Lecture Notes

Big Data in Medical Informatics

Week 4:
Ontologies - Exercises

Oya Beyan, PhD.
Prof. Dr. Stefan Decker

RWTH AACHEN UNIVERSITY

OWL: Restrictions

- Use rdfs:domain and rdfs:range to classify data

- A vegetarian diet:
  :Person a owl:Class.
  :Food a owl:Class.
  :eats rdfs:domain :Person.
  :eats rdfs:range :Food.

- instance data
  :Maverick :eats :Steak.

What can you conclude ?

OWL: Restrictions

- Use rdfs:domain and rdfs:range to classify data

- A vegetarian diet:
  :Person a owl:Class.
  :Food a owl:Class.
  :eats rdfs:domain :Person.
  :eats rdfs:range :Food.

- instance data
  :Maverick :eats :Steak.

What can you conclude ?

:Maverick a:Person.
:Steak a:Food.

OWL: Restrictions

- define a variety of diets
  - we have a particular kind of person, called a Vegetarian,
  - and the kind of food that a Vegetarian eats, which we will call simply VegetarianFood

- How to represent this ?

OWL: Restrictions

- define a variety of diets
  - we have a particular kind of person, called a Vegetarian,
  - and the kind of food that a Vegetarian eats, which we will call simply VegetarianFood

- represent as subclasses of Person and Food

```
:Vegetarian a owl:Class;
    rdfs:subClassOf :Person.
:VegetarianFood a owl:Class;
    rdfs:subClassOf :Food.
```

OWL: Restrictions

- Given Instance

  :Jen a:Vegetarian;

  :eats :Marzipan.

- How can we want to infer that?

  :Marzipan a:VegetarianFood.

- define the set of things that only eat VegetarianFood using a restriction owl:allValuesFrom
- assert that any Vegetarian satisfies this condition using rdfs:subClassOf.

```
:Vegetarian rdfs:subClassOf
   [a owl:Restriction;
      owl:onProperty :eats;
      owl:allValuesFrom :VegetarianFood].
```

OWL: Restrictions

- How it works:

```
:Vegetarian rdfs:subClassOf
   [a owl:Restriction;
      owl:onProperty :eats;
      owl:allValuesFrom :VegetarianFood].
```

- Since
  :Jen a:Vegetarian.
- we can conclude that
  :Jen a [a owl:Restriction;
  owl:onProperty :eats;
  owl:allValuesFrom :VegetarianFood].
- Combined with the fact that
  :Jen :eats :Marzipan.
- we can conclude that
  :Marzipan a:VegetarianFood.

- *RIMBAUD: I saw a James Dean movie last night.*
- *ROCKY: Was it* Giant?
- *RIMBAUD: No.*
- *ROCKY: Was it* East of Eden?
- *RIMBAUD: No.*
- *ROCKY: James Dean only made three movies; it must have been* Rebel Without a Cause.
- *RIMBAUD: Yes, it was.*

- This sort of inference relies on the fact that James Dean made only three movies.

- How to represent a particular class (James Dean movies), all of its members are known ?

- *Answer: Enumerating sets with owl:one of*

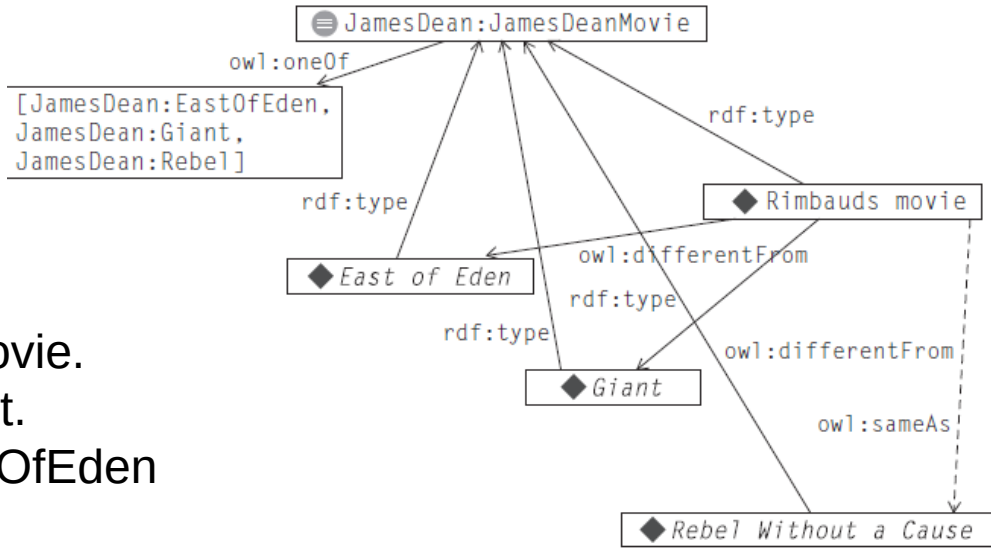
:JamesDeanMovie a owl:Class;
owl:oneOf (:Giant :EastOfEden :Rebel).

- When one is in a position to enumerate the members of a class, a number of inferences can follow.

:Giant rdf:type :JamesDeanMovie.
:EastOfEden rdf:type :JamesDeanMovie.
:Rebel rdf:type :JamesDeanMovie.

:RimbaudsMovie rdf:type :JamesDeanMovie.
:RimbaudsMovie owl:differentFrom :Giant.
:RimbaudsMovie owl:differentFrom :EastOfEden

then we can infer :
:RimbaudsMovie owl:sameAs :Rebel.



- the class JamesDeanMovie was defined using owl:oneOf to indicate that these are the only James Dean movies in existence.

- Question :
- Now add an additional statement saying that these three movies are distinct.

