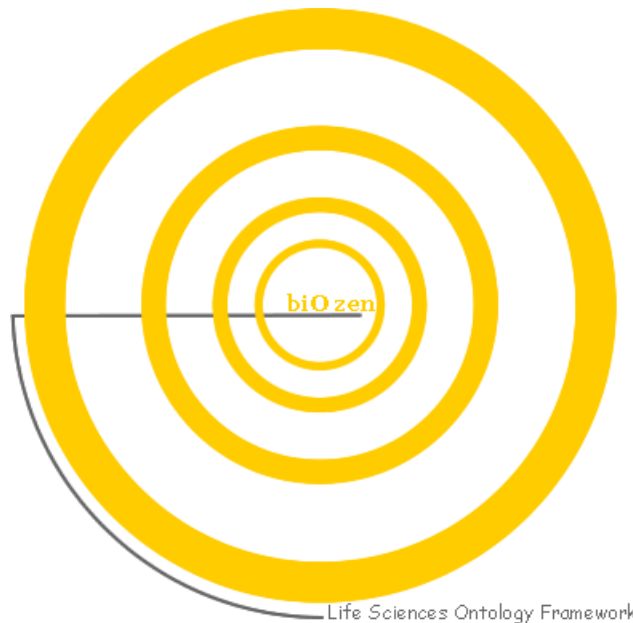


The *bio-zen* Semantic Web framework

Unifying the life sciences on the Semantic Web



Author: Matthias Samwald

Contact: samwald@neuroscientific.net

More information and downloads: <http://neuroscientific.net/index.php?id=semantic>

Discussion forum: http://neuroscientific.net/index.php?id=forum&view=single_conf&cat_uid=4&conf_uid=10

All of the developments described here are still a work in progress and need feedback (from you!). The first stable release is scheduled for September 2006.

1. Introduction

The development of the *bio-zen ontology framework* is an attempt to represent data, information and knowledge from research in all facets of the *life sciences* on the Semantic Web. The goal of this project is the unification of information that is now scattered through a multitude of different data structures, exchange formats and databases. Through the use of Semantic Web technologies, the decentralised and barrier-free development and exchange of experimental data, hypotheses and biological models becomes possible.

Conventional databases (e.g. relational databases or XML databases) do a poor job of representing biological reality. Researchers that want to publish or search for information do not only have to know about the biological structures they are investigating, they also have to deal with the structures of the database tables, file formats or

XML documents – all of which are in most cases only remotely similar to the mental representation we have in mind when thinking about biological facts.

Furthermore, most databases in use nowadays were designed with only a small and limited field of investigation in mind, and so we are now confronted with a convolute of small databases that can only be made to work together through a lot of additional work.

We also see that *systems biology* with its focus on the simulation of complex biological systems has an ever growing impact on classical molecular biology. However, the realm of **qualitative information** that is represented in texts and databases like *Uniprot* or *BIND* is completely disconnected from the world of **simulation and modelling**, which is currently represented with languages like SBML, CellML or NEURON models. To realize the promises of systems biology, the division between these two worlds has to be bridged.

The *bio-zen* framework uses Semantic Web technologies to overcome the limitations of current information systems in the life sciences. The descriptions of biological reality in the *bio-zen* framework are very similar to the cognitive models researchers have about their subjects of investigation, making the work with information systems more intuitive for the individual scientist. *bio-zen* is exceptionally flexible and extensible, making it easy to represent information from a wide variety of fields in a common framework. It also allows for a seamless integration of mathematical descriptions and simulation parameters into qualitative information, enabling a quick transition from data and information to model simulations and back.

Bio-zen is designed to be very agile and open for collaborative participation and extension of information bases while retaining full logical consistency. It allows for the distributed creation of uncontrolled vocabularies (so-called *folksonomies*¹). In contrast to controlled vocabularies, such folksonomies are open-ended and can therefore respond quickly to changes and innovations in the way researchers categorize their observations. The philosophy behind this loosely controlled, collaborative annotation is similar to that of other peer production systems such as Wikipedia² or Nature's Connotea³. Although the participating individuals possess varying levels of tagging sophistication, such a production process can produce results that compare favourably to professionally curated, centralised systems.

If successful, the development of Semantic Web infrastructure could mark the beginning of a whole new paradigm in the organisation and dissemination of information in the life sciences.

1.1. Recommended reading

To understand the motivations, philosophies and visions behind the Semantic Web effort:

[‘The Semantic Web’ by Tim Berners-Lee, James Hendler and Ora Lassila](#)

[‘RDF – The Web’s Missing Link’ by Eric Neumann](#) ⁴

Introductions to the Semantic Web standards RDF and OWL:

[‘RDF primer’ from the World Wide Web Consortium](#) ⁵

¹ See <http://en.wikipedia.org/wiki/Folksonomy>

² See <http://wikipedia.org>

³ See <http://www.connotea.org/>

⁴ See <http://www.bio-itworld.com/issues/2006/march/rdf/>

⁵ See <http://www.w3.org/TR/rdf-primer/>

1.2. Recommended software

To view and experiment with the ontology the Stanford **Protégé Ontology Editor** (and its OWL plugin) is recommended:

<http://protege.stanford.edu>

It can be used to view the main ontology and example files in this manual. It can also be used to enter data (albeit in a quite tedious way).

1.3. Nomenclature and conventions in this document

The terms ‘class’ (sometimes ‘OWL class’), ‘individual’ (sometimes ‘OWL individual’) and property, (sometimes ‘OWL property’) have a special meaning in RDF and OWL and are used as they are defined in the RDF and OWL manuals. Please note that contrary to RDF, instances of classes are not called ‘instances’, but ‘individuals’. We say that a certain individual ‘belongs’ to a certain class.

RDF subject-predicate-object triples (or sets of many triples) are written in the following notation:

```
<molecule-a> <binds-to> <molecule-b>
```

This triple can be read as ‘molecule A binds to molecule B’.

`molecule-a`, `binds-to` and `molecule-b` are the URIs of the resources used in the statement.

Sometimes, abbreviations are used to clarify the URI-namespace of a resource (for example, ‘`rdf:type`’ refers to the *type* property in the standard RDF namespace).

In this document, such triples are also sometimes sketched as graphs with nodes and edges. As a graph, a single triple would look like this:



Of course, graphs usually consist of many triples, forming a network of resources that are connected by their properties:



⁶ See <http://www.w3.org/TR/owl-guide/>

2. Basic structure of the ontology

One of the advantages of the *bio-zen* framework is that it is based on established foundational Semantic Web ontologies and metadata standards. This means that pre-existing tools and ontologies optimized for these standards can be used together with the *bio-zen* framework. This design principle stands in contrast to most other ontologies in the life sciences (e.g. BioPAX⁷, FuGO⁸, MGED ontology⁹), which are not based on such standards and re-invent the wheel again and again – thereby erecting unnecessary barriers to interoperability.

