The label defines the information carried by the *unit of information*. For certain predefined types of information for the compartment, such as physical characteristics, 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 Activity Flow Level 1 are described in Section 2.2.3 and summarized in the following list:

- pc:T Temperature (SBO:0000147)
- pc:V Voltage (SBO:0000259)
- pc:pH pH (SBO:0000304)

Auxiliary items:

A *unit of information* does not carry any auxiliary items.

2.5 Glyph: *Annotation*

SBGN Activity Flow Level 1 defines a glyph to add additional information to a map, that does not modify the semantic of the graph. This glyph can be used to add free text, or links to external information.

SBO Term:

SBO:0000550 ! annotation

Container:

An *annotation* is represented by a rectangular container with a folded corner, as illustrated in Figure 2.8. This container is linked to the annotated element with a callout. The link ends up on the border of the annotated element.

Label:

An *annotation* contains information placed in an unbordered box containing a string of characters. The characters can be distributed on several lines to improve readability, although this is not mandatory. The label box must be attached to the center of the container. The label may spill outside of the container.

Auxiliary items:

An *annotation* does not carry any auxiliary unit.



Figure 2.8: *The Activity Flow glyph for annotation.*

2.6 Container nodes

Containers are SBGN constructions that contain one or several other SBGN constructs. In SBGN Activity Flow Level 1 *compartment* is the only container node.

2.6.1 Glyph: *Compartment*

A compartment is a logical or physical structure where the function or activity is located. At the moment, an activity can only belong to one compartment. Therefore, the “same” biochemical activities located in two different compartments are in fact two different activities, and should be represented separately.

SBO Term:

SBO:0000289 ! functional compartment

Container:

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 ANs. A compartment can take **any** geometry. A compartment must always be entirely enclosed.

Label:

The identification of the compartment is carried by an unbordered box containing a string of characters. The characters can be distributed on several lines to improve readability, although this is not mandatory. The label box can be attached anywhere in the container box. Note that the label can spill-over from the container box.

Auxiliary items:

A *compartment* can carry a certain number of *units of information*, that will add information, for instance, about the physical environment, such as pH, temperature or voltage,

see Section 2.4.2. The center of the bounding box of a *unit of information* is located on the mid-line of the border of the compartment.

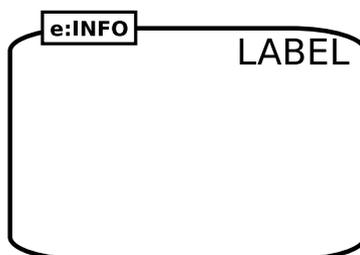


Figure 2.9: *The Activity Flow glyph for compartment.*

It is important to note that a compartment never contains another compartment. To allow more aesthetically pleasing and understandable diagrams, compartments are allowed to overlap each other visually, but it must be kept in mind that this does not mean one compartment contains part or entire of the other compartment. Figure 2.10 shows three semantically equivalent placement of compartments:

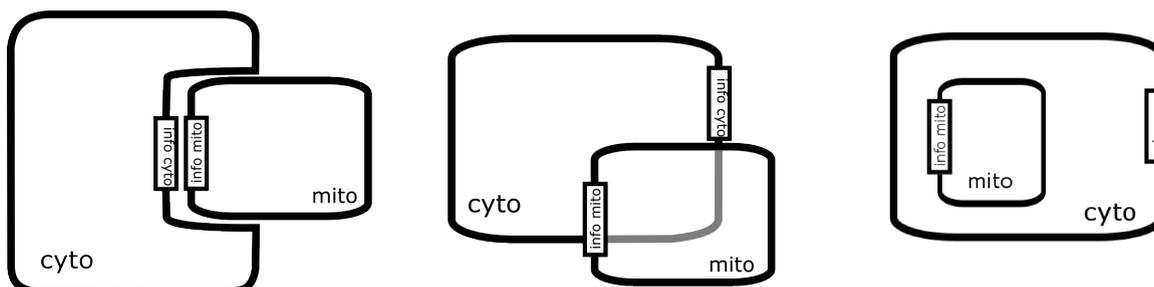


Figure 2.10: *Overlapped compartments are permitted, but the overlap does not imply containment.*

Overlapped (hidden) part of the compartment should not contain any object which could be covered by an overlapping compartment. Figure 2.11 illustrates the problem using an incorrect diagram.

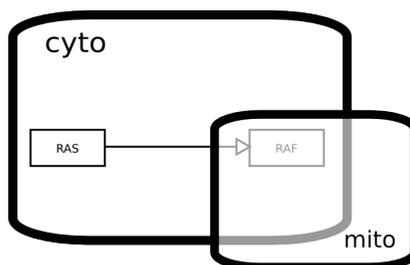


Figure 2.11: *Example of an incorrect diagram. Overlapped compartments must not obscure other objects.*

2.7 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 displays only input terminals (or ports), linked to *ANs*. In the case of an SBGN diagram 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 for the submap. The tool might then expand and show the submap within the same diagram (on the same canvas), or it might open it in a different canvas.

SBO Term:

SBO:0000395 ! encapsulating process

Container:

The *submap* is represented as a rectangle box to remind the viewer that it is fundamentally a biological activity node.

Label:

The identification of the *submap* is carried by an unbordered box containing a string of characters. The characters may be distributed on several lines to improve readability, although this is not mandatory. The label box has to be attached to the center of the container box.

Auxiliary items:

A *submap* carries labeled terminals. When the submap is represented folded, those terminals are linked to external *ANs*. In the unfolded view, exposing the internal structure of the submap, a set of *tags* point to the corresponding internal *ANs*.

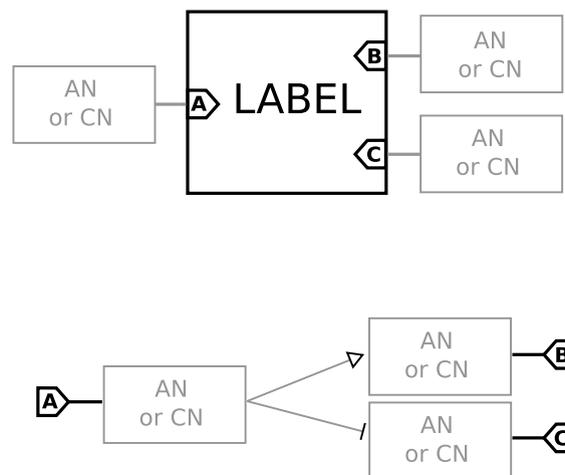


Figure 2.12: *The Activity Flow glyph for submap. (Upper part) folded submap. (Lower part) content of the submap.*

Figure 2.13 on the following page represents a *submap* of inhibitory G-protein coupled receptor signaling. The *submap* carries five terminals, three linked to biological *ANs*, and two linked to *compartments*. Note that the terminals do not define a “direction”, such as input or output. The flux of the activities is determined by the context.

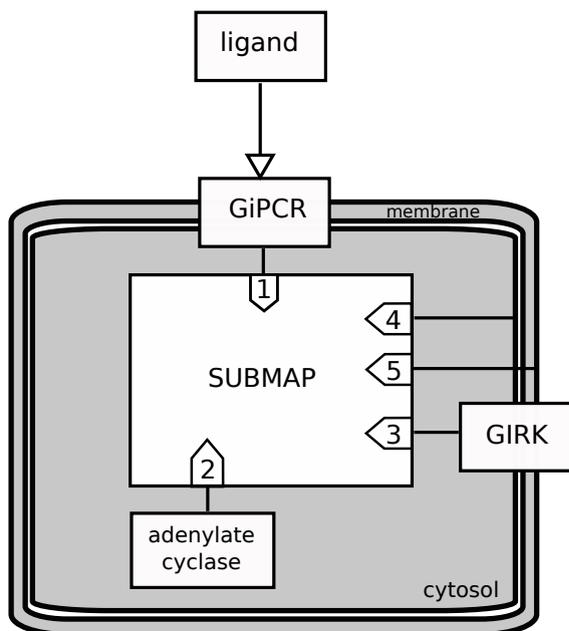


Figure 2.13: Example of a submap with contents elided.