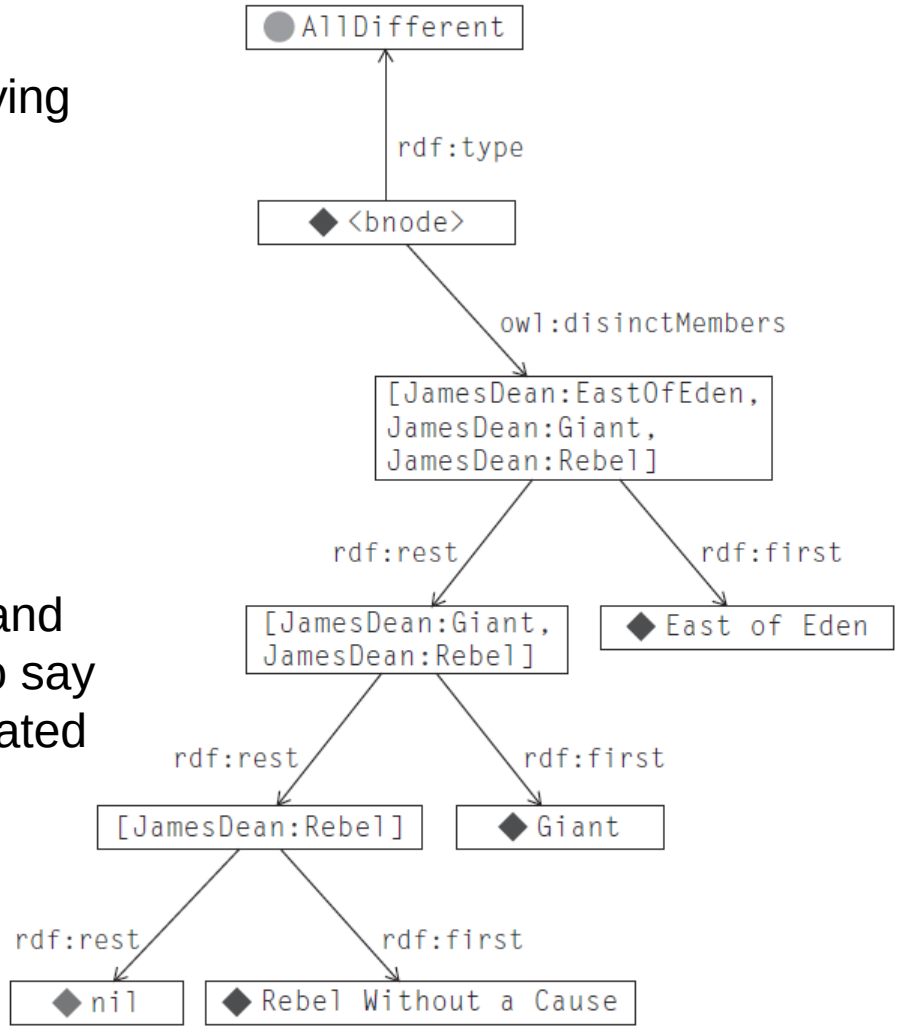
Question :

- Now add an additional statement saying that these three movies are distinct.

- Answer:

[a owl:AllDifferent;
owl:distinctMembers (:EastOfEden
:Giant
:Rebel)].

- It is quite common to use owl:oneOf and owl:AllDifferent together in this way to say that a class is made up of an enumerated list of distinct elements.



- *RIMBAUD: Do you own any James Dean movies?*
- *ROCKY: They are the only ones I own.*
- *RIMBAUD: Then I guess you don't own very many movies! No more than three.*

- Model Rocky's statement that he owns only James Dean movies.

- *RIMBAUD: Do you own any James Dean movies?*
- *ROCKY: They are the only ones I own.*
- *RIMBAUD: Then I guess you don't own very many movies! No more than three.*

- Model Rocky's statement that he owns only James Dean movies.
- We will need a property called ownsMovie to indicate that someone owns a movie:
  :ownsMovie a owl:ObjectProperty

say that Rocky owns only James Dean movies by using the owl:allValuesFrom restriction

```
:JamesDeanExclusive owl:equivalentClass
[a owl:Restriction;
owl:onProperty :ownsMovie;
owl:allValuesFrom :JamesDeanMovie].
:Rocky a:JamesDeanExclusive.
```

- Model Rimbaud's conclusion : "*you don't own very many movies*"

- Model Rimbaud's conclusion : "*you don't own very many movies*"

- We define the class of things that don't own many movies (where by "not many," we mean at most three) as follows:

- :FewMovieOwner owl:equivalentClass
[a owl:Restriction;
owl:onProperty:ownsMovie;
owl:maxCardinality 3].

- Now Rimbaud's conclusion can be formulated as a triple:
:Rocky a:FewMovieOwner.

- Represent the following statement with OWL restriction
- “a baseball team has exactly nine (distinct) players”

- Represent the following statement with OWL restriction
- “a baseball team has exactly nine (distinct) players”

[a owl:Restriction;
owl:onProperty :hasPlayer;
owl:cardinality 9]

.

- Every Person is either Male or Female.