The ontology is designed to conform to the OWL DL standard, which guarantees computability (the possibility to use automated reasoning software) and eases the development of tools that work with the ontology.

Some parts of the current version of *bio-zen* have been derived from the BioPAX ontology. Its development would not have been possible without the enormous effort of the BioPAX members.

The following is an overview of the most important classes defined in the core *bio-zen* ontology, the DOLCE ontology and SKOS.

2.1.1. The root classes: *spatio-temporal-particular* versus *abstract*

In *bio-zen* (and DOLCE, the foundational ontology *bio-zen* is based on), two fundamentally different classes of things are distinguished: *spatio-temporal-particulars* and *abstract* things. *Spatio-temporal-particulars* are concrete, real things that exist in a certain place in space and time. This is not the case with *abstract* things.

In *bio-zen*, the *abstract* class is also used for the uncomplicated representation of things in situations where the concise representation through *spatio-temporal-particulars* would be too complicated for current needs. In future versions of the ontology some of the things that are now done with *abstract* classes could be represented through more concise *spatio-temporal-particulars*.

2.1.2. The three kinds of *spatio-temporal-particulars*: *endurant*, *perdurant* and *quality* classes – and a bit of philosophy

The following three definitions are copied from the DOLCE ontology description.

Definition of an *endurant* thing:

“The main characteristic of endurants is that all of them are independent essential wholes. This does not mean that the corresponding property (being an endurant) carries proper unity, since there is no common unity criterion for endurants. Endurants can ‘genuinely’ change in time, in the sense that the very same endurant as a whole can have incompatible properties at different times. To see this, suppose that an endurant - say ‘this paper’ - has a property at a time t ‘it’s white’, and a different, incompatible property at time t' ‘it’s yellow’: in both cases we refer to the whole object, without picking up any particular part of it. Within endurants, we distinguish between physical and non-physical endurants, according to whether they have direct spatial qualities. Within physical endurants, we distinguish between amounts of matter, objects, and features.”

⁷ See <http://biopax.org>

⁸ See <http://fugo.sourceforge.net/>

⁹ See <http://mged.sourceforge.net/>

Examples for *endurants* are: a physical object, a non-physical object, a cell, a protein molecule, a hole in a cheese, the Eiffel tower.

Definition of a *perdurant* thing:

“Perdurants (AKA occurrences) comprise what are variously called events, processes, phenomena, activities and states. They can have temporal parts or spatial parts. For instance, the first movement of (an execution of) a symphony is a temporal part of the symphony. On the other hand, the play performed by the left side of the orchestra is a spatial part. In both cases, these parts are occurrences themselves. We assume that objects cannot be parts of occurrences, but rather they participate in them. Perdurants extend in time by accumulating different temporal parts, so that, at any time they are present, they are only partially present, in the sense that some of their proper temporal parts (e.g., their previous or future phases) may be not present. E.g., the piece of paper you are reading now is wholly present, while some temporal parts of your reading are not present yet, or any more. Philosophers say that endurants are entities that are in time, while lacking temporal parts (so to speak, all their parts flow with them in time). Perdurants, on the contrary, are entities that happen in time, and can have temporal parts (all their parts are fixed in time).”

Examples for *perdurants* are: an event, a process, molecular transport, molecular binding, exocytosis, the beating of a heart.

Definition of a *quality*:

“Qualities can be seen as the basic entities we can perceive or measure: shapes, colors, sizes, sounds, smells, as well as weights, lengths, electrical charges... ‘Quality’ is often used as a synonym of ‘property’, but this is not the case in this upper ontology: qualities are particulars, properties are universals. Qualities inhere to entities: every entity (including qualities themselves) comes with certain qualities, which exist as long as the entity exists.”

Examples for *qualities* are: the colour of an object, the pH of a solution, the speed of a moving train.

The following basic properties can connect things from these different classes:

Endurants can *participate* in perdurants.

Endurants can only have other endurants as *parts*, perdurants can only have other perdurants as *parts*.

Qualities can *inhere* in both endurants and perdurants.

2.1.3. The classes *molecule-population*, *molecular-complex-population* and their parts

The *bio-zen* ontology deals with molecules and molecular complexes in a way that might seem quite peculiar at the first glance. The ontology is focused on the description of *populations* of molecules, not on singular molecules themselves. This design choice was made because it reflects biological reality better in most occasions, and because it also makes modelling of stochastic processes much easier.

A *molecule-population* is a ‘mass’ of molecules located in a certain location (e.g. a cell, a vesicle, or the extracellular fluid of a certain organism). It is discouraged to pool molecules from distant locations (e.g. from two distinct organisms) into one *molecule-population*.

We can describe subparts of *molecule-populations* with the *part* property (see Figure 1). For example, we can define a sub-population of a *molecule-population* that is made up of a subgroup of its molecules (Figure 1, B). To do so, we can state

```

<molecule-population-123> <rdf:type> <biozen:molecule-population>
<subpopulation-of-molecule-population-1 > <rdf:type> <biozen:molecule-population>
<molecule-population-123> <dol:part> <subpopulation-of-molecule-population-123>

```

But how do we make statements about parts of the molecules themselves? For instance, we would like to talk about molecular domains or binding sites on proteins! To make statements about these things, we can create individuals that belong to the *population-of-parts* class. A *population-of-parts* is the 'mass' of certain, mutually somehow similar parts of molecules in a certain *molecule-population*.

For example, each molecule in a *molecule-population* of serotonin receptors contains a binding site for serotonin. We can define the mass of these binding sites as a *population-of-parts*, which is a part of the *molecule-population* of serotonin receptors. In RDF triples, this would look like:

```

<a-certain-serotonin-receptor-population> <rdf:type> <biozen:molecule-population>
  <serotonin-binding-site-population> <rdf:type> <biozen:population-of-parts>
<a-certain-serotonin-receptor-population> <dol:part> <serotonin-binding-site-population>