The diagram in Figure 2.14 represents an unfolded version of a submap. Here, anything outside the submap has disappeared (e.g., ligand in Figure 2.13), and the internal *tags* are not linked to the corresponding external *terminals*.

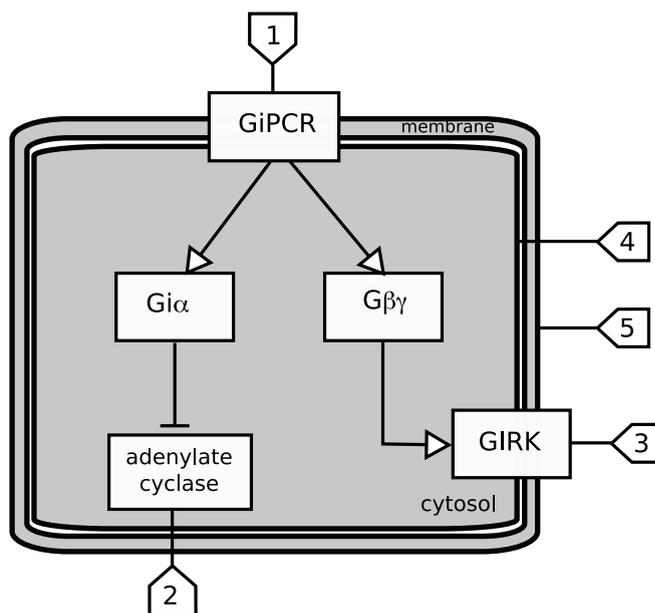


Figure 2.14: Example of an unfolded submap. The unfolded submap corresponds to the folded submap of Figure 2.13.

2.8 Modulation arcs

Modulation arcs are lines that link ANs together. The symbols attached to their end extremities indicate their semantics. The modulation arcs can be used to represent direct influence from one activity to another, such as nicotine to nicotinic acetylcholine receptor activity, or indirect influence.

2.8.1 Glyph: *Positive influence*

In SBGN Activity Flow Level 1, a *positive influence* is defined as an action that produces **positive** or **activating** effect from one activity to another.

SBO Term:

SBO:0000170 ! stimulation

Origin:

Any *biological activity* (Section 2.3.1) or any *logical operator* (Section 2.9).

Target:

Any *biological activity* (Section 2.3.1) or *phenotype* (Section 2.3.2).

Symbol:

The target extremity of a *positive influence* carries an open arrow pointing to the target activity node (Figure 2.15).



Figure 2.15: *The Activity Flow glyph for positive influence.*

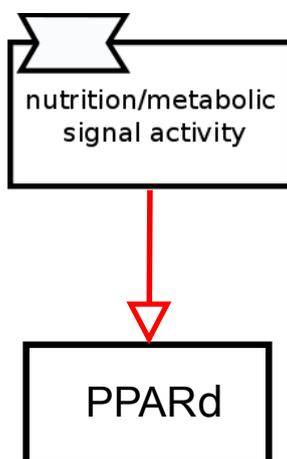


Figure 2.16: *An example, taken from Figure 2.1 on page 4, of positive influence from a perturbation to the nuclear hormone receptor PPAR δ .*

2.8.2 Glyph: *Negative influence*

A *negative influence* is defined as an action that produces a **negative** or **inhibiting** effect from one activity to another.

SBO Term:

SBO:0000169 ! inhibition

Origin:

Any *biological activity* (Section 2.3.1) or any *logical operator* (Section 2.9).

Target:

Any *biological activity* (Section 2.3.1) or *phenotype* (Section 2.3.2).

Symbol:

The target extremity of a *negative influence* carries a bar perpendicular to the arc (Figure 2.17).



Figure 2.17: *The Activity Flow glyph for negative influence.*



Figure 2.18: *An example, taken from Figure 2.1 on page 4, of negative influence from "Twist-1" protein activity to "PGC-1α" activity.*

2.8.3 Glyph: *Unknown influence*

An *unknown influence* is used when the effect exerted from one activity to another is not well understood, or sometimes understood but complex.

SBO Term:

SBO:0000168 ! control

Origin:

Any *biological activity* (Section 2.3.1) or any *logical operator* (Section 2.9).

Target:

Any *biological activity* (Section 2.3.1) or *phenotype* (Section 2.3.2).

Symbol:

The target extremity of a *unknown influence* carries an open diamond (Figure 2.19 on the next page).

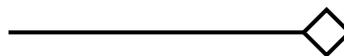


Figure 2.19: *The Activity Flow glyph for unknown influence.*

2.8.4 Glyph: *Necessary stimulation*

A *necessary stimulation* is an influence that has to be present for the target activity to take place (to become true). An activity modulated by a necessary stimulation can only exist when this stimulation is true, whatever are the other influences this activity is subjected to.

SBO Term:

SBO:0000171 ! necessary stimulation

Origin:

Any *biological activity* (Section 2.3.1) or any *logical operator* (Section 2.9).

Target:

Any *biological activity* (Section 2.3.1) or *phenotype* (Section 2.3.2).

Symbol:

The target extremity of a *necessary stimulation* carries a perpendicular bar followed by an open arrow pointing to the target activity node (Figure 2.20). The bar has to be at least as long as the based of the arrowhead.



Figure 2.20: *The Activity Flow glyph for necessary stimulation.*

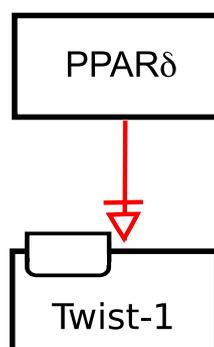


Figure 2.21: *An example, taken from Figure 2.1 on page 4, of necessary stimulation where nuclear hormone receptor PPAR δ transcription factor activity is necessary for the stimulation of the Twist-1 gene expression.*

2.8.5 Glyph: *Logic arc*

Logic arc is used to represent the fact that an activity influences the outcome of a logic operator.

SBO Term:

SBO:0000398 ! logical relationship.

Origin:

Any *biological activity* (Section 2.3.1) or any *logical operator* (Section 2.9).

Target:

Any *logical operator* (Section 2.9).

Symbol:

No particular symbol is used to represent the end a logic arc.

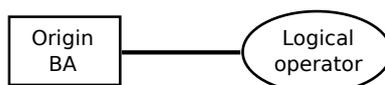


Figure 2.22: *The Activity Flow glyph for logic arc. BA = biological activity*

2.8.6 Glyph: *Equivalence arc*

Equivalence Arc is used to represent the fact that all activities or compartments marked by a *tag* are equivalent. In an Activity Flow map, it is used to show that an AN in a submap and another AN in the main map are equivalent.

SBO Term:

Not applicable.

Origin:

Any *Activity node* (Section 2.3) or any *compartment*.

Target:

Tag or *Terminal* (submap).

Symbol:

No particular symbol is used to represent the end of an *equivalence arc*.

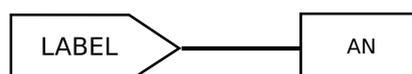


Figure 2.23: *The Activity Flow glyph for Equivalence arc.*

2.9 Logical operators

2.9.1 Glyph: *And*

The glyph *and* is used to denote that all the *ANs* linked as input are necessary to influence the target activity.

SBO Term:

SBO:0000173 ! and.

Origin:

More than one *biological activity* (Section 2.3.1) and *logical arcs* (Section 2.8.5).

Target:

Modulation arc (Section 2.8) other than *equivalence arc*.