- Every Person is either Male or Female.

- ```
  <owl:Class rdf:ID="Person">
     <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Woman" />
        <owl:Class rdf:about="#Man" />
     </owl:unionOf>
  </owl:Class>
  ```
  ```

- A Guitarist is a subclass of the set of entities which play at least one instrument that is a Guitar.

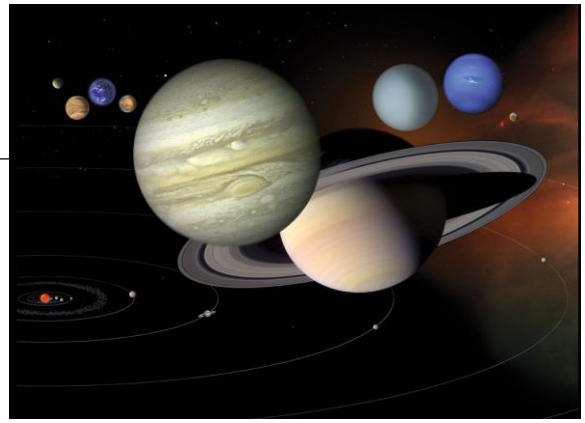
- A Guitarist is a subclass of the set of entities which play at least one instrument that is a Guitar.
-
```
<owl:Class rdf:ID=”Guitarist”>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource=”#playsInstrument”/>
      <owl:hasValue rdf:resource=”#Guitar”/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

- “All planets orbiting the sun”

- “All planets orbiting the sun” is actually the intersection of all things that orbit the sun (hasValue restriction) and all planets.

```
:SolarPlanet a owl:Class;
      owl:intersectionOf
        ( :Planet
            [a owl:Restriction;
            owl:onProperty :orbits;
            owl:hasValue :TheSun
        ]
      ).
```

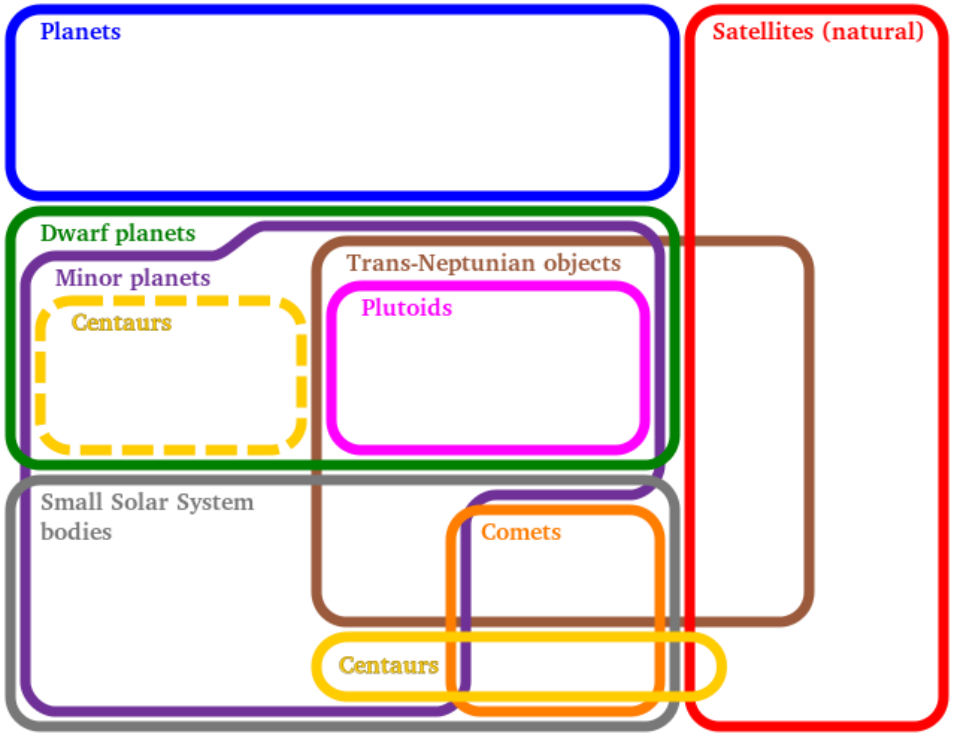
Bodies in the Solar System



- Solar System objects more massive than 10^{21} kilograms (one yottagram [Yg]) are known or expected to be approximately spherical.

- This list contains the Sun, the planets, dwarf planets, many of the larger small Solar System bodies (which includes the asteroids), all named natural satellites, and a number of smaller objects of historical or scientific interest, such as comets and near-Earth objects.

types of Bodies in the Solar System.



RWTH Informatik 5 | Ahornstr. 55 D-52056 Aachen Germany |
Tel +49/241/8021501 | Fax +49/241/8022321 | http://dbis.rwth-aachen.de/cms

The semantics of rdfs:subClassOf and owl:equivalentClass

- X rdfs:subClassOf Y.
  - simple IF/THEN relation;
  - if something is a member of X, then it is also a member of Y.
- X owl:equivalentClass Y.
  - IF and only IF relation : that is two IF/THEN relations, one going each way; if something is a member of X, then it is also a member of Y, and vice versa.
- Example: two classes:
  - one is a named class SolarBody, (class A)
  - unnamed class: defined by a restriction onProperty orbits that it hasValue TheSun, (class B)

The semantics of rdfs:subClassOf and owl:equivalentClass

- Example: two classes:
  - one is a named class SolarBody, (class A)
  - unnamed class: defined by a restriction onProperty orbits that it hasValue TheSun, (class B)

Case 1:

- all solar bodies orbit the sun: means if something is a solar body, then it orbits the sun.
  A rdfs:subClassOf B.

- “orbiting the sun is a necessary condition for SolarBody.”
- “orbiting the sun is a partial definition for the class SolarBody.”

The semantics of rdfs:subClassOf and owl:equivalentClass

- Example: two classes:
  - one is a named class SolarBody, (class A)
  - unnamed class: defined by a restriction onProperty orbits that it hasValue TheSun, (class B)

Case 2:

- solar bodies are the same as the set of things that orbit the sun
  A owl:equivalentClass B.

- If something orbits the sun, then it is a SolarBody, and if it is a SolarBody, then it orbits the sun.

- “orbiting the sun is a necessary and sufficient condition for SolarBody.”
- “orbiting the sun is a complete definition for the class SolarBody.”