```

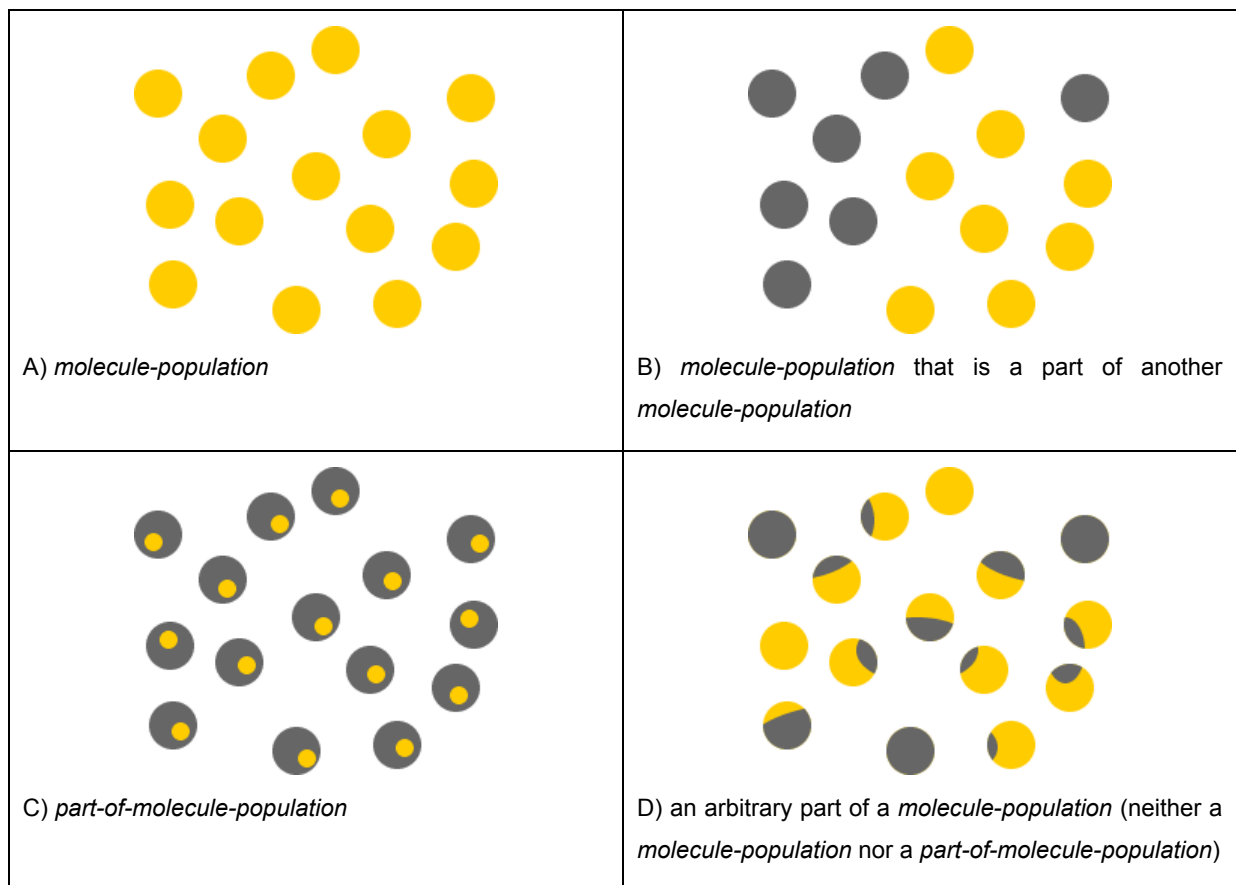


Figure 1: Any given *molecule-population* (sketched as a small group of molecules in *picture A*) can have different types of parts. *Picture B:* A part of a molecule-population can be a *molecule-population* in itself (which contains a subset of the molecules that make up the larger *molecule-population*). *Picture C:* a *part-of-molecule-population* can be understood as a population of 'parts of molecules', e.g. certain protein domains on a population of proteins. These parts-of-molecules are relatively uniform and are dispersed over the molecules of a *molecule-population*. *Picture D:* It is also possible (although discouraged) to define an arbitrary part of a *molecule-population* that is neither a *molecule-population* nor a *part-of-molecule-population*.

It is also possible to make statements about other, arbitrary parts of a *molecule-population*. In this case, of course, these OWL individuals do not belong to the *molecule-population* or *population-of-parts* classes (instead, they may be stated to belong to the generic class *physical-object*).

molecule-population is complemented by the *molecular-complex-population* class. The two classes are almost similar in their practical use – the main difference is that the latter is used to describe populations of molecular complexes, not molecules. A molecular complex is assembled from two or more molecules through non-covalent bonds.

In this ontology, it is **not** allowed to describe sub-complexes of molecular complexes. This information can be represented through the temporal order of complex assembly, though.

2.1.4. compartment, cell, tissue and organism classes

These classes are used to describe the most basic organisational forms of an organism. The names of these classes are more or less self-describing.

Of course it is possible to describe other kinds of parts of an organism on a different scale or granularity (e.g. organs or limbs). These structures should be defined as belonging to the generic *physical-object* class.

2.1.5. molecular-process class

Chemical reactions, binding events, complex formations etc. are represented as stochastic *processes* in *bio-zen*. *Molecule-populations* and *molecular-complex-populations* are participating in these processes, which are essentially made up of many tiny events (e.g. the binding and un-binding of many thousands of molecules of two *molecule-populations*). These tiny events are not modelled in *bio-zen*.

The generic class *molecular-process* has five sub-classes: *molecular-transport-process*, *molecular-binding-process*, *chemical-conversion-process*, *molecular-transport-with-chemical-conversion-process* and *metabolic-pathway*.

molecular-transport-process:

A process in which molecules of a certain molecule population change their location and become part of another molecule population (which is located elsewhere). Transporters are linked to transport interactions via the catalyzed-by property.

Synonyms: translocation.

Examples: The exocytosis of a neurotransmitter from a cellular vesicle into the synaptic cleft.

molecular-binding-process:

The process of the binding of molecules from different molecule populations to each other. Events of 'unbinding' (molecules that are bound lose their bonds) are also parts of such a process.

If you want to emphasise that an interaction results in the formation of a complex, you should consider using the subclass *complex-assembly-process* instead.

chemical-conversion-process:

A process in which molecules undergo covalent changes to become other molecules (thereby becoming part of another molecule population).

Examples: $\text{ATP} + \text{H}_2\text{O} = \text{ADP} + \text{P}_i$

Note: When writing biochemical reactions, it is not necessary to attach charges to the biochemical reactants or to include ions such as H⁺ and Mg²⁺ in the equation. Polymerization reactions involving large polymers whose structure is not explicitly captured should generally be represented as unbalanced reactions in which the monomer is consumed but the polymer remains unchanged, e.g. glycogen + glucose = glycogen.

metabolic-pathway:

A process that has several molecular interaction processes as its parts, often forming a network, which biologists have found useful to group together for organizational, historic, biophysical or other reasons.

Comment: It is possible to define a pathway without specifying the interactions within the pathway. In this case, the pathway instance could consist simply of a name and could be treated as a 'black box'.