Node:

And is represented by a circle carrying the word “AND”, with two connectors located at the opposite side for inputs and output.



Figure 2.24: *The Activity Flow glyph for and. Only two inputs are represented, but more would be allowed.*

2.9.2 Glyph: *Or*

The glyph *or* is used to denote that any of the *ANs* linked as input is sufficient to influence the target activity.

SBO Term:

SBO:0000174 ! or.

Origin:

More than one *biological activity* (Section 2.3.1) and *logical arcs* (Section 2.8.5).

Target:

Modulation arc (Section 2.8) other than *equivalence arc*.

Node:

Or is represented by a circle carrying the word “OR”, with two connectors located at the opposite side for inputs and output.

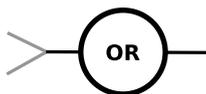


Figure 2.25: *The Activity Flow glyph for or. Only two inputs are represented, but more would be allowed.*

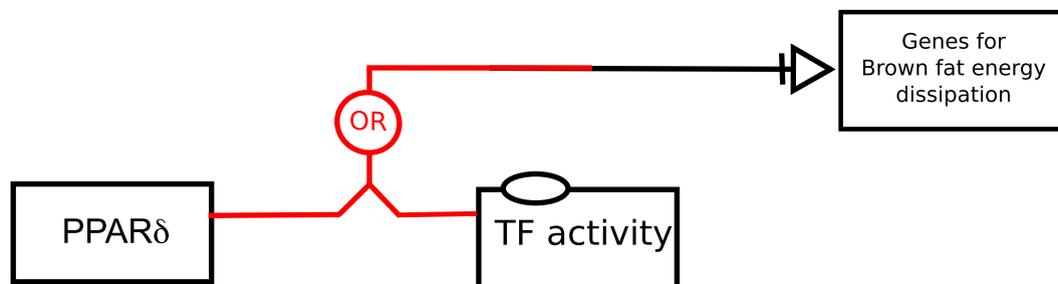


Figure 2.26: An example, taken from Figure 2.1 on page 4, of the or logic operator, where the activity from "Genes for brown fat energy dissipation" is stimulated by either the PPAR delta activity or an unspecified transcription factor activity.

2.9.3 Glyph: *Not*

The glyph *not* is used to denote that the *AN* linked as input cannot influence the target activity.

SBO Term:

SBO:0000238 ! not.

Origin:

One *biological activity* (Section 2.3.1) or and a *logical arc* (Section 2.8.5).

Target:

A *modulation arc* (Section 2.3) other than *equivalence arc*.

Node:

Not is represented by a circle carrying the word "NOT", with two connectors located at the opposite side for inputs and output.



Figure 2.27: The Activity Flow glyph for not.

2.9.4 Glyph: *delay*

The glyph *delay* is used to denote that the *activity node* linked as input does not produce the influence immediately.

SBO Term:

SBO:0000225 ! delay.

Origin:

One *biological activity* (Section 2.3.1) or and a *logical arc* (Section 2.8.5).

Target:

A *modulation arc* (Section 2.3) other than *equivalence arc*.

Container:

Delay is represented by a circle, with two connectors located at the opposite side for inputs and output.

Label:

Delay is identified by the greek letter “ τ ” (“TAU”) placed in an unbordered box attached to the center of the node.

Auxiliary items:

Delay does not carry any auxiliary items.

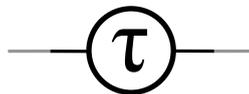


Figure 2.28: *The Activity Flow glyph for delay.*

Chapter 3

Activity Flow language grammar

3.1 Overview

In this chapter, we describe how the glyphs of SBGN Activity Flow can be combined to make a valid SBGN Activity Flow map. To do this, we must at the very least define what glyphs can be connected to each other. This is called syntax. Next, we must define rules over and above connection rules, such as whether duplicate symbols are permitted. In addition, we must define what the notation “means” – how does it represent a biological pathway? This is semantics, and it is essential if a reader is to understand a SBGN map without external help, and a writer is to create one that reflects his understanding of a biological system.

In this section we start off by describing the concepts of the Activity Flow notation. Next a detailed description of the syntax is provided followed by a description of the semantic rules of the notation.

3.2 Concepts

The SBGN Activity Flow is more than a collection of symbols. It is a visual language that uses specific abstractions to describe the biological activities that make up a model, a signaling pathway or a metabolic network. This abstraction is the semantics of SBGN, and to describe it requires more than a definition of the symbols and syntax of the language.

The Activity Flow in SBGN describes biological activities involving biological entities. A *biological activity* can influence, or be influenced by, other *biological activities*, and such relationships are represented in Activity Flow by lines with arrows and other decorations. So the essence of Activity Flow is to show the flow of activities from one entity to another or within the same entity. The underlying mechanisms of how the influence occurs may not be known and is not captured in the diagram. If the mechanism is known, the details should be described in annotation or captured in other SBGN languages, such Process Description and/or Entity Relationship.

3.3 Syntax

The syntax of the SBGN Activity Flow language is defined in the form of an incidence matrix. An incidence matrix has arcs as rows and nodes as columns. Each element of the matrix represents the role of an arc in connection to a node, as described below.

- Input (I) means that the arc can begin at that node.
- Output (O) indicates that the arc can end at that node.
- Numbers in parenthesis represent the maximum number of arcs of a particular type to have this specific connection role with the node.
- Empty cells means the arc is not able to connect to the node.

3.3.1 Activity Nodes connectivity definition

<i>Arc\Node</i>	<i>biological activity</i>	<i>phenotype</i>	<i>tag</i>	<i>submap (terminal)</i>	<i>and</i>	<i>or</i>	<i>not</i>	<i>delay</i>
<i>positive influence</i>	I & O	O			I(1)	I(1)	I(1)	I(1)
<i>negative influence</i>	I & O	O			I(1)	I(1)	I(1)	I(1)
<i>unknown influence</i>	I & O	O			I(1)	I(1)	I(1)	I(1)
<i>necessary stimulation</i>	I & O	O			I(1)	I(1)	I(1)	I(1)
<i>logic arc</i>	I				O	O	O(1)	O(1)
<i>equivalence arc</i>	I	I	O	O				

3.3.2 Containment definition

Container (compartment) is a node type allowing containment of other elements of AF in SBGN. The next table describes relationship between AF elements of SBGN and the compartment. Plus sign (+) means that the element is able to be contained within a node. An empty cell means containment is not allowed.

<i>elements\Containers</i>	<i>compartment</i>
<i>biological activity</i>	+
<i>phenotype</i>	+
<i>tag</i>	+
<i>compartment</i>	
<i>submap</i>	+
<i>positive influence</i>	+
<i>negative influence</i>	+
<i>unknown influence</i>	+
<i>logic arc</i>	+
<i>equivalence arc</i>	+
<i>and</i>	+
<i>or</i>	+
<i>not</i>	+
<i>delay</i>	+

3.3.3 Syntactic rules

There are additional syntactic rules that must be applied in addition to those defined above.

3.3.3.1 ANs

1. Each *biological activity* has at most one *unit of information*.
2. *Phenotype* is never origin of a modulation arc.

3.4 Semantic rules

3.4.1 Activity node and Compartment

Each AN can only appear once in a particular compartment. An AN can only *belong* to one compartment. However, an AN can be *drawn* over more than one compartment. In such cases the decision on which is the owning compartment is deferred to the drawing tool or the author.

The layout of compartments in an SBGN diagram does not imply anything about the topology of compartments in the cell. Compartments should be bounded and may overlap. However, adjacency and the nesting of compartments does not imply that these compartments are next to each other physically or that one compartment contains the other.

3.4.2 Modulation

It is implied, but not defined explicitly that an activity has a rate at which the input AN effects the output AN.