OWL: Restrictions

Fundamental concepts

- owl:Restriction—The building block in OWL that describes classes by restricting the values allowed for certain properties.
- owl:hasValue—A type of restriction that refers to a single value for a property.
- owl:someValuesFrom—A type of restriction that refers to a set from which some value for a property must come.
- owl:allValuesFrom—A type of restriction that refers to a set from which all values for a property must come.
- owl:onProperty—A link from a restriction to the property it restricts.

Example Ontologies

- Toy ontology:

http://mowl-power.cs.man.ac.uk/2009/07/sssw/people.owl

Gene Ontology

- represents a collaborative effort to unify descriptions of gene products of all species
- achieved through incorporation of many different plant/animal/microbial genomes databases by the Gene Ontology Consortium.
- http://purl.obolibrary.org/obo/go.obo

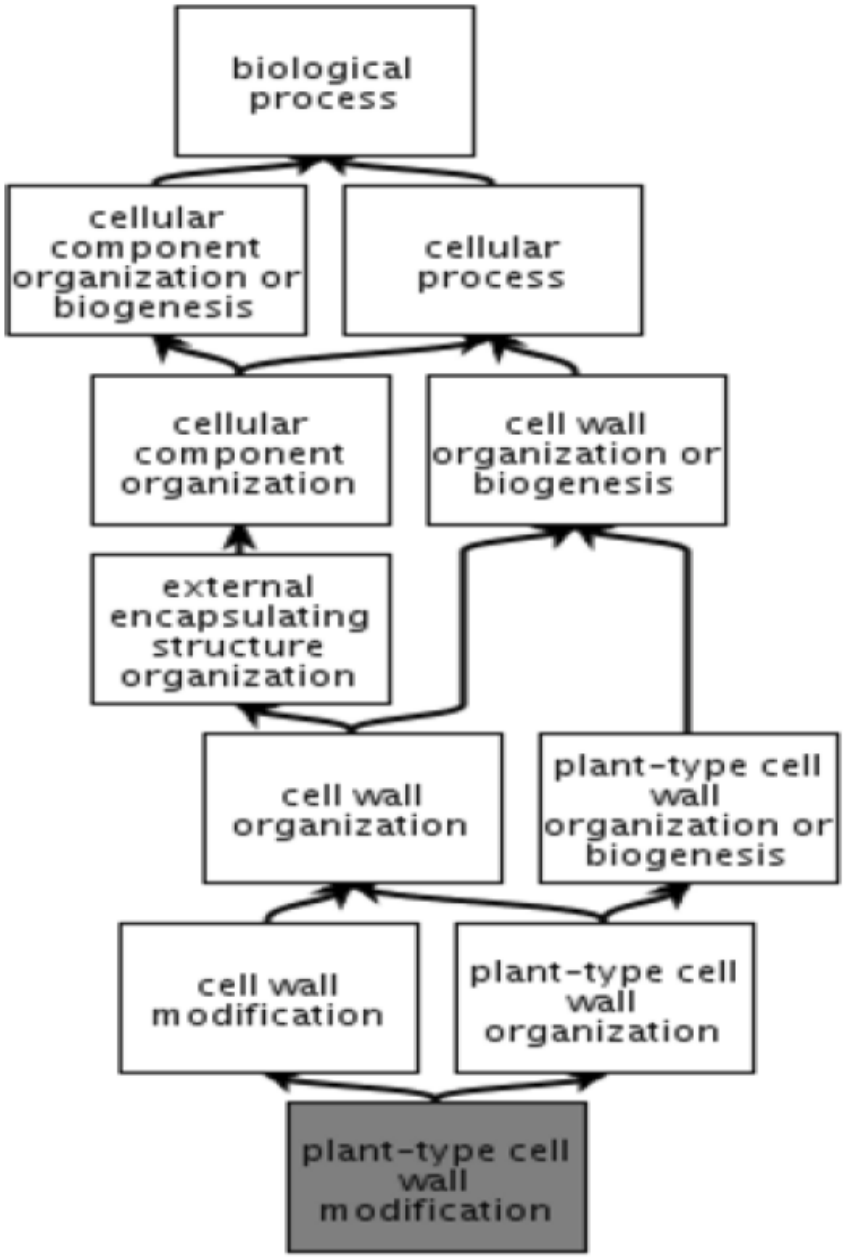
```
[Term]
id: GO:0019898
name: extrinsic component of membrane
namespace: cellular_component
alt_id: GO:0030396
def: "The component of a membrane consisting of gene products and
protein complexes that are loosely bound to one of its surfaces,
but not integrated into the hydrophobic region."
subset: gosubset_prok
synonym: "extrinsic to membrane" EXACT []
synonym: "peripheral membrane protein" EXACT []
xref: Wikipedia:Peripheral_membrane_protein
is_a: GO:0044425 ! membrane part
```

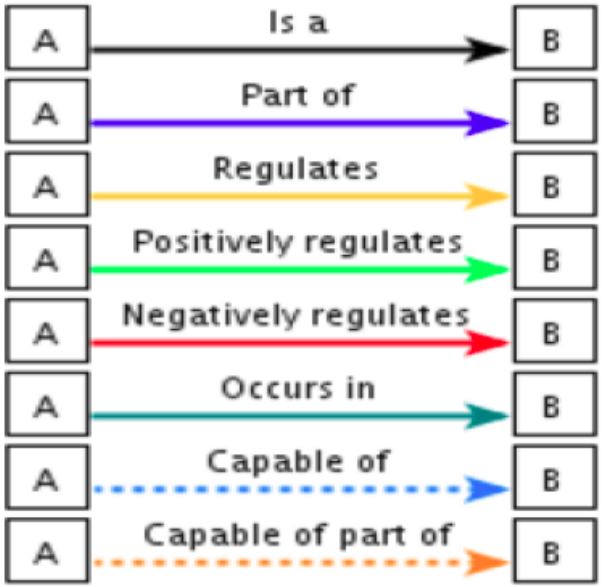
Gene Ontology

- The Gene Ontology (GO) is a resource for functional annotation of gene products.
  over 27 000 terms