Synonyms: network

Examples: glycolysis, valine biosynthesis, synthesis of serotonin from tryptophan.

2.1.6. About qualities

“Qualities inhere to entities: every entity (including qualities themselves) comes with certain qualities, which exist as long as the entity exists.”

The ontology distinguishes between physical qualities (qualities that inhere in endurants) and temporal qualities (qualities that inhere in perdurants).

Spatio-temporal-particulars are connected to qualities through the *quality* property. Viewed as a graph, the individuals that belong to the *quality* class surround the *spatio-temporal-particulars* (Figure 2). Numeric values are attached to these qualities (e.g. a *quality* describing the temperature of a solution can contain a floating point number that represents the temperature in Kelvin).

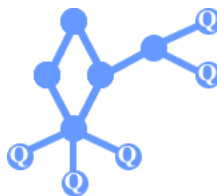


Figure 2: Individuals of the *quality* class (,Q') are attached to other individuals (e.g. physical objects) through the *has-quality* property.

All of the qualities use **SI units** (e.g. second, kilogram, mole, kelvin...) where applicable. This also means that, for example, the value for the length of a cell is given in meters! Of course, the resulting number will be very small, but this can be represented through floating point values. It is up to the graphical user interface of a program that uses *bio-zen* to translate these values into something that is easier understandable to the user (e.g. converting the value for the length of a cell into micrometers).

Some of the physical or temporal qualities are also subclasses of *one-dimensional-quality*. One-dimensional qualities have the special feature that they can be fully described by a single value (e.g. the temperature of a solution can be fully described by one value). This differentiates them from other qualities that need more than one value to be fully described (e.g. the position of an object in three-dimensional coordinates would need to be described with three values for the x, y and z axis).

One-dimensional-qualities are special because they can be correlated to each other through individuals of the *correlation* class.

2.1.7. The *correlation* class

Individuals of the *correlation* class can be used to state correlations between different qualities over time. These correlations can be described conceptually (e.g. *described-by* the concept 'positively correlated') and through mathematical descriptions expressed in MathML. The latter feature can be used to describe numeric models and simulations similar to the Systems Biology Markup Language (SBML).

Examples of statements that can be made using the *correlation* class: "Concentration of metabolite A is positively correlated with concentration of metabolite B", "Concentration of metabolite A is negatively correlated with the conversion rate of the enzymatic reaction B", "The rate of influx of metabolite A into compartment B is equal to 123,4 times the first derivative of the concentration of metabolite B minus the second derivative of the concentration of metabolite C" etc.

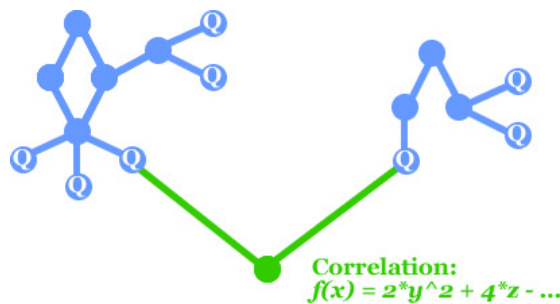


Figure 3: Two qualities that show some correlation over a certain time course can be related to each other through individuals that belong to the *correlation* class. The correlation is connected to the qualities that participate in it through the *correlates* property and its sub-properties (green edges).

2.1.8. The *Concept* classes

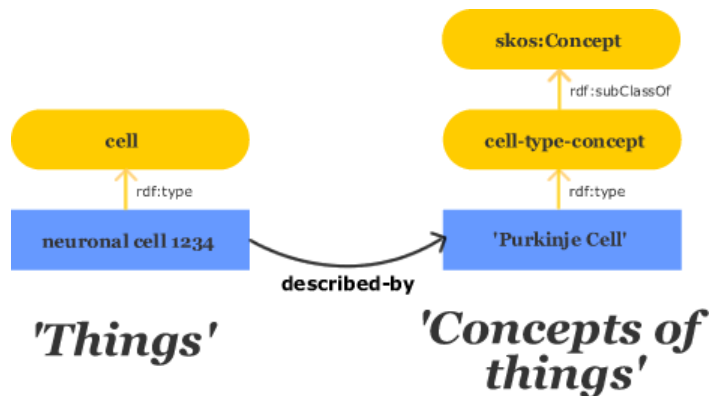


Figure 4: The two different 'worlds' in the *bio-zen* framework: the world of concrete, spatio-temporal 'things' and the world of abstract 'concepts about things'. Both worlds can only be connected through the 'described-by' property – otherwise, they are completely separated.

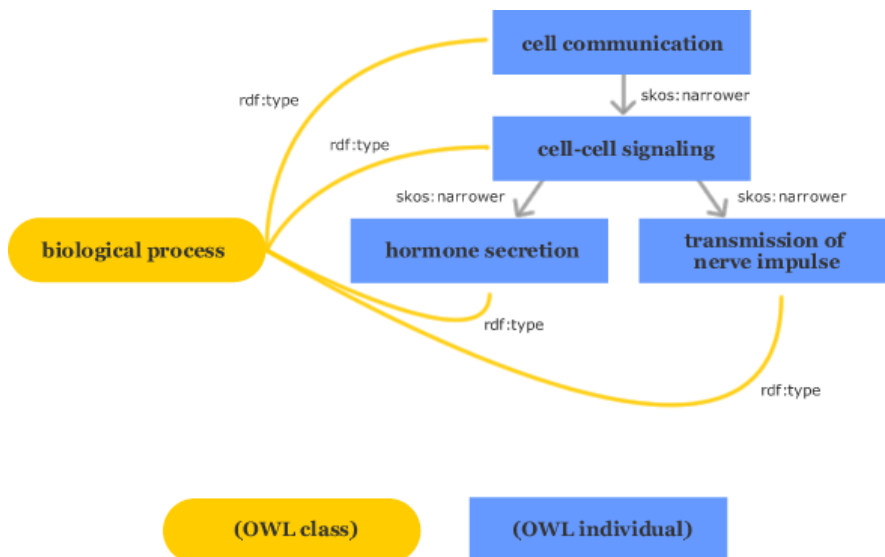


Figure 5: all of the concepts are individuals that belong to one or more concept – classes. In the case depicted here, all of the concepts have the type 'biological process'. Having subclasses of skos:Concept gives us the ability to restrict annotation of certain entities to certain concepts and certain controlled vocabularies. This feature can also be used to determine to which controlled vocabulary a certain concept belongs.

