



Ontologies: a Foundation for Knowledge Capture

Stephen C. Waterbury
NASA / Goddard Space Flight Center

September 13, 2006



Introduction

- **Today's Audience:** everyone (including users and technical managers)
- **Main focus:** what ontologies are and how to apply them
- ... with just a few tantalizing glimpses into the underlying machinery
- If you hunger for more ... future presentations that I will give:
 - Ontology / software-developers' technical presentation (late fall)
 - Systems Engineers' technical presentation (early winter)



What is Knowledge Capture, and why do we need it?

- **NASA has a continuing “brain drain” problem**
 - *Constant loss of knowledge when folks retire*
- **We build (and do) complicated stuff!**
 - *As in any engineering organization, training is important (and mentoring can only do so much)*
- **We used complicated software tools**
 - *Sometimes knowledge gets “locked up” in unfriendly formats ...*
- **As in any complex activity, we tend to “repeat ourselves” and reinvent things we already have**
 - *Partly because we don’t KNOW what we know ... ☺*



Okay ... um ... what is it again?

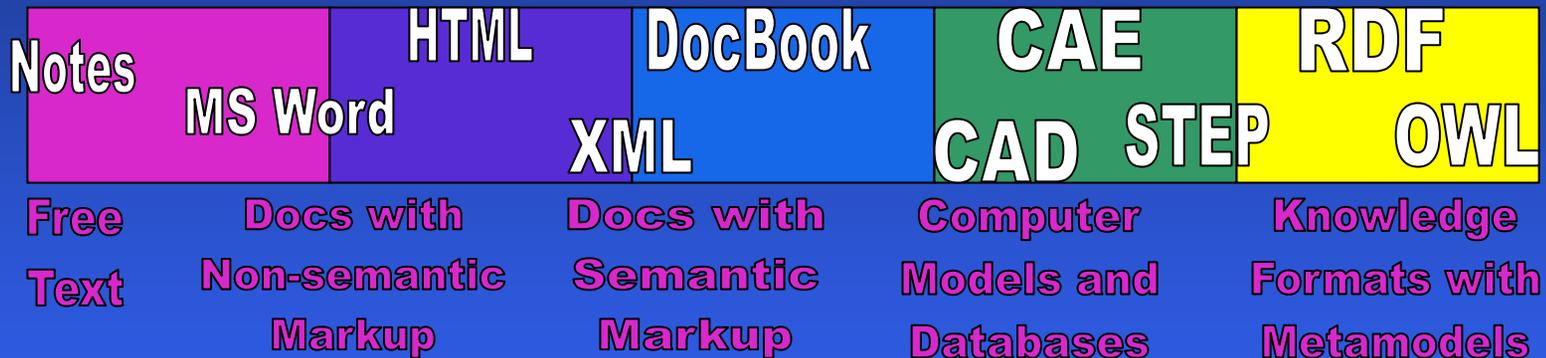
- The Knowledge Capture we're talking about here is the kind that enables man-machine collaboration
 - *Computers can help us work smarter, if we make THEM smarter*
- So how do we stuff this knowledge into the computers?
 - *A tricky problem, but we're getting there!*
- First, we need to represent our knowledge in a way that the computer can help us navigate and find things
 - *So the computer needs to know the "structure" of our knowledge*
- So ... first, WE have to know the structure of it!
 - *That's where building these ontologies comes in ... ☺*



Right ... structure!

Information Spectrum

← Less Semantic Structure More Semantic Structure →



- Less structure -> less “knowledge” the computer can help us with
- How can we add more structured information objects?
 - Better, friendlier, structured authoring tools for new information
 - For legacy data, translation and “mining” – neither easy nor cheap!



What's are RDF and OWL?