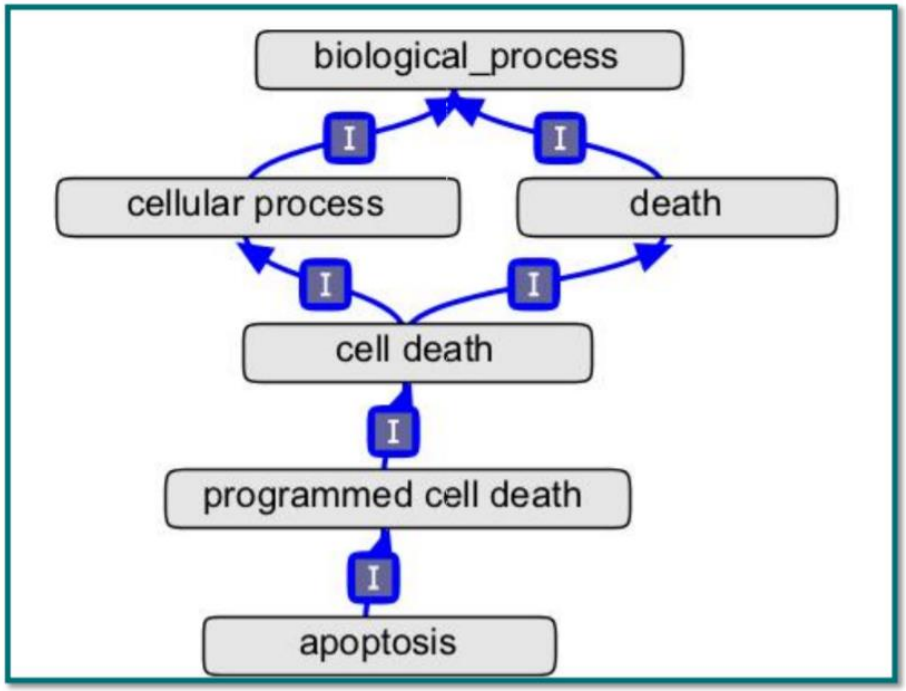
http://geneontology.org/

- Cellular Component: describe a component of a cell that is part of a larger object, such as an anatomical structure (e.g. rough endoplasmic reticulum or nucleus) or a gene product group (e.g. ribosome, proteasome or a protein dimer).
- Biological Process: describes a series of events accomplished by one or more organized assemblies of molecular functions. Examples of broad biological process terms are "cellular physiological process" or "signal transduction". Examples of more specific terms are "pyrimidine metabolic process" or "alpha-glucoside transport".
- Molecular Function: describes activities that occur at the molecular level, such as "catalytic activity" or "binding activity".