The concept – individuals can act as a kind of 'glue' between different models (Figure 1 (Figure 6)). Different things (e.g. *spatio-temporal-particulars*) can be described with the same concept, acting as a hint that they belong to the same class of things – without the need for creating complicated hierarchies of OWL classes. This technique also allows for interoperability between different (and possibly incompatible) ontologies. This is much easier and less prone to involuntary errors than making ontology alignments (with owl:subClassOf etc.).

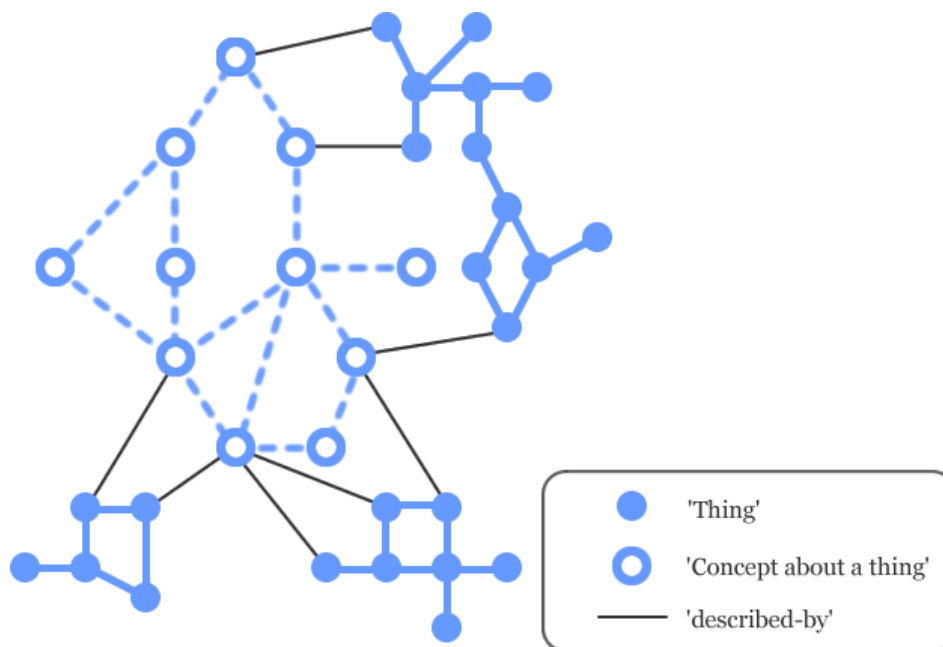


Figure 6: Concepts can act as a kind of 'glue' between different models.

2.1.9. Making references to databases and publications: xref and the Dublin Core metadata standard

[...]

3. The bio-zen ontology framework: demonstrated with simple examples

The following chapter will present simple examples of what can be done using the *bio-zen* ontology.

3.1. An interaction network

“Molecules from protein-population-1 bind to molecules from protein-population-2. Molecules from protein-population-2 bind to the binding domain of molecules from protein-population-3”. Note that the second interaction is described in more detail than the first interaction, because we know which domain on protein-population-3 other molecules bind to.

```
<protein-population-1> <rdf:type> <protein-population>
<protein-population-2> <rdf:type> <protein-population>
<protein-population-3> <rdf:type> <protein-population>
<binding-domain-of-p3> <rdf:type> <part-of-protein-population>
<p1-p2-binding-process> <rdf:type> <molecular-binding-process>
<p2-p3-domain-binding-process> <rdf:type> <molecular-binding-process>

<p1-p2-binding-process> <participant> <protein-population-1>
<p1-p2-binding-process> <participant> <protein-population-2>

<protein-population-3> <part> <binding-domain-of-p3>

<p2-p3-domain-binding-process> <participant> <protein-population-2>
<p2-p3-domain-binding-process> <participant> <binding-domain-of-p3>
```

3.2. Complex formation

“Molecules from protein-population-1 form complexes with molecules from protein-population-2.” Note that the complexes that are created form their own population. As with molecule binding, this describes both the assembly and the disassembly of complexes (which might be in some state of equilibrium).

```
<protein-population-1> <rdf:type> <protein-population>
<protein-population-2> <rdf:type> <protein-population>
<p1-p2-complex-population> <rdf:type> <molecular-complex-population>
<p1-p2-assembly-process> <rdf:type> <complex-assembly-process>

<p1-p2-assembly-process> <participant> <protein-population-2>
<p1-p2-assembly-process> <participant> <protein-population-1>
<p1-p2-assembly-process> <participant> <p1-p2-complex-population>
```

3.3. A chemical pathway

“The adrenaline-synthesis-pathway has two reactions as its parts: the conversion of DOPA to dopamine that is catalyzed by a decarboxylase, and the conversion of dopamine to adrenaline that is catalyzed by a hydroxylase”

```

    <DOPA> <rdf:type> <small-molecule-population>
    <Dopamine> <rdf:type> <small-molecule-population>

    <Adrenaline> <rdf:type> <small-molecule-population>

    <DOPA-decarboxylase> <rdf:type> <protein-population>
    <Dopamine-hydroxylase> <rdf:type> <protein-population>

    <DOPA-to-Dopamine> <rdf:type> <catalyzed-chemical-conversion-process>
    <Dopamine-to-Adrenaline> <rdf:type> <catalyzed-chemical-conversion-process>
    <Adrenaline-synthesis-pathway> <rdf:type> <chemical-pathway>

    <DOPA-to-Dopamine> <left-participant> <DOPA>
    <DOPA-to-Dopamine> <right-participant> <Dopamine>
    <DOPA-to-Dopamine> <catalyzed-by> <DOPA-decarboxylase>

    <Dopamine-to-Adrenaline> <left-participant> <Dopamine>
    <Dopamine-to-Adrenaline> <right-participant> <Adrenaline>
    <Dopamine-to-Adrenaline> <catalyzed-by> <Dopamine-hydroxylase>

    <Adrenaline-synthesis-pathway> <part> <DOPA-to-Dopamine>
    <Adrenaline-synthesis-pathway> <part> <Dopamine-to-Adrenaline>

```

Note: it is not necessarily the case that the net flux of a chemical conversion is from the 'left side' participants to the 'right side' participants of the reaction. The directionality of a reaction needs to be explicitly stated (e.g. via the *reaction-direction-and-speed* quality).

3.4. Protein structure and sequence

The following is a brief description of the structure of a serotonin receptor: its sequence, length and transmembrane domain.

```

    <serotonin-receptor-population> <rdf:type> <protein-population>
    <transmembrane-domain-population> <rdf:type> <subsequence-part-of-protein-population>
    <serotonin-receptor-sequence> <rdf:type> <protein-sequence-data>

    <serotonin-receptor-population> <sequence-length> "471"
    <serotonin-receptor-population> <described-by> <serotonin-receptor-sequence>
    <serotonin-receptor-sequence> <sequence-string> "mdilceentslssttnslmqlddtrlysn..."

    <serotonin-receptor-population> <part> <transmembrane-domain-population>
    <transmembrane-domain-population> <sequence-start-position> "91"
    <transmembrane-domain-population> <sequence-end-position> "380"

```

Note: we could also describe the sequence of the sub-domains of the proteins, but this was omitted in this example.