- **RDF = “Resource Description Format”**
 - *A formalism for describing “resources” – i.e., things on the web*
 - *Has an associated XML format: RDF/XML*
- **OWL = “Ontology Language for the Web”**
 - *A language for representing ontologies*
 - *Consistent with RDF, and can be encoded in RDF/XML (and other, more human-friendly, formats)*
 - *Inherits from and benefits by much academic work, quite rigorous*
 - *For complete logic, we still need “rules” ... beyond OWL’s scope*
- **RuleML and SWRL (Semantic Web Rule Language)**
 - *Can be used with OWL to express rules*



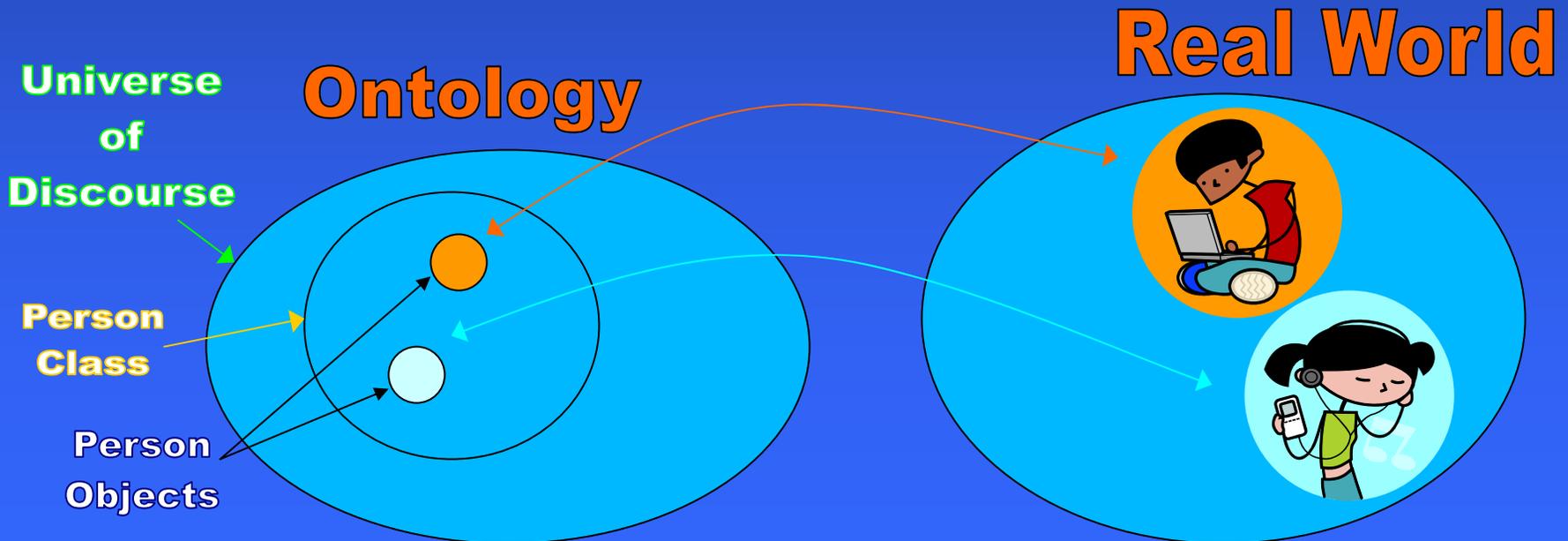
What's an Ontology?

- **A logical description of a world**

- From a point of view: could be just the things that interest **us**
- Formal: defined in a branch of logic called “Description Logic”

- **A “model”**

- Model Theory ... maps: Ontological “objects” \leftrightarrow things in Real World





Ingredients of an Ontology

- **Classes**

- *A Class is basically a “set” of things with some defining characteristic(s)*
- *Classes can have sub-Classes, which can be thought of as subsets*

- **Properties**

- *A Property can be thought of as an “attribute” of a Class’s members*
- *It’s represented just like a mathematical “Relation” between two sets*
- *Can be a “Data Property” or an “Object Property”*
 - *Data Properties point to “datatypes” – e.g. “age” (a number)*
 - *Object Properties point to other objects – e.g. “parent” (another Person)*