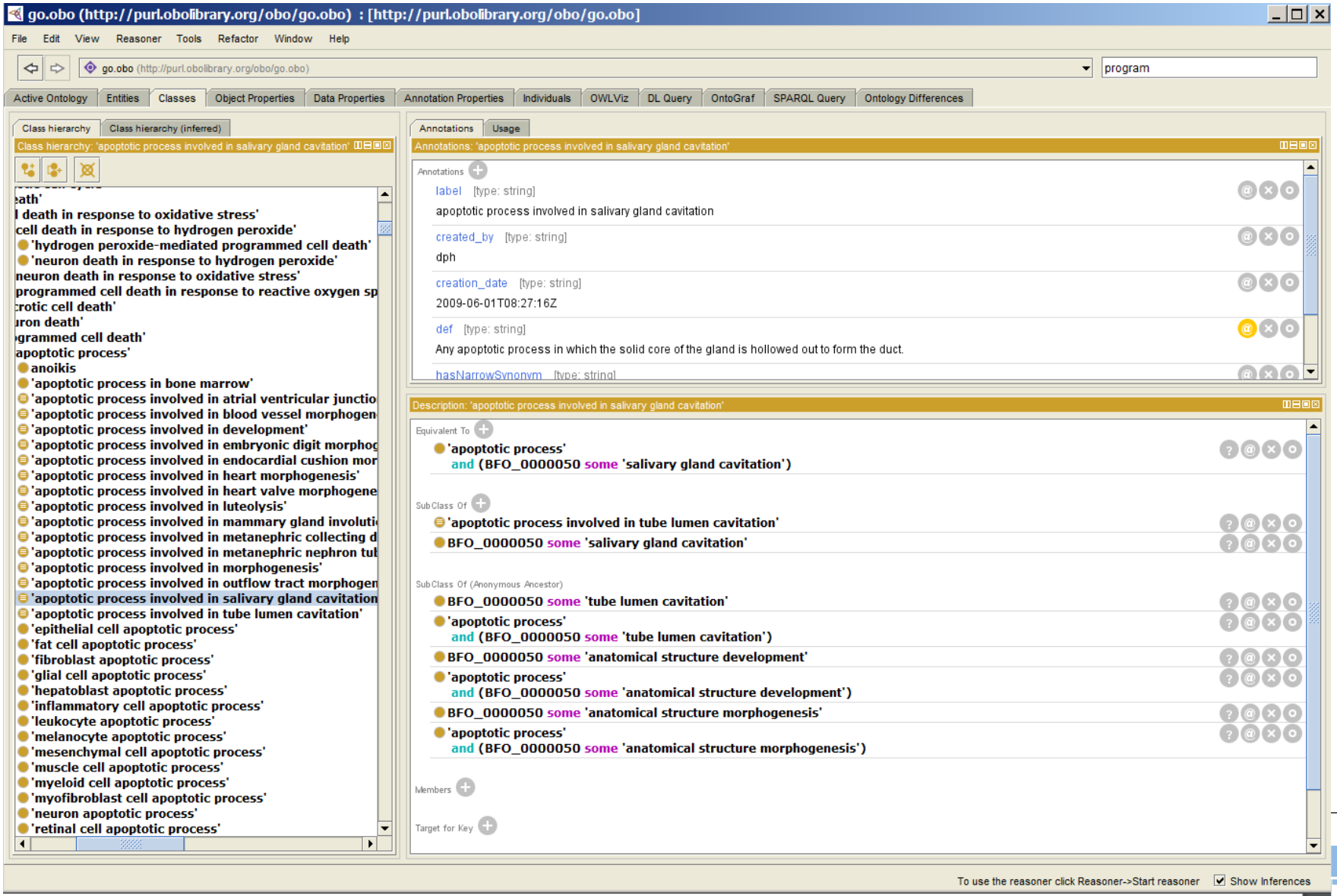

biological process
cellular component organization or biogenesis
cellular process
cellular component organization
cell wall organization or biogenesis
external encapsulating structure organization
cell wall organization
plant-type cell wall organization or biogenesis
cell wall modification
plant-type cell wall organization
plant-type cell wall modification


A
Is a
B
A
Part of
B
A
Regulates
B
A
Positively regulates
B
A
Negatively regulates
B
A
Occurs in
B
A
Capable of
B
A
Capable of part of
B

- GO:0006915 apoptosis
- Definition: : A form of programmed cell death that begins when a cell receives internal or external signals that trigger the activity of proteolytic caspases, proceeds through a series of characteristic stages typically including rounding-up of the cell, retraction of pseudopodes, reduction of cellular volume (pyknosis), chromatin condensation, nuclear fragmentation (karyorrhexis), and plasma membrane blebbing (but maintenance of its integrity until the final stages of the process), and ends with the death of the cell.

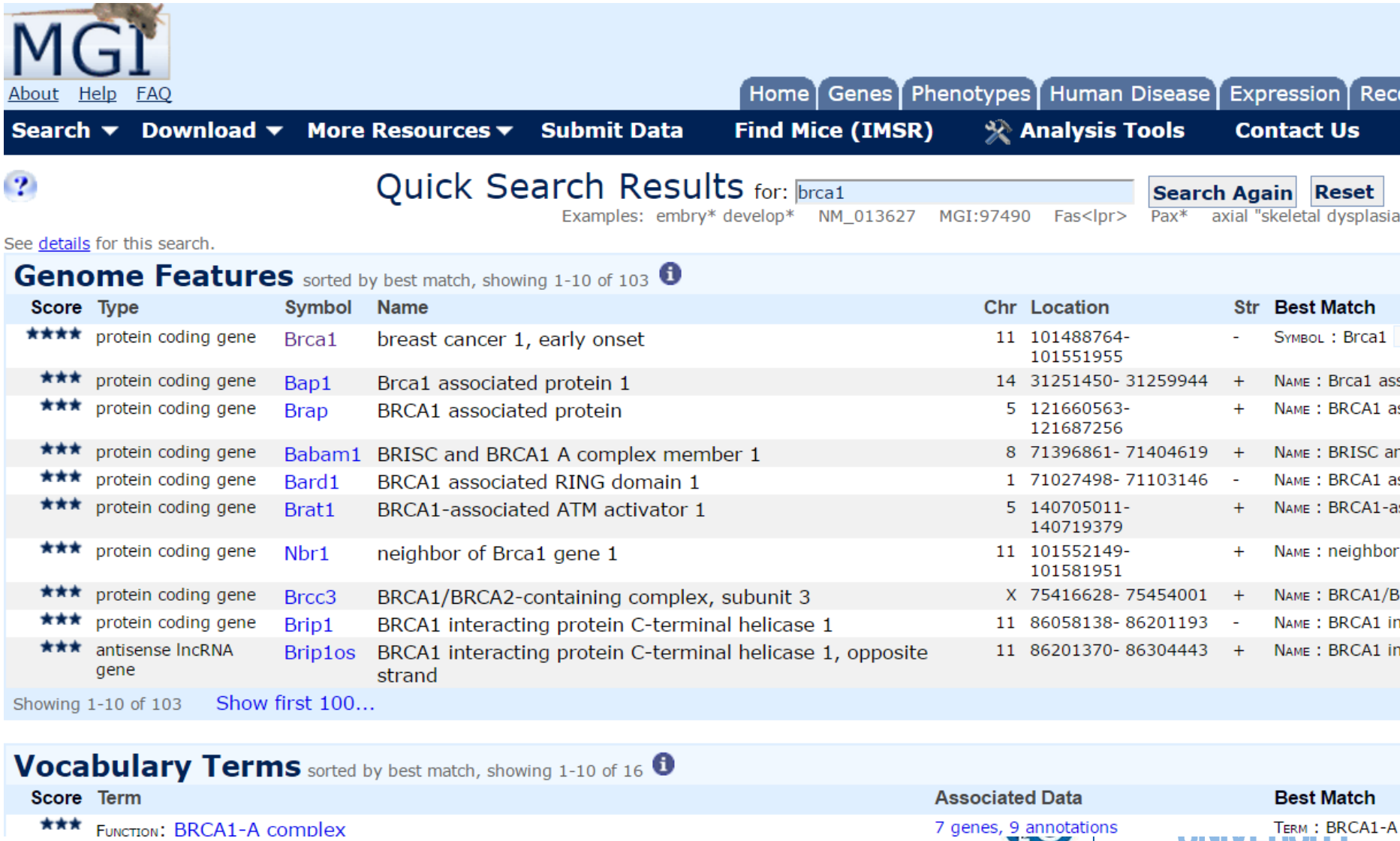


RWTH Informatik 5 | Ahornstr. 55 D-52056 Aachen Germany |
Tel +49/241/8021501 | Fax +49/241/8022321 | http://dbis.rwth-aachen.de/cms