1. Positive influence is a modulation that's effect is to increase the basal activity.
2. Negative influence is a modulation that's effect is to decrease the basal activity.
3. Unknown influence is a modulation where the effect and basal activity is unknown.
4. At most one necessary stimulation can be assigned to an activity. Two necessary stimulations would imply an implicit Boolean AND or OR operator. For clarity only one necessary stimulation can be assigned to an activity node and such combinations must be explicitly expressed as the Boolean operators.

3.4.3 Submaps

Submaps are a visual device that allow a map to be split into several views. They remain, however, part of the main map and share its namespace. As a test of validity it should be possible to reintroduce a submap into the main map by eliminating the submap *terminals* and merging the equivalent nodes in both maps.

3.4.3.1 Rules for mapping to submaps

An AN in the main map can be mapped to one in the submap using a *tag* in the submap and *terminals* (see Section 2.7) in the main map. For a mapping between map and submap to exist the following must be true:

1. The identifiers in the *tags* and *terminals* must be identical.
2. The ANs must be identical.

3.4.3.2 Requirement to define a mapping

If a map and submap both contain the same AN, then a mapping between them must be defined as above.

Chapter 4

Layout Guidelines for an Activity Flow Maps

4.1 Introduction

The previous chapters describe the appearance and meaning of SBGN Activity Flow Level 1 components. Objects are *activity nodes*, *container nodes*, *logical operators*, *submaps* as well as *connecting arcs*. The components of an Activity Flow map have to be placed in a meaningful way – a random distribution with spaghetti-like connections will most likely hide the information encoded in the underlying model, whereas an elegant placement of the objects, giving a congenial appearance of the maps, may reveal new insights. The arrangement of components in a map is called a *layout*.

SBGN Activity Flows should be easily recognisable not only by the glyphs used, but also by the general style of the layout. However, the arrangement of the components is a complex art in itself, and there is no simple rule which can be applied to all cases. Therefore this section provides guidelines for the layout of Activity Flow maps, divided into two categories:

1. requirements, i.e., rules which **must** be fulfilled by a layout, and
2. recommendations, i.e., rules which **should** be followed if possible.

In addition, we provide a list of additional suggestions which may help in producing aesthetically more pleasant layouts, possibly easier to understand.

Those layout guidelines are independent of the method used to produce the map, and apply to both manually drawn maps as well as maps produced by an automatic layout algorithm. The guidelines do not deal with interactive aspects (e.g., the effect of zooming). Further information about automatic network layout (graph drawing) can be found, for example, in the books of Di Battista and co-authors [3] and Kaufmann and Wagner [4].

Please note that the color of objects do not carry any meaning in SBGN. Although one can use colors to emphasize part of a diagram or encode additional information, the meaning of the diagram should not depend on the colors. Furthermore, objects can have different sizes and size is also meaningless in SBGN. For example, one biological activity node may be larger than another node. Also the meaning of a graph should be conserved upon scaling as far as possible.

4.2 Layout guidelines

4.2.1 Requirements

Requirements are rules which **must** be fulfilled by a layout to produce a valid SBGN Activity Flow Level 1 map.

4.2.1.1 Node-node overlaps

Nodes are only allowed to overlap in two cases:

1. the overlapping nodes define a glyph (e.g., *auxiliary unit*).
2. nodes overlapping compartments (e.g., a *biological activity* placed on a compartment).
3. compartment overlapping compartment (e.g., Figure 2.10 on page 12). However, it should be noted that it does not have implication of containment of one compartment to the other.

Otherwise, nodes are not allowed to overlap (Figure 4.1). This includes the touching of nodes, which is also not allowed. Also submaps are not allowed to overlap.

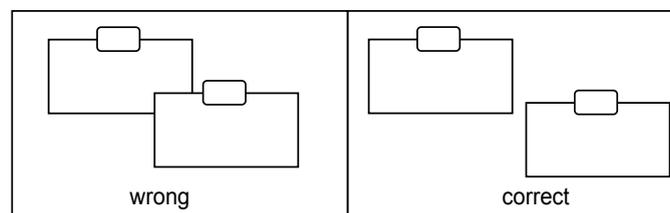


Figure 4.1: Nodes must not overlap.

4.2.1.2 Node-edge crossing

In general, such crossing should be avoided. In case this can't be avoided, the edge must be drawn on the top of the node (Figure 4.2). See also recommendation 4.2.2.1

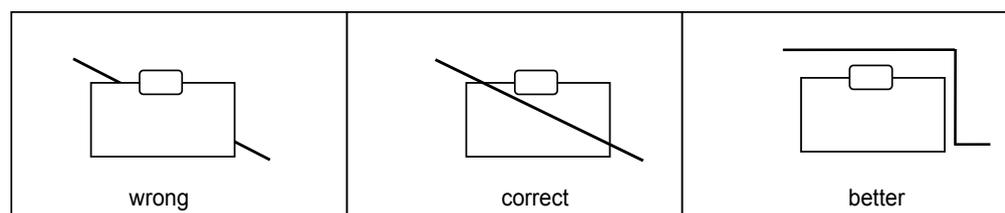


Figure 4.2: If an edge crosses a node, the edge must be drawn on top of the node.

4.2.1.3 Node border-edge overlaps

Edges are not allowed to overlap the border lines of nodes (Figure 4.3 on the following page).

4.2.1.4 Edge-edge overlaps

Edges are not allowed to overlap (Figure 4.4 on the next page). This includes touching of edges. Furthermore, an edge is neither allowed to cross itself nor to cross a boundary of node more than twice or other edges more than once.

4.2.1.5 Node orientation

Nodes have to be drawn horizontally or vertically, any other rotation of elements is not allowed (Figure 4.5 on the following page).

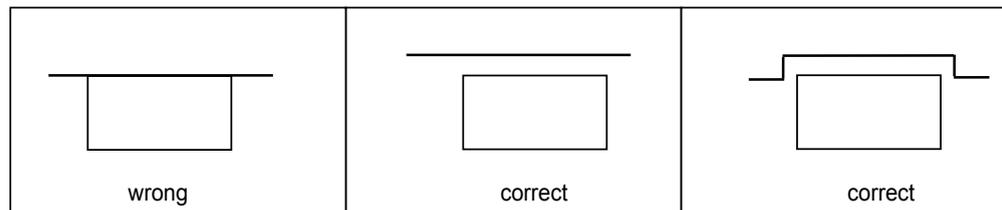


Figure 4.3: Edges must not overlap node borders.

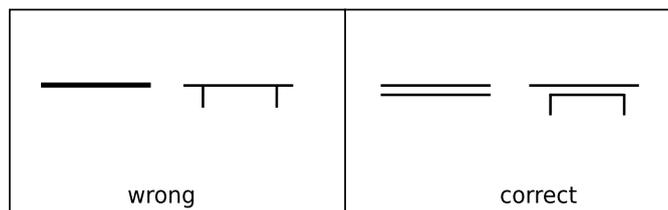


Figure 4.4: Edges must not overlap.

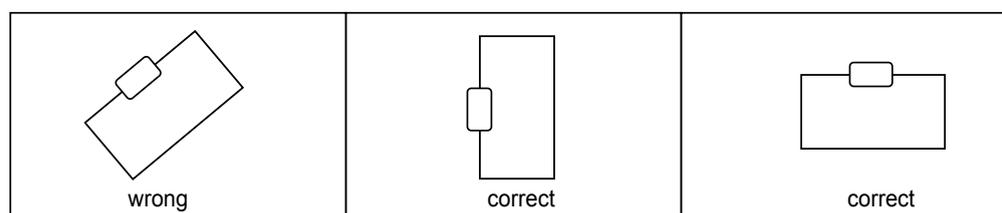


Figure 4.5: The node orientation must be horizontally or vertically.

4.2.1.6 Node labels

At least a part of the label (unbordered box containing a string of characters) has to be placed inside the node it belongs to. Node labels are not allowed to overlap nodes or other labels (this includes touching of other nodes or labels).