3.5. Tagging and annotation with concepts

You can use the *described-by* property to annotate (or 'tag') certain *spatio-temporal-particulars* with a concept. For example, a scientist could make the statement:

```
<serotonin-receptor-molecule-population-123> <described-by> <serotonin-receptor>
```

Another scientist could describe another population of serotonin receptors. In doing so, she can re-use the same concept to annotate it:

```
<serotonin-receptor-molecule-population-456> <described-by> <serotonin-receptor>
```

Someone who is interested in finding some information about serotonin receptors on the Semantic Web could run a query to find all of the entities that have been described with the concept 'serotonin-receptor'. A search would yield both of the *molecule-populations* described above.

Concepts can be ordered in hierarchies with the *narrower* and *broader* properties of SKOS. In general, the statement

```
<concept1> <narrower> <concept2>
```

Means that concept2 is in some way more specific than concept1. This does not necessarily imply a strict 'is a' relationship between the entities represented by these concepts. To explicitly state that the two concepts point to a real 'is a' relationship, we can use the *narrowerGeneric* property, e.g.:

```
<serotonin-receptor> <narrowerGeneric> <5HT2A-serotonin-receptor>
```

This points to the fact that all things that can be described as 'serotonin receptors of the 5HT2A subtype' can also be described as 'serotonin receptors' (a class – subclass relationship). The narrower / broader hierarchies of concepts can be very large.

Rule of thumb for using concepts and annotations: Express as much as possible in the world of *spatio-temporal-particulars* and not in the world of abstract *concepts*. Example:

You are describing a protein with enzymatic function that is important in the process of glial cell differentiation. You want to annotate its function with a concept of GO (e.g. the *molecular-process-concept* "GO_0010001: glial cell differentiation"). The simplest way to do this would be to just say

```
<protein-population-123> <described-by> <GO_0010001>
```

This might be very brief, but not very elegant, as we have described a population of molecules with a concept that actually refers to a *process*, not a molecule. Of course, most people will understand what is implied in this statement, but it is still preferable to be a little more precise.

To add some precision to our statement, we could make the explicit statement that protein-population-123 participates in a process. Then we could annotate this process with the concept for "glial cell differentiation". The new version of our statement would therefore be:

```
<process-123> <participant> <protein-population-123>  
<process-123> <described-by> <GO_0010001>
```

Such concise descriptions should be preferred wherever possible.

3.5.1. Gluing incompatible ontologies together with concept – tags

[...]

3.6. Correlation: describing the dynamics of biological systems

“The speed of the reaction protein-1 => protein-2 is positively correlated with the concentration of lactose”

First, the description of the reaction and the three molecule populations protein-1, protein-2 and lactose:

```
<protein-population-1> <rdf:type> <protein-population>
<protein-population-2> <rdf:type> <protein-population>
<lactose-population> <rdf:type> <small-molecule-population>
<p1-p2-conversion> <rdf:type> <chemical-conversion-process>

<p1-p2-conversion> <left-participant> <protein-population-1>
<p1-p2-conversion> <right-participant> <protein-population-2>
```

Now we can add the appropriate qualities (reaction speed, concentration):

```
<speed-of-p1-p2-conversion> <rdf:type> <reaction-direction-and-speed>
<p1-p2-conversion> <quality> <speed-of-p1-p2-conversion>

<lactose-concentration> <rdf:type> <concentration>
<lactose-population> <quality> <lactose-concentration>
```

Finally, we can make the statement that there is a correlation between the two qualities and annotate with the concept ‘positive correlation’.

```
<correlation123> <rdf:type> <correlation>
<correlation123> <correlates-assign> <speed-of-p1-p2-conversion>
<correlation123> <correlates-A> <lactose-concentration>
<correlation123> <described-by> <positive-correlation_>
```

Note: <positive-correlation_> is an individual that belongs to the *correlation-concept* class and is part of the core *bio-zen* ontology.

correlates-A, *correlates-B* etc. and *correlates-assign* are sub-properties of the *correlates* property. They postfixes (assign, A, B etc.) are used to distinguish different qualities when they are used in mathematical formulas.

Besides describing the correlation with concepts, we can also describe it with mathematical formulas expressed in MathML. A formula can be attached to a correlation through the *mathML* property of the correlation:

```
<correlation123> <mathML> "<mrow> ...some formula in MathML format... </mrow>"
```

The use of correlations should enable the user to describe biological models that can be used for numerical simulation, similar to the Systems Biology Markup Language (SBML). The first stable release of *bio-zen* should be able to represent most models that are now represented in SBML. A *bio-zen* – SBML converter should also make it possible to use simulation and modelling software that is designed for SBML.

Note: the use of MathML in *bio-zen* is still not fully standardized. If you have expertise in the use of MathML, feel free to participate in the development.

3.7. Causation

```
<short-heat-shock> <rdf:type> <event>
<heat-shock-response-pathway> <rdf:type> <chemical-pathway>

<short-heat-shock> <causes> <heat-shock-response-pathway>
```

3.8. From correlation to causation and back

The transition between observed correlations and the postulation of causative processes that lie behind these correlations are central to scientific research.

First observation: “The conversion of molecules from protein-population-1 is negatively correlated with the concentration of protein-population-ABC” (for a description of correlations, see chapter 3.6). This can be expressed like we have seen in previous chapters:

```
<protein-population-1> <rdf:type> <protein-population>
<protein-population-ABC> <rdf:type> <protein-population>

<protein-1-concentration> <rdf:type> <concentration>
<protein-population-1> <quality> <protein-1-concentration>

<protein-ABC-concentration> <rdf:type> <concentration>
<protein-population-ABC> <quality> <protein-ABC-concentration>

<correlation123> <rdf:type> <correlation>
<correlation123> <correlates-assign> <protein-ABC-concentration>
<correlation123> <correlates-A> <protein-1-concentration>
<correlation123> <described-by> <negative-correlation_>
```

Some months later, another researcher finds the *cause* for this correlation: molecules from protein-population-ABC have enzymatic activity and catalyze the conversion of protein-1 to protein-2, thereby depleting the population of protein-1 molecules. This is the reason why we made the observation that the concentration of protein-ABC was negatively correlated with the concentration of protein-1! The correlation was *caused* by a *chemical-conversion-process*.