go.obo (http://purl.obolibrary.org/obo/go.obo) : [http://purl.obolibrary.org/obo/go.obo]
File Edit View Reasoner Tools Refactor Window Help
go.obo (http://purl.obolibrary.org/obo/go.obo)
program
Active Ontology Entities Classes Object Properties Data Properties Annotation Properties Individuals OWLViz DL Query OntoGraf SPARQL Query Ontology Differences
Class hierarchy Class hierarchy (inferred)
Class hierarchy: 'apoptotic process involved in salivary gland cavitation'
'hydrogen peroxide-mediated programmed cell death'
'neuron death in response to hydrogen peroxide'
'anoikis'
'apoptotic process in bone marrow'
'apoptotic process involved in atrial ventricular junctio
'apoptotic process involved in blood vessel morphogen
'apoptotic process involved in development'
'apoptotic process involved in embryonic digit morpho
'apoptotic process involved in endocardial cushion mor
'apoptotic process involved in heart morphogenesis'
'apoptotic process involved in heart valve morphogene
'apoptotic process involved in luteolysis'
'apoptotic process involved in mammary gland involuti
'apoptotic process involved in metanephric collecting d
'apoptotic process involved in metanephric nephron tub
'apoptotic process involved in morphogenesis'
'apoptotic process involved in outflow tract morphogen
'apoptotic process involved in salivary gland cavitation
'apoptotic process involved in tube lumen cavitation'
'epithelial cell apoptotic process'
'fat cell apoptotic process'
'fibroblast apoptotic process'
'glial cell apoptotic process'
'hepatoblast apoptotic process'
'inflammatory cell apoptotic process'
'leukocyte apoptotic process'
'melanocyte apoptotic process'
'mesenchymal cell apoptotic process'
'muscle cell apoptotic process'
'myeloid cell apoptotic process'
'myofibroblast cell apoptotic process'
'neuron apoptotic process'
'retinal cell apoptotic process'
Annotations Usage
Annotations: 'apoptotic process involved in salivary gland cavitation'
Annotations
label [type: string]
apoptotic process involved in salivary gland cavitation
created_by [type: string]
dph
creation_date [type: string]
2009-06-01T08:27:16Z
def [type: string]
Any apoptotic process in which the solid core of the gland is hollowed out to form the duct.
hasNarrowSynonym [type: string]
Description: 'apoptotic process involved in salivary gland cavitation'
Equivalent To
'apoptotic process'
and (BFO_0000050 some 'salivary gland cavitation')
SubClass Of
'apoptotic process involved in tube lumen cavitation'
BFO_0000050 some 'salivary gland cavitation'
SubClass Of (Anonymous Ancestor)
BFO_0000050 some 'tube lumen cavitation'
'apoptotic process'
and (BFO_0000050 some 'tube lumen cavitation')
BFO_0000050 some 'anatomical structure development'
'apoptotic process'
and (BFO_0000050 some 'anatomical structure development')
BFO_0000050 some 'anatomical structure morphogenesis'
'apoptotic process'
and (BFO_0000050 some 'anatomical structure morphogenesis')
Members
Target for Key
To use the reasoner click Reasoner->Start reasoner
Show Inferences