4.2.1.7 Edge labels

Edge labels are not allowed to overlap nodes. This includes touching of nodes.

4.2.1.8 Compartments

If a network has all participants (nodes and edges) in the same compartment, all the activity nodes and edges/arcs have to be in this compartment.

Edges/arcs are allowed to cross the compartment boundaries when the input and output ANs are in two different compartments.

4.2.2 Recommendations

Recommendations are rules which should be followed if possible to produce layouts that are easier to understand.

4.2.2.1 Node-edge crossing

Crossings between edges and nodes should be avoided. Some crossings may be unavoidable, e.g., the crossing between an edge and a compartment border. See also requirement 4.2.1.2 (in case of node-edge crossings the edge must be drawn on the top of the node).

4.2.2.2 Labels

Labels should be horizontal. Node labels should be placed completely inside the node if possible. Edge labels should be placed close to the edge and avoid overlapping the edge as well as other edge labels.

4.2.2.3 Avoid edge crossings

The amount of crossings between edges should be minimized.

4.2.3 Additional suggestions

Here is a list of additional layout suggestions which may help in producing aesthetically more pleasing layouts which may be easier to understand.

- Angle of edge crossings: If edge crossings are not avoidable edges should cross with an angle close to 90 degrees.
- Drawing area and width/height ratio: The drawing should be compact and the ratio between the width and the height of the drawing should be close to 1.
- Edge length: Long edges should be avoided if possible.
- Number of edge bends: Edges should be drawn with as few bends as possible.
- Similar and symmetric parts: Similar parts of a map should be drawn in a similar way, and symmetric parts should be drawn symmetrically.
- Proximity information: Related elements (e.g., nodes connected by edges within a compartment) should be drawn close together.
- Directional information: Subsequent activities (e.g., a sequence of activities) should be drawn in one direction (e.g., from top to bottom or from left to right).
- Compartments: Different compartments should have different background shade or color.

Chapter 5

Acknowledgments

Here we acknowledge those people and organisations that assisted in the development of this and previous releases of the SBGN Activity Flow language specification. First we specifically acknowledge those who contributed directly to each revision of the specification document, followed by a comprehensive acknowledgement of contributors that attended workshops and forum meetings or in some other way provided input to the standard. Finally, we acknowledge the bodies that provided financial support for the development of the standard.

We aim this list to be rather complete. We are very sorry if we forgot someone, and will be grateful if you notified us of any omission.

5.1 Level 1 Release 1.0

The specification was written by Huaiyu Mi, Falk Schreiber, Nicolas Le Novère, Stuart Moodie, Anatoly Sorokin, Michael Hucka, Emek Demir, Yukiko Matsuoka, Katja Wegner and Hiroaki Kitano. In addition, the specification benefited much from the help of Frank Bergmann, Sarala Dissanayake and Paul Thomas.

5.2 Level 1 Release 1.1

The specification of Activity Flow Level 1 Release 1.1 was written by Huaiyu Mi, with significant contributions from Falk Schreiber and Stuart Moodie. Other contributors include (in alphabetic order) Tobias Czauderna, Emek Demir, Nicolas Le Novère, Yukiko Matsuoka, Anatoly Sorokin, Alice Villéger and Katja Wegner.

5.3 Level 1 Release 1.2

The specification of Activity Flow Level 1 Release 1.2 was written by Huaiyu Mi, with significant contributions from Falk Schreiber, Stuart Moodie, Tobias Czauderna and Anatoly Sorokin. Other contributors include (in alphabetic order) Michael Blinov, Emek Demir, Robin Haw, Anushya Muruganujan, Nicolas Le Novère, Augustin Luna, Paul Thomas and Katja Wegner.

5.4 Comprehensive list of acknowledgements

Here is a more comprehensive list of people who have been actively involved in SBGN development, either by their help designing the languages, their comments on the specification, help with development infrastructure or any other useful input. We aim this list to be rather complete. We are very sorry if we forgot someone, and will be grateful if you notify us of any omission.

Mirit Aladjem, Frank Bergmann, Michael Blinov, Bernard de Bono, Sarah Boyd, Laurence Calzone, Melanie Courtot, David Croft, Tobias Czauderna, Emek Demir, Johannes W. Dietrich, Ugur Dogrusoz, Damien Fleury, Tom Freeman, Akira Funahashi, Ralph Gauges, Peter Ghazal,

Samik Ghosh, Igor Goryanin, Anja Hartmann, Robin Haw, Michael Hucka, Matthias Jeschke, Mathias John, Akiya Jouraku, Astrid Junker, Hideya Kawaji, Douglas Kell, Sohyoung Kim, Hiroaki Kitano, Christian Klukas, Kurt Kohn, Fedor Kolpakov, Nicolas Le Novère, Lu Li, Augustin Luna, Yukiko Matsuoka, Carsten Maus, Alexander Mazein, Huaiyu Mi, Stuart Moodie, Ulrike Münzner, Anushya Muruganujan, Michael Pedersen, Jacqueline Quinn, Stefan Rybacki, Sven Sahle, Chris Sander, Herbert Sauro, Esther Schmidt, Falk Schreiber, Jacky Snoep, Anatoly Sorokin, Jessica Stephens, Linda Taddeo, Carolyn Talcott, Lin Uhrmacher, Martijn van Iersel, Alice Villéger, Steven Watterson, Katja Wegner (Wengler), Sarala Wimalaratne (Dissanayake), Guanming Wu, Röbbbe Wünschiers.

The authors are also grateful to all the attendees of the SBGN meetings, as well as to the subscribers of the sbgn-discuss@sbgn.org mailing list.

5.5 Financial Support

The development of SBGN was mainly supported by a grant from the Japanese New Energy and Industrial Technology Development Organization (NEDO, <http://www.nedo.go.jp/>). The Okinawa Institute of Science and Technology (OIST, <http://www.oist.jp/>), the AIST Computational Biology Research Center (AIST CBRC, <http://www.cbrc.jp/index.eng.html>), the British Biotechnology and Biological Sciences Research Council (BBSRC, <http://www.bbsrc.ac.uk/>) through a Japan Partnering Award, the European Media Laboratory (EML Research gGmbH, <http://www.eml.org/english/>), the Beckman Institute at the California Institute of Technology (<http://bnmc.caltech.edu>), Ontario Institute for Cancer Research (OICR, <http://oicr.on.ca/>), Ontario Genomics Institute (OGI, <http://www.ontariogenomics.ca/>), National Science Foundation (NSF, <http://www.nsf.gov/>), USC Norris Comprehensive Cancer Center (<http://uscnorriscancer.usc.edu/>), Martin Luther University Halle-Wittenberg (<http://www.uni-halle.de/>), Monash University (<http://www.monash.edu.au/>), IPK Gatersleben (<http://www.ipk-gatersleben.de/en/>), University of Rostock (<http://www.uni-rostock.de/>) and German Federal Ministry of Research and Education (<http://www.bmbf.de/>) provided additional support for SBGN workshops. Some help was provided by the Japan Science and Technology Agency (JST, <http://www.jst.go.jp/>) and the Genome Network Project of the Japanese Ministry of Education, Sports, Culture, Science, and Technology (MEXT, <http://www.mext.go.jp/>) for the development of the gene regulation network aspect of SBGN, and from the Engineering and Physical Sciences Research Council (EPSRC, <http://www.epsrc.ac.uk>) during the redaction of the specification.

Appendix A

Examples

The following diagrams present examples of SBGN Activity Flow diagrams representing biological activities and their influences among each other in pathway networks. They by no mean exhaust the possibilities of SBGN Activity Flow Level 1.

