

CogPO "Annotation" properties (i.e., properties used to annotate concepts in the ontology)
Derived from the NEMO Annotation policies (by Gwen Frishkoff, Ph.D., 9/2009)

COGPO will use one annotation ontology called: [COGPO_annotation_properties.owl](#)

...which is derived (but NOT imported) from NEMO_annotation_properties.owl file, which in turn is derived from the two NIF annotation ontologies:

http://ontology.neuinfo.org/NIF/Backend/OBO_annotation_properties.owl

http://ontology.neuinfo.org/NIF/Backend/BIRNLex_annotation_properties.owl

CogPO annotations will be a **closed set**, meaning, ALL annotations used by CogPO will be from CogPO_annotation_properties.owl, and no other annotation ontology (not core, not dc, not nif_annot). *The only exception might be rdfs:label, which is used to facilitate visualization of the ontology.*

The following annotation fields are the ONLY ones used for COGPO. **BOLD** indicates only one entry is allowed.

REQUIRED annotation properties:

- **COGPO_annotation_properties:hasURI** -- This property defines the CogPO URI for a concept. The URI is a non-human-readable label that also provides a stable web address for accessing the concept.
 - For an imported term, the URI should be the source ontology URI (if possible?). MIREOT recommends that the source term URI be included specifically.
- **COGPO_annotation_properties:preferredlabel** -- An underscore-delimited, human-readable label for a concept
 - Equivalent to rdfs:label and core:prefLabel.
 - We are re-creating COGPO_annotation_properties:label to keep all annotation properties for COGPO closed.
- **COGPO_annotation_properties:createdDate** -- [YYYY-MM-DD] The date when the current concept was created.
 - Equivalent to the same term in NeuroLex.
- **COGPO_annotation_properties:curator** -- This property defines the CogPO ontology curators who've contributed to the definition of a class. More than one instance of this property is valid. In the case of multiple curators, the individual curator name may also be associated with the definition, if that person was primarily responsible for it.
 - Range-restriction: COGPO_annotation_properties:COGPO_curator.
 - COGPO_curator enumerates the valid set of individuals designated as COGPO curators, which currently is {Jessica Turner, Angela Laird}.
- **COGPO_annotation_properties:definition** -- A natural language definition that adheres to Aristotelian constraints.
 - Equivalent to core:definition.
 - We are re-creating COGPO_annotation_properties:definition to keep all annotation properties for CogPO closed.
- **COGPO_annotation_properties:hasCurationStatus** -- This provides specific info on the level of curation a given class has received and is specifically designed to promote automated reporting on the status of classes in an ontology. It is likely some classes will require more than one instance of this property to fully describe their status. Is a type of skos_editorial_note.
 - Range-restriction: COGPO_annotation_properties:curation_status. The set of valid curation_status fields are as follows:

- *limbo_class* (resp., *limbo_property*) -- Indicates a class (resp., property) has been created by curators but has not been assigned to a superclass and therefore has an uncertain place within the ontology class hierarchy. By implication, the class has no definition or one that is minimally specified.
 - By implication, the next steps required are to: identify a valid definitionSource, specify the definitionSource, update the hasCurationStatus from *limbo_class* to *definition_incomplete* or *pending_final_vetting*.
- *definition_incomplete* -- Indicates a class has been assigned to a superclass and therefore has a clear place within the ontology class hierarchy and at least a minimal definition. However, the definition may require considerable amending before it complies with Aristotelian/OBO requirements for a good, noncircular definition.
 - By implication, the next steps required are to: verify the definition adheres to Aristotelian constraints, update hasCurationStatus from *definition_incomplete* to *pending_final_vetting*.
- *pending_final_vetting* -- Indicates a class needs to be vetted by two or more COGPO curation authorities.
 - By implication, the next steps required are to: vet the definition among COGPO consortium members, update hasCurationStatus from *pending_final_vetting* to *curation_complete*.
- *curation_complete* -- Indicates a class has been fully vetted by COGPO curation authorities and deemed ready for use.
- *Deprecated* -- indicates the concept has been retired (may not be necessary)