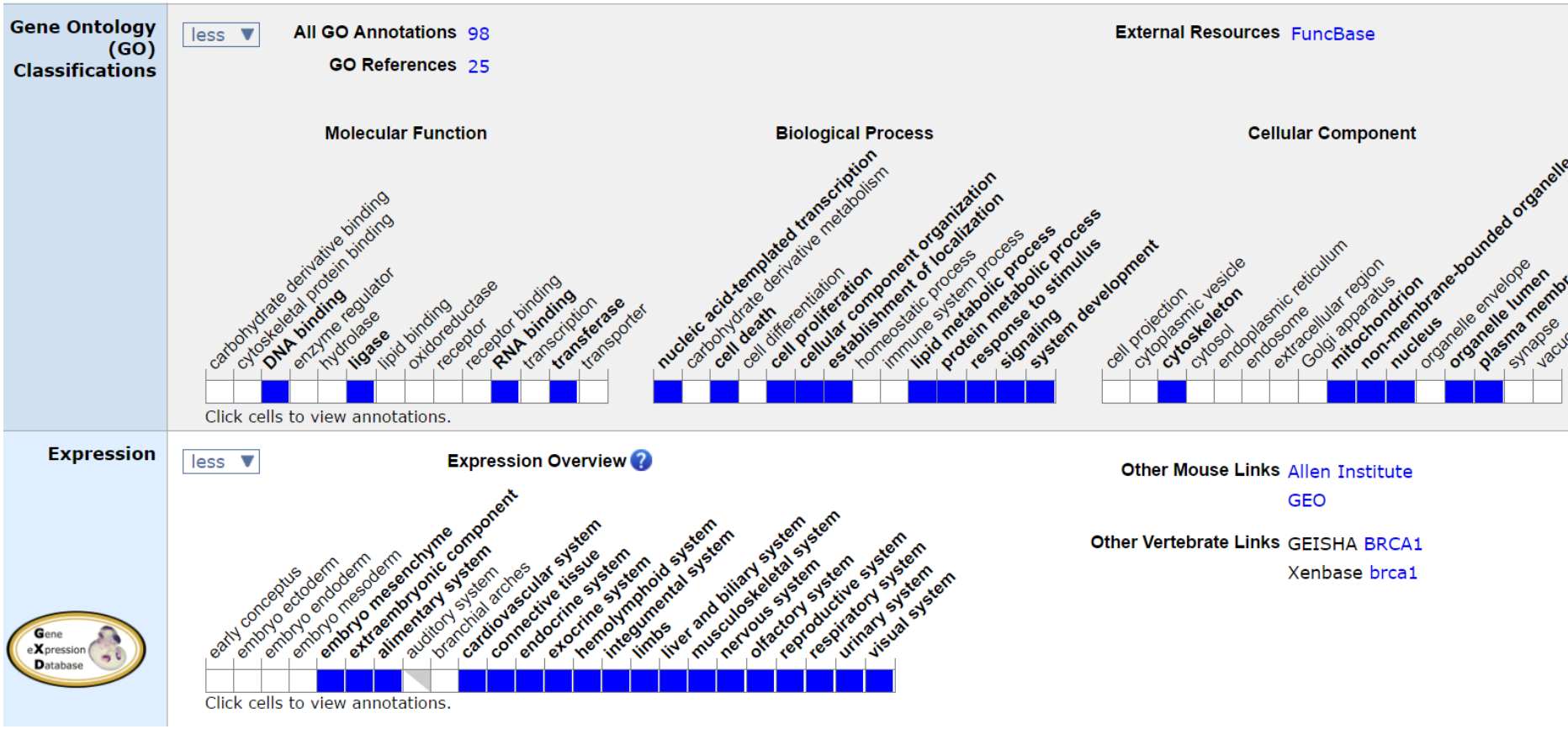
GO

- http://www.informatics.jax.org/



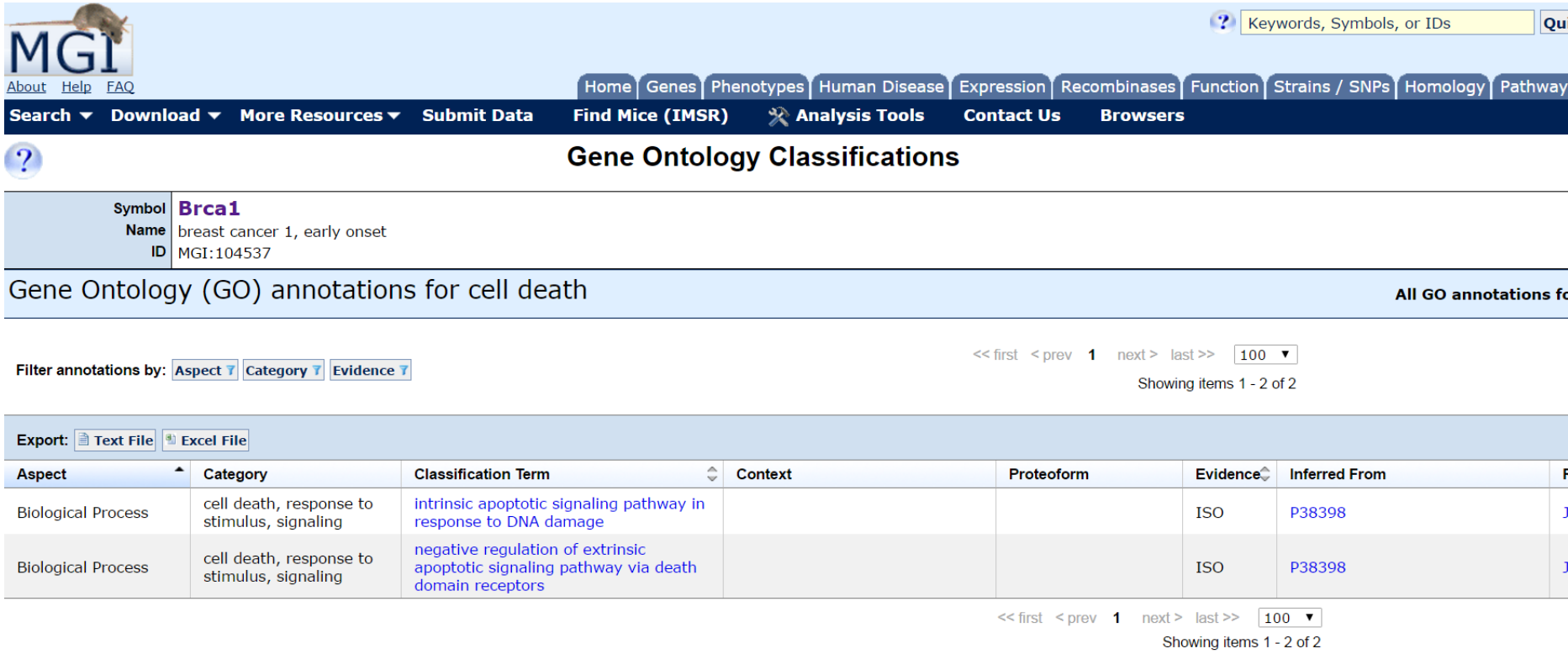
GO

- Select cell death



GO

- Gene Ontology (GO) annotations for cell death



Disease Ontology

- http://disease-ontology.org/
- http://www.berkeleybop.org/ontologies/doid.owl

- For further reading
- Disease Ontology: a backbone for disease semantic integration
- Lynn Marie Schriml,1,2,* Cesar Arze,2 Suvarna Nadendla,2 Yu-Wei Wayne Chang,1,2 Mark Mazaitis,2 Victor Felix,2 Gang Feng,3 and Warren Alden Kibbe3,*

- https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3245088/

DINTO

- DINDO
- Ontology for Drug drug interactions
- https://raw.githubusercontent.com/labda/DINTO/master/DINTO%201/DINTO_1.owl