Figure A.1 presents an example of a signaling pathway involving the regulation of TGF β -induced metastasis. The pathway was described in a report titled "A Mutant-p53/Smad Complex Opposes p63 to Empower TGF β -Induced Metastasis" in the 2009 April issue of Cell [5]. The figure shows the usage of *biological activity nodes*, *phenotype*, *positive influence arc*, *negative influence arc*, *necessary stimulation arc*, and *logic operator*.

Figure A.2 on the following page presents a more complicated example of signaling pathway involving the intracellular signaling through the epidermal growth factor receptor (EGFR). This example is a redraw of the Epidermal Growth Factor Receptor Pathway described in the

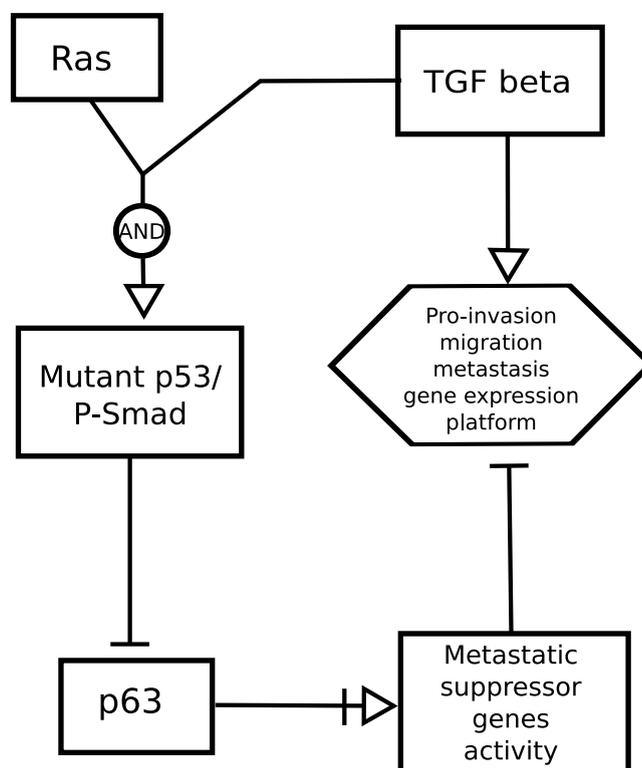


Figure A.1: Regulation of TGF β -induced metastasis.

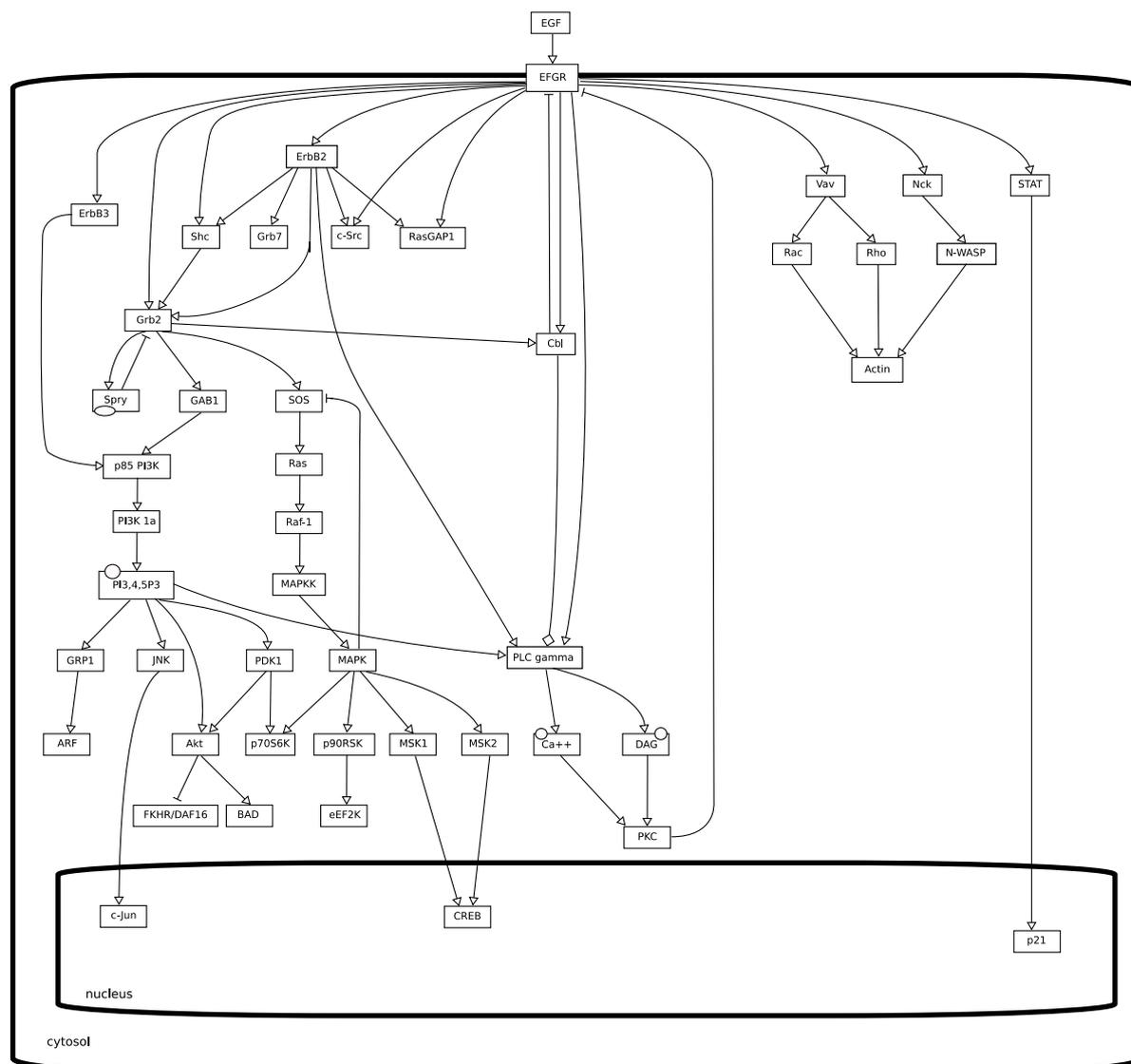


Figure A.2: Epidermal Growth Factor Receptor Pathway.

Signal Transduction Knowledge Environment (http://stke.sciencemag.org/cm/stkecm;CMP_14987).

Figure A.3 on the next page shows the transforming growth factor beta (TGF β) signaling pathway. The map is a redraw of the TRG-beta Signaling Pathway described in the PANTHER Pathway System (<http://www.pantherdb.org/pathway/pathwayDiagram.jsp?catAccession=P00052>) and is based on reviews by Massague [6] and Derynck [7].

Figure A.4 on page 34 presents the simplest view of action potential propagation mediated by the voltage-gated sodium channels. There are two views how voltage-gated ion channels are involved. The diagram on the left side shows that the *increase in membrane potential* activates *voltage-gated sodium channel activity*, which in turn triggers membrane *depolarization*. The membrane depolarization activates *voltage-gated potassium channel activity*, which results in repolarizing the membrane potential. The diagram on the right side provides more detail in the mechanism. It shows that the *increase in membrane potential* first activates the *gating activity* of the channel, which in turn activates the *conductance activity* leading to the membrane