The other researcher can add the following triples to describe his experimental findings:

```
<protein-population-2> <rdf:type> <protein-population>
<p1-p2-conversion> <rdf:type> <chemical-conversion-process>

<p1-p2-conversion> <left-participant> <protein-population-1>
<p1-p2-conversion> <right-participant> <protein-population-2>
<p1-p2-conversion> <catalyzed-by> <protein-population-ABC>

<p1-p2-conversion> <causes> <correlation123>
```

3.9. The flow of time: representing temporal relations

Temporal relations (e.g. *precedes*, *temporally-includes*, *started-by* etc.) can be used intuitively to relate the timely order of perdurants like processes and events. Examples:

```
<mitosis> <started-by> <prophase>  
<prophase> <meets> <metaphase>  
<metaphase> <precedes> <interphase>
```

3.10. Evolution of description models

The descriptions about biological reality we are making on the Semantic Web are not static. Modern science produces a steady flow of new information, data, and newer and refined models about the world that are the result of the cooperation of the international scientific community.

Semantic Web technologies allow for cooperation with unequalled flexibility and ease. Through the use of globally unique URIs, every structure of a model can be enhanced and refined with additional descriptions by everyone on the Semantic Web (Figure 7).

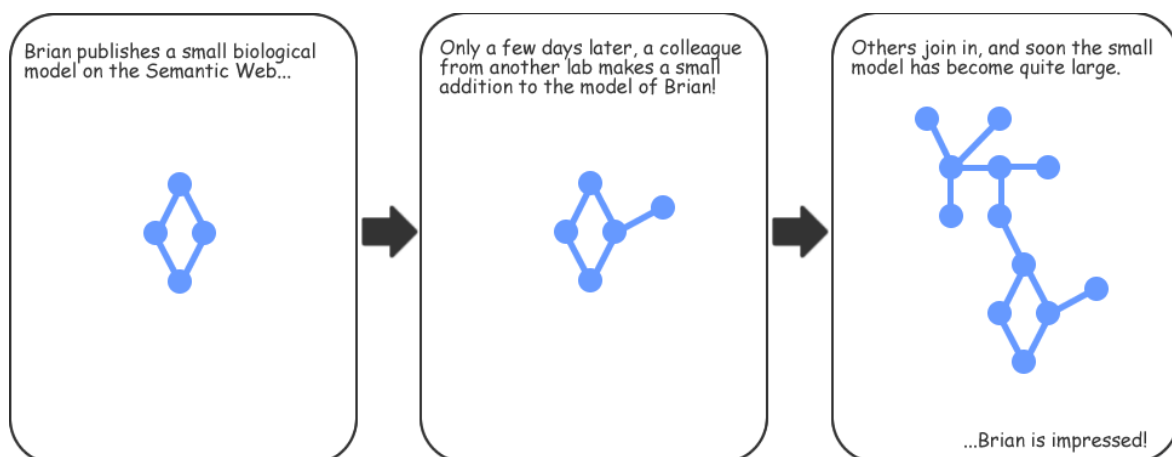


Figure 7: Multiple users in different locations can work collaboratively on a single graph. New information is added to existing information, making the graph grow.

While adding new details to existing descriptions might be useful in many situations, there are also situations where we want to leave the graph in its existing state and create our own graph based on the existing graph. There might be many reasons for doing so: maybe we think that the original description is fine in itself, and that not everyone will agree with the refinements we are planning; maybe we feel that we do not have the authority to add things to the graphs of others; maybe we are in fact describing something different (e.g. a metabolic pathway in a certain disease state, while the original graph is describing the pathway in a healthy organism); or maybe we simply want to make statements that are incompatible with the original graph (e.g. we want to remove some statements that are made in the original graph).

This can be done through a process that is called '*graph cloning*' in bio-zen. If graph B is a clone of graph A, then graph B has the same structure as graph A and the same values for its datatype properties as graph A. The URIs of the individuals of graph B, however, are all (or mostly) different from the URIs of the individuals of graph A – during the process of cloning, the nodes have been given a new URI. In other words, the things described by a

graph and its clone are similar but not identical. In most cases, all of the *spatio-temporal-particulars* of a cloned graph will have new URIs, while the URIs of the *abstracts* can be left unchanged in most cases.

When we have cloned a graph, we can make all the changes to it that we want, without thinking about the impact on the original graph – both graphs are completely separated.

However, the relationship between a graph and its clone might seem interesting to many people. For instance, if you have placed a graph on the Semantic Web, you might want to know which graphs have been derived from your work.

Relating a clone to the original graph it was derived from can be done with the *prototype* property. The resulting phylogenies of models can be automatically queried to get an overview of the progression of graphs (Figure 8).

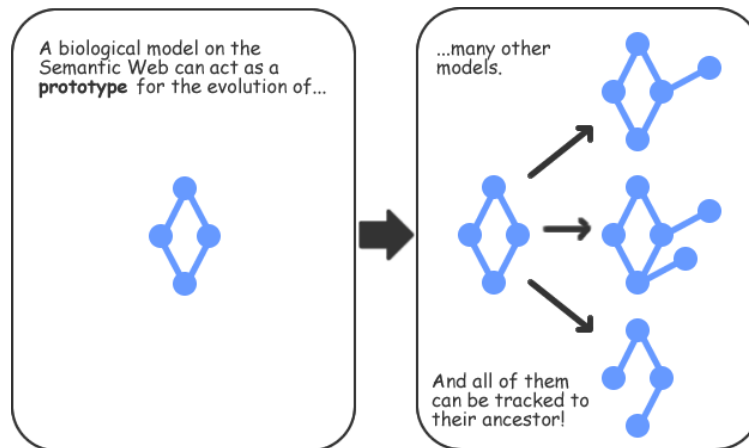


Figure 8: Graphs and parts of graphs can be cloned, yielding new graphs with similar structure but different URIs for their nodes. Clones can be connected to their ‘ancestor’ with the *prototype* property of the DOLCE/*bio-zen* ontology. The new graphs can be modified without influencing the ancestor. They can contain alternative or refined descriptions, or they can describe different things altogether (example: one graph describing a metabolic pathway in a normal human brain can act as a prototype for a graph describing the metabolic pathway in an Alzheimer’s patient). Of course, all of the derived graphs can again act as prototypes for further graphs, producing a phylogeny of graphs. The ‘evolutionary’ history of the graphs can be tracked through the *prototype* property.

3.11. Bridging the gap between different system scales: exemplary parts

We can use the *exemplary-part* property to refer to a part of a thing that is in fact one of many (‘uncountable’) similar parts.

