

The core ontology and its annotations

Overview

The Core Ontology and Its Use

The VIVO Core Ontology, which is installed automatically with the VIVO software and used at all installations, allows us to retrieve data across the national network in a consistent way. The core ontology should include only those terms and relationships that are necessary to achieve this goal. For example, we can easily imagine wanting to find librarians across different VIVO sites; therefore it makes sense for VIVO Core to have a class called Librarian (or possibly some other way of representing librarians). While there are many things the core ontology *could* include to describe librarians and the types of things that they do, the core ontology must limit itself to stating only those things that are always true across different institutions. This is no easy task. Incorporating and reusing existing ontologies can help us with this problem, but unless an existing ontology has been widely adopted it is likely to include its own assumptions that conflict with our needs. Being parsimonious in what we add to the core ontology will minimize our need to make later corrections, but the entire effort is a delicate balancing act. We must include enough classes and properties to enable useful national searches while keeping the ontology simple enough to be broadly applicable to very diverse institutions. The details of those institutions may then be modeled in additional ontologies that *extend* the core ontology without changing it.

Some Semantic Web Principles

Axioms and Annotations

The Web Ontology Language (OWL) allows us to assert a number of different types of axioms for constructing ontologies. An axiom might specify that one class is a subclass of another, or that one property is the inverse of another. These types of axioms have well-defined semantics according to the OWL specification. This has a precise technical definition, but for our purposes it is important to note only that these axioms have meaning that a reasoner can understand and use to draw inferences.

OWL also allows us to define properties that are used only for annotation. The annotations we make using these properties are technically also axioms in a strict sense, but they have no formal semantics and are ignored by reasoners. Annotation values in no way affect the inferences we can draw from Semantic Web data. VIVO uses quite a number of annotation properties to store internal metadata about the classes and properties in an ontology. VIVO makes annotations to represent the "display rank" of a property in an ontology, or record who is allowed to edit it or view its values. VIVO also uses annotations to indicate which "class group" or "property group" a given class or property is in. Class groups and property groups are not part of the OWL language: they are just mechanisms for displaying data in VIVO. Another common type of annotation is the human-readable label (sometimes called the "version for public display") of a class or property.

Monotonicity

You may hear VIVO project members use the term "monotonic reasoning" or come across it in the literature. It may sound esoteric, but the basic principle is quite simple and important to understand when working with local ontologies. Because reasoning on the Semantic Web is monotonic, any new data we discover while crawling the semantic web cannot invalidate or remove inferences that we may already have drawn. For example, if the core ontology says that all librarians are academic employees, any time we encounter a librarian we infer that he or she is also an academic employee. Imagine an institution where librarians are not academic employees. There is no new ontology axiom that this institution can add that will erase this inference for its librarians. This institution faces the tough choice of using the existing ontology's Librarian class and accepting the incorrect inference, or creating its own entirely different Librarian class and making it difficult for its librarians to be discovered on the national network.

Modifying Core Annotations

The axioms about the VIVO Core ontology are split into separate sets of files and handles separately by the VIVO software. The semantically meaningful axioms about classes and properties, as well as class and property labels and descriptions, are loaded in a special way and are not intended to be changed. The rest of the annotations about things like display ranks, class groups, and property groups are loaded in to each installation's database and are editable like any other data. The version 0.9 user interface does not make a visual distinction between what should be edited and what should not be edited, but refer to the lists below.

Editable annotations

Annotations on core classes

- Display Level
- Update Level
- Class Group
- Display Limit
- Display Rank
- custom entry form
- custom display view
- custom short view
- custom search view

Annotations on core object properties

- Property Group
- Display Level
- Update Level
- display tier for this property
- display tier for this property's inverse
- related object individuals to display without collapsing
- sort related individuals by
- sort direction
- data property by which to sort related individuals
- sort related object individuals of inverse property by
- inverse sort direction
- Related object individuals of non-inverse property to display without collapsing

Annotations on core data properties

- Property Group
- Display Level
- Update Level
- Display Tier
- Display Limit
- custom entry form