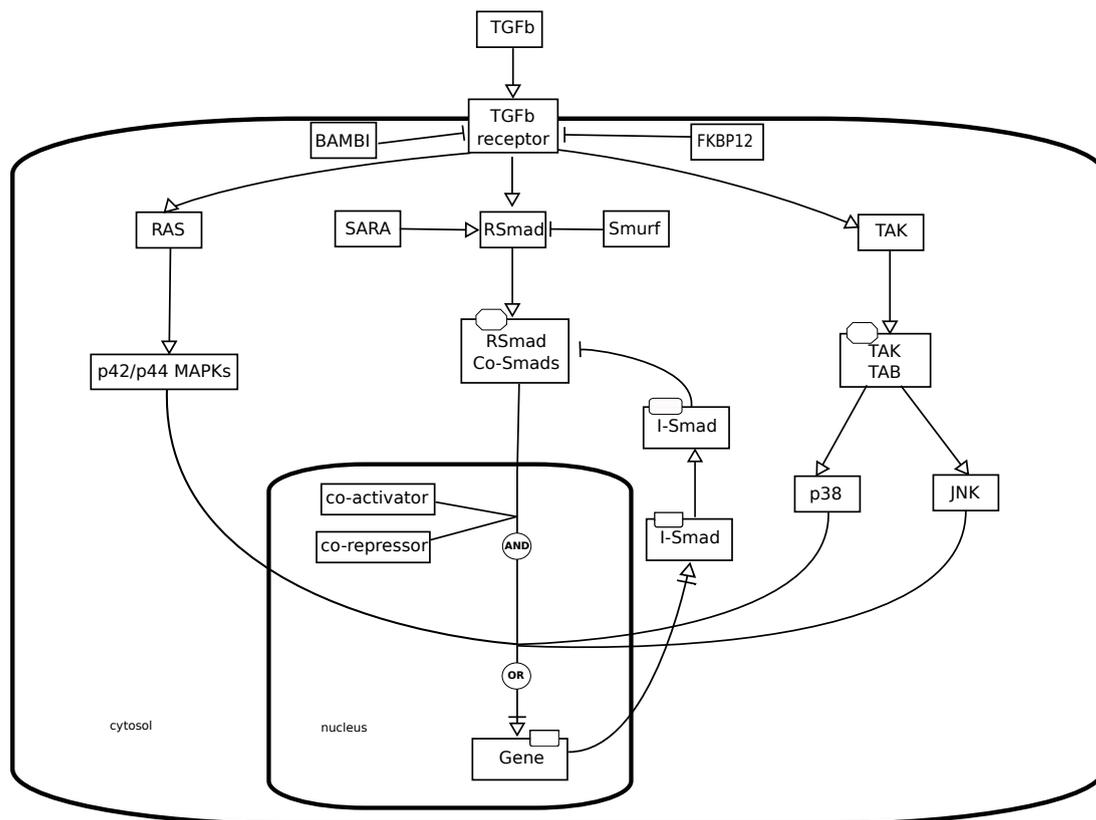


Figure A.3: Transforming Growth Factor beta signaling pathway.

depolarization. In this case, both *gating activity* and *conductance activity* come from the sodium channel, which is indicated as the *unit of information*. In addition, this example also shows the advantage of using Activity Flow maps, because certain activities, such as gating and conductance, come from a few amino acids in particular three dimensional structure that is not able to be illustrated in either Process Description or Entity Relationship maps.