Example:

```
<liver-tissue> <exemplary-part> <typical-liver-cell>
```

`typical-liver-cell` here stands as an example for many other, similar liver cells that are also part of the liver-tissue, but which are not explicitly described.

3.12. Rules and reasoning

[...]

3.13. Fuzzy facts: the metrics of realness and interestingness

A property called 'realness' can be used to represent a measure of uncertainty to every *spatio-temporal-particular* in the ontology. The value of this property should be a floating point number between 0 and 1 (inclusive). This is a metric to state how valid the assertions made by an entity are, i.e. how probable it is that this entity is really found in nature.

If a user applies a realness-value of 1 to an entity this essentially means that he or she is sure that the entity exists in nature. Lower values mean that he or she is less certain. A value of 0 means that the entity is not a valid representation of reality at all. However, this should NOT be understood as a NEGATION of any assertions we can associate with the entity. It just means that the entity described in OWL is useless for our understanding of reality.

Another property called 'interestingness'. The term *interestingness* is borrowed from [Flickr.com](http://www.flickr.com).

Scores for interestingness and realness can be meaningfully combined. For instance, an interesting but unproven new theory might be assigned a high interestingness score but a low realness score.

3.14. Concept extension packages: new concepts and vocabularies

There are several extension packages available for download at

<http://neuroscientific.net/index.php?id=download>

These packages contain various concept hierarchies and vocabularies that can be used for annotation. Currently, the following packages are available:

Gene Ontology, see <http://www.geneontology.org/>

Medical Subject Headings (MeSH), see <http://www.nlm.nih.gov/mesh/>

Celltype ontology, see <http://obo.sourceforge.net/cgi-bin/detail.cgi?cell>

Sequence ontology, see <http://song.sourceforge.net/>

INOH Molecule role ontology, see <http://www.inoh.org/download.html#OntologyData>

Each one of these packages imports the main *bio-zen* ontology via the internet. More packages will be added.

3.15. Using generic classes to model things that are not (yet) covered in bio-zen

Example: At the time, there is no dedicated class for a "gene" in bio-zen, although this will surely change in the near future. Luckily, the flexibility of OWL and the foundational ontology allow us to describe a gene with more generic classes despite the lack of a class for genes.

The class that is closest to "gene" is the bio-zen class "subsequence-part-of-dna-population". With this we can model a gene as a part of a DNA strand that can be described by a sequence of nucleotides. We can also create a concept of 'gene', which we can use to annotate our *subsequence-part-of-dna-population* so everyone recognizes that we are describing a gene.

```
<gene001> <rdf:type> <subsequence-part-of-dna-population>
<gene001> <described-by> <gene-concept>
```

We could also define narrower concepts of the gene-concept to distinguish different kinds of genes.

Additions like these only require the end-user to define new individuals, but no new classes. Once some ontology developer has defined a dedicated class for 'Gene', all of the individuals that have been previously described using *subsequence-part-of-dna-population* and the concept annotation can be automatically classified by OWL under the new class. This can be done by automated OWL reasoning over the *described-by-OR-narrower* property.

4. Design principles

>>**Consistency and interoperability.** The ontology is built upon existing foundational ontologies ([DOLCE](#) and [SKOS](#)). This means that it is rooted on a sound ontological framework, easing the interoperability with ontologies from other domains. Dublin Core and other existing metadata standards are used.

>>**Flexibility and simplicity.** Over-specialisation is avoided where possible. Generic properties like 'part of', 'constituent of', 'broader concept than', 'caused by' etc. are the backbone of the whole ontology. The class hierarchy is kept simple and lean.

>>**'Batteries attached'.** The framework includes a rich set of controlled vocabularies (in the form of SKOS concepts) with URIs that anyone can use right away.

>>The ontology integrates two different approaches of information representation in a common framework: 'realist' ontological descriptions and 'conceptualist' taxonomies and concept hierarchies.

- Making a clear distinction between both approaches reduces the susceptibility to inconsistencies.
- Unifying both approaches in one common framework makes it possible to combine the specific advantages of each approach in the best way possible. The consistency of the realist approach is complemented with the flexibility of the conceptualist approach.

>>Users of the ontology only need to make OWL individuals to represent information. The definition of new classes is not necessary, which helps to avoid many of the associated problems.

>>The ontology is focused on the description of spatio-temporal particulars (concrete biological things and models). This focus limits the problems that arise when users of the ontology can define universals.

>>The ontology allows the seamless integration of mathematical models (similar to and compatible with SBML 2) into qualitative information.

>>Molecular interactions are modelled as stochastic processes involving populations of molecules, not as singular events that involve singular molecules. This approach is much closer to biological reality in most occasions, and avoids some grave consistency problems associated with the other approach.

>>Strict use of SI units for physical measurements.

>>The ontology includes simple means to 'fuzzify' information (i.e. basic support for fuzzy logic - like constructs).

5. bio-sparqlets

The **bio-sparqlets project** will use the *lingua franca* of server-side programming on the web, PHP, the newly standardized SPARQL query language and the *bio-zen* ontology to develop an infrastructure of lightweight scripts to build an infrastructure for the biological Semantic Web.

The bio-sparqlets will make use of the *RAP RDF API for PHP*¹⁰.

[...]

6. bio-zen and the Life Science Identifiers (LSID)

[...]

7. Relations between bio-zen and other ontologies

7.1. bio-zen classes and BioPAX classes

Some of the classes in the core *bio-zen* ontology have been created by mapping classes from the BioPAX ontology to DOLCE. Here are some examples:

bio-zen class	BioPAX class
~abstract	~utilityClass
particular	entity
molecular-process	physicalInteraction
~correlation	~modulation
molecular-transport-with-chemical-conversion-process	transportWithBiochemicalReaction
chemical-conversion-process	biochemicalReaction
metabolic-pathway	pathway
complex-assembly-process	complexAssembly
molecular-transport-process	transport
physical-object	physicalEntity
molecular-complex-population	complex
dna-population, protein-population etc.	dna, protein etc.

¹⁰ See <http://www.wiwiss.fu-berlin.de/suhl/bizer/rdfapi/>

7.2. Properties in *bio-zen* and properties from the *OBO relations ontology*

The *OBO relations ontology* defines some basic properties that are the base of the planned *OBO foundry*. Some of these properties have an equivalent in the *DOLCE ontology* on which *bio-zen* is based.

<i>bio-zen</i> property	OBO relations ontology property
	Has_agent
dol:participant	Has_participant
dol:part	part_of (<i>inverse</i>)
	integral_part_of
dol:proper-part	Proper_part_of (<i>inverse</i>)
	Improper_part_of
	Located_in
	Contained_in
	Adjacent_to
	Transformation_of
	Derives_from
	Preceded_by