OPTIONAL annotation properties (main ones highlighted):

- *COGPO_annotation_properties:modifiedDate* -- [YYYY-MM-DD] The date when the current concept was last updated.
- *COGPO_annotation_properties:synonym* -- A lexical variant carrying the same latent semantic value.
- *COGPO_annotation_properties:acronym* -- A synonym consisting of a sequence of letters typically taken from the beginning of words of which either the preferred label or another synonym are composed.
- *COGPO_annotation_properties:definitionSource* -- A traceable source for the current definition.
 - o In free-form text, a source could be a person's name, a URI, an informal publication reference, a PubMed ID, whatever most succinctly identifies a source.
 - o We originally had an EnumerationClass of Definition_Sources but decided that was too restrictive.
- *COGPO_annotation_properties:example* -- An example of a given concept.
 - o Equivalent to core:example.
 - o We are re-creating *COGPO_annotation_properties:example* to keep all annotation properties for COGPO closed.
- *COGPO_annotation_properties:SourceOntologyURI* -- for imported terms, the URI of the source ontology, in keeping with MIREOT recommendations.
- *COGPO_annotation_properties:TargetDirectSuperClassURI* -- for imported terms, the URI of the parent class for the imported term, in keeping with MIREOT recommendations.
- *COGPO_annotation_properties:usageNote* -- Supplies some, possibly partial, information about the intended meaning of a concept, especially as an indication of how the use of a concept is limited in indexing practice. Could indicate connection of this term to a menu in BrainMap if relevant.
- *COGPO_annotation_properties:comment* -- A general comment.
 - o Please consider using *editorialNote*, *historyNote* or *changeNote* before attempting to apply a general comment.

- o We are re-creating COGPO_annotation_properties:comment to keep all annotation properties for COGPO closed. Otherwise, we consider COGPO_annotation_properties:comment to be equivalent to rdfs:comment and core:note.
- [COGPO_annotation_properties:historyNote](#) – Describes significant changes or historical justification for the current version of a concept.
- [COGPO_annotation_properties:changeNote](#) – Documents fine-grained changes to a concept, for the purposes of administration and maintenance.
- [COGPO_annotation_properties:editorialNote](#) – Supplies information that is an aid to administrative housekeeping, such as reminders of editorial work still to be done, or warnings in the event that future editorial changes might be made.
- [COGPO_annotation_properties:_deprecated](#) – A graveyard for deprecated classes.
- [COGPO_annotation_properties:_deprecatedProperty](#) – A graveyard for deprecated properties.
- [COGPO_annotation_properties:_limbo](#) – A class that has been created by curators but has not been assigned to a superclass and therefore has an uncertain place within the ontology class hierarchy. By implication, the class has no definition or one that is minimally specified.
- [COGPO_annotation_properties:_limboProperty](#) – A property that has been created by curators but has an uncertain place within the ontology. By implication, the property has no definition or one that is minimally specified.

TECHNICAL NOTES (from NEMO file originally):

AnnotationType -- A category designed to help data processing systems automatically determine how to process a given annotation.

- AnnotationProperty types are recommended by Alan Ruttenberg (2007-05-20) for representing fixed enumeration class instances for ontology class annotation.
- This is required, since sub-classing AnnotationProperties is not allowed in OWL-DL. Using this descriptive metadata AnnotationProperties, we can construct a chained code representation of a classification graph for AnnotationProperties very likely to be useful for information systems built to parse and process OWL content.
- The only valid values for this property are DC or SKOS (core) AnnotationProperty URIs.
- example:


```
<owl:AnnotationProperty rdf:ID="synonym">
  <annotationType>&core;altLabel</annotationType>
</owl:AnnotationProperty>
```

EnumerationClass -- A parent class for all enumerations used for class annotation. An enumeration class provides a strict set of possible values for a given property range.

- When creating or subclassing a new EnumerationClass, we recommend also specifying that all enumerations are mutually distinct.
- example: COGPO_curator is an enumeration class.