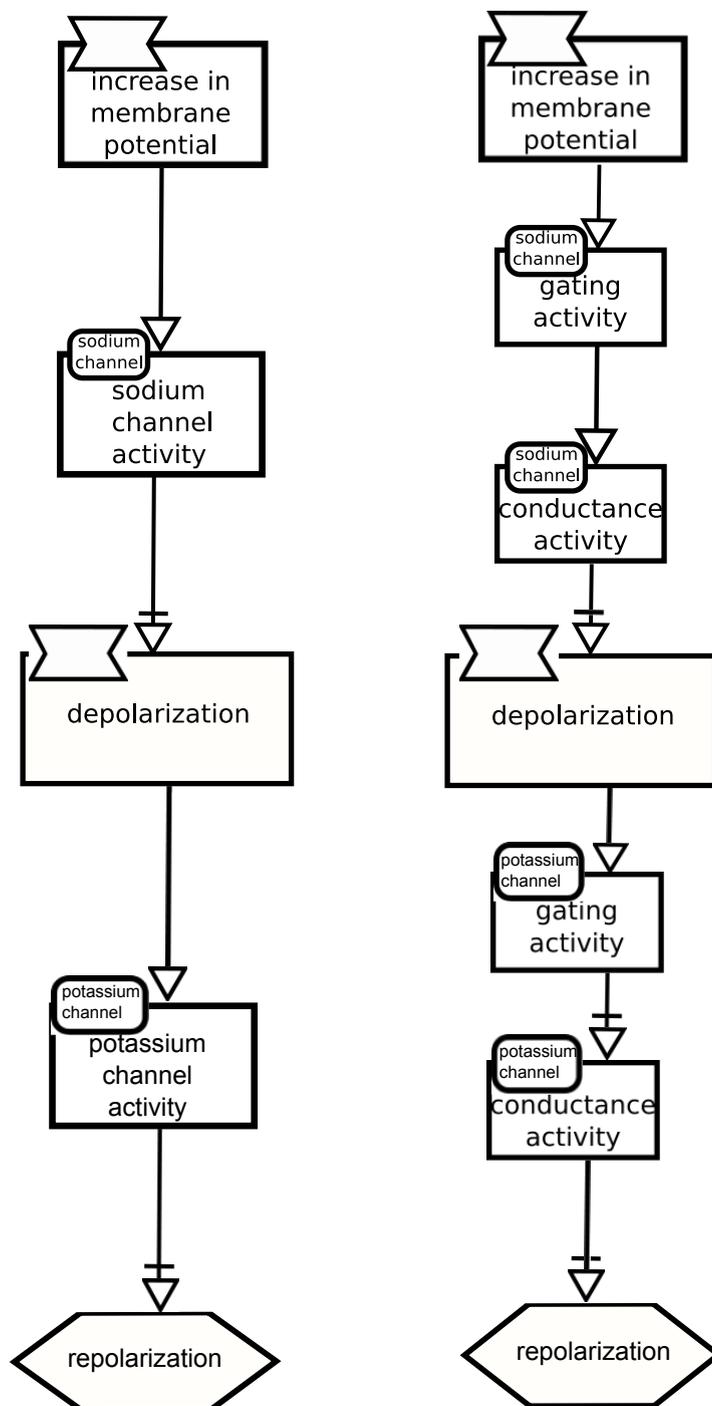


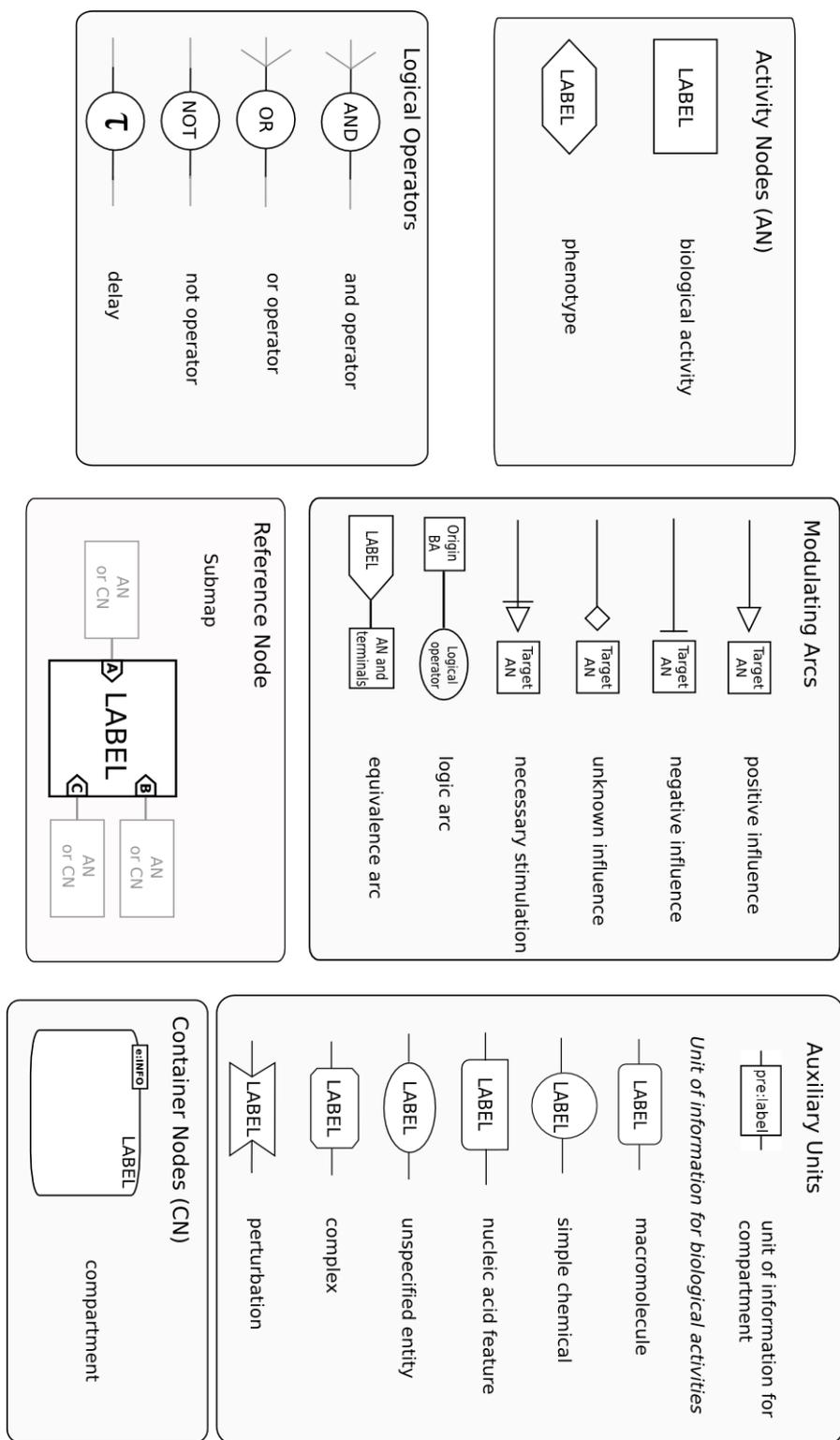
Figure A.4: Two views of the role voltage-gated sodium channel plays in action potential generation illustrated by SBN Activity Flow Level 1.

Appendix B

Reference card

Print this summary of SBGN Activity Flow symbols for a quick reference.

SYSTEMS BIOLOGY GRAPHICAL NOTATION ACTIVITY FLOW DIAGRAM REFERENCE CARD



Appendix C

Revision History

C.1 Version 1.0 to Version 1.1

There are three major changes in glyphs in Version 1.1 of the SBGN Activity Flow Level 1 specification.

1. Add *unit of information* to the compartment. This is done in accordance to SBGN Process Description Level 1 specification.
2. Remove *perturbation* from activity node, and add it to *unit of information* to decorate a *biological activity* node. The change was based on a survey conducted August 2011.
3. Add *process* as an activity node. This is rather controversial at the beginning. The issue was to make a process both an origin and a target of arcs. It was not possible in AF V. 1.0 because a *phenotype* cannot be the origin of an arc. The details of this issue and subsequent survey (August 2011) is described on the SBGN website http://www.sbgn.org/AF_phenotype. The survey did not yield a concrete solution. Through additional discussion on the SBGN discuss list, this seems to be the best solution for the purpose without violating semantic rules.

Version 1.1 was never officially released due to some disagreements in the change, especially in the use of process. It served as a working draft only.

C.2 Version 1.1 to Version 1.2

There are two changes in Version 1.2 of the SBGN Activity Flow Level 1 specification.

1. Remove the *process* node proposed in Version 1.1.
2. Modify Chapter 1 to make the text consistent with those in the SBGN Process Description Level 1 specification
3. Modify the compartment overlap rule. Removed the requirement of two lines separate the overlapping compartment in the earlier version of spec.
4. Fix bugs caused by the removal of *perturbation* glyph.

Description	Tracker ID
Add <i>Unit of information for Compartment</i> , section 2.4.2	
Add <i>Unit of information for Compartment</i> to the ref card (Appendix B)	
Remove <i>perturbation</i> from Activity node section (2.3)	
Add <i>perturbation</i> to the unit of information section (2.4.1)	
Modify <i>perturbation</i> glyph in figures 2.1, 2.5, 2.17, A.4 and refcard accordingly (Appendix B).	
Add <i>process</i> to the Activity node section (2.3)	
Modify figure A4 by adding the <i>process</i> glyph.	
Modify the description for <i>biological activity</i> in section 2.3.1 to clarify that it is for molecular activity, so to differentiate it from the activities from <i>process</i> .	
Modify the description for <i>phenotype</i> in section 2.3.3.	
Modify tables in section 3.3.1 and 3.3.2 - Remove <i>perturbation</i> - Add <i>process</i>	
Add <i>phenotype</i> as input for <i>equivalent arc</i> in section 3.3.1	
Remove the sentence "If an activity has participants in at least two different compartments, the activity node has to be either in a compartment where the activity has at least one participant or in the empty space." in section 4.2.1.8	
In section 4.2.1.8, change the first sentence "If an activity has all participants ..." to "If a network has all participants..."	
In section 4.2.1.8, add a sentence "Edges/arcs are allowed to cross the compartment boundaries when the input and output ANs are in two different compartments."	
Add <i>delay</i> in the line of Logical operators in Table 2.1	
Correct figure legend for Figure 2.10. The example on the right side shows an incorrect representation.	
In section 2.2, controlled vocabularies, clarify that the terms are to describe the unit of information of the ANs	
Rewrite figure legends with more details throughout section 2	
Modify Figure A3 in Appendix A (TGF beta signaling pathway) by changing the <i>Gene transcription</i> phenotype node to an biological activity node. However, it should be discussed whether phenotype can be an input or not.	2992312
Update SBO term (SBO:0000255 to SBO:0000002) in section 2.2.3	3068940

Bibliography

- [1] Nicolas Le Novère, Michael Hucka, Huaiyu Mi, Stuart Moodie, Falk Schreiber, Anatoly Sorokin, Emek Demir, Katja Wegner, Mirit I. Aladjem, Sarala M. Wimalaratne, Frank T. Bergman, Ralph Gauges, Peter Ghazal, Hideya Kawaji, Lu Li, Yukiko Matsuoka, Alice Villéger, Sarah E. Boyd, Laurence Calzone, Melanie 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):735–41, 8 2009.
- [2] D. Pan, M. Fujimoto, A. Lopes, and Y. Wang. Twist-1 is a PPAR δ -inducible, negative-feedback regulator of PGC-1 α in brown fat metabolism. *Cell*, 137(1):73–86, 2009.
- [3] G. Di Battista, P. Eades, R. Tamassia, and I. G. Tollis. *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice Hall, New Jersey, 1998.
- [4] M. Kaufmann and D. Wagner. *Drawing Graphs: Methods and Models*, volume 2025 of *Lecture Notes in Computer Science Tutorial*. Springer, 2001.
- [5] M. Adorno, M. Cordenonsi, M. Montagner, S. Dupont, C. Wong, B. Hann, A. Solari, S. Bobisse, M. B. Rondina, V. Guzzardo, A. R. Parenti, A. Rosato, S. Bicciato, A. Balmain, and S. Piccolo. A mutant-p53/smad complex opposes p63 to empower TGFbeta-induced metastasis. *Cell*, 137(1):87–98, 2009.
- [6] J. Massague. TGF-beta signal transduction. *Annual Review Biochemistry*, 67:753–791, 1998.
- [7] R. Derynck, R. J. Akhurst, and A. Balmain. TGF-beta signaling in tumor suppression and cancer progression. *Nature Genetics*, 29(2):117–129, 2001.