

Ontologies Ontologies Everywhere – but Who Knows What to Think?

Michael Uschold

*Protégé Users Conference
Keynote Address*

*July 24, 2006
Stanford, CA*



About this Talk

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

What it is NOT:

- Overview of Ontology & Protégé Activities at Boeing
- A Detailed Look at a Particular Project or Approach
- Strongly focused on Protégé, *per se*

GOALS:

- To quench a thirst for understanding ‘ontologies’
- To cover interesting and fun ontology-related topics
- Stimulate thinking about key issues
- Something for everyone (including a short nap?)

Ontologies Ontologies Everywhere, but...

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

Who Knows What to Think?

RDF Schema
 Glossary
 UML
 IDEF5
 Controlled Vocabulary
 Unambiguous Semantics
 Iode
 IDEF1X
 Philosophy vs. IT
 T-box
 Meta Models
 Knowledge Base
 RDF
 Triple Store
 Thesaurus
 Flogic
 Visual Knowledge
 Description Logic
 Database Schema
 Swoop
 Object-Model
 Ontoclean
 Topic Maps
 XML Schema
 A-box
 Cerebra
 Taxonomy
 Protégé
 Unicorn
 OWL

Three Questions

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

- 1. What is the difference between an Ontology and a: Taxonomy? XML/DB Schema? Thesaurus? UML Model? Ontology – as a branch of philosophy?**
- 2. When people say things like:**
“Ontologies Have Unambiguous Semantics”
 - *What do you think they mean?*
 - *Do you believe them?*
 - *Why or Why not?*
- 3. Are Natural Language Comments important for Ontology Engineering?**
 - *Why or Why not?*
 - *Is there a way to leverage natural language semantics?*

Questions to the AUDIENCE!

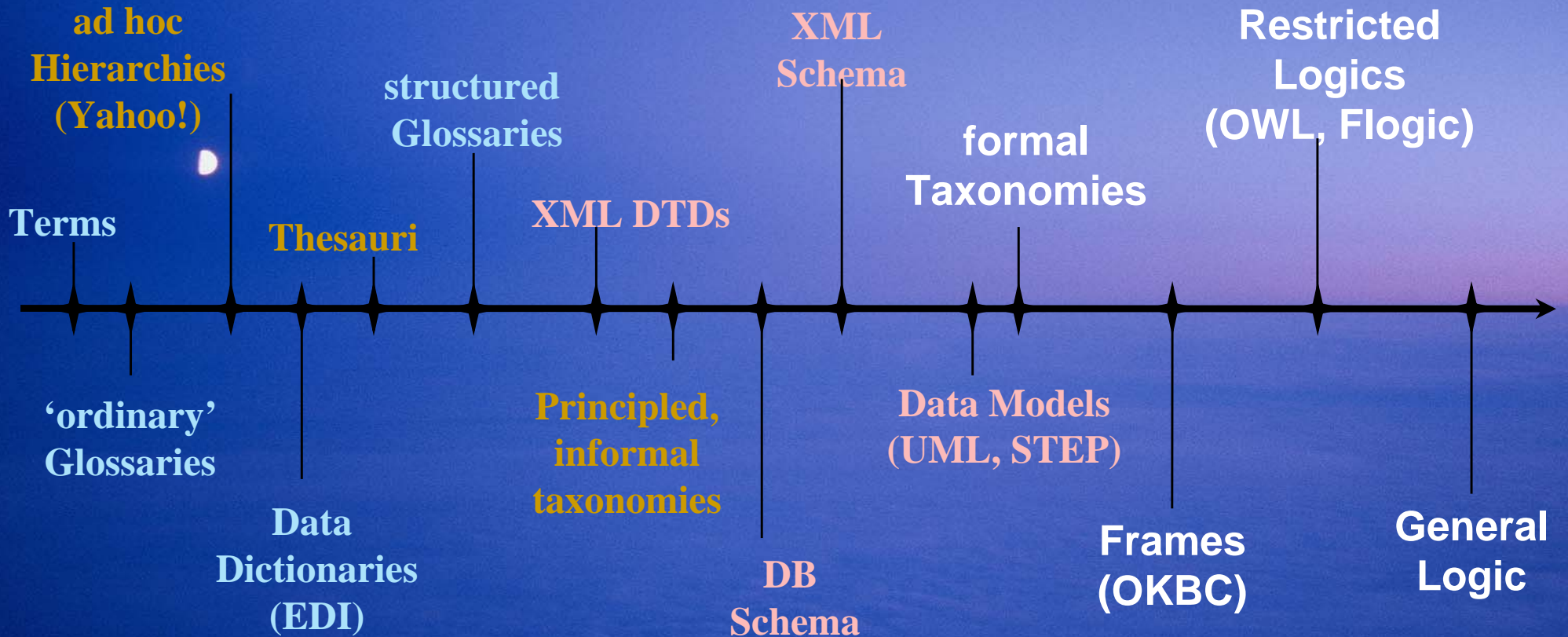
A Plethora of 'Ontology-Like Things'

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

Glossaries / Controlled Vocabularies

Data and Document Metamodels



Informal Taxonomies and Thesauri

Formal Knowledge Bases & Inference

Many Different Ways of Expressing Meaning

Common Threads among 'Ontology-Like Things'

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

- These terms reflect **ill-defined notions** that are used in many different ways by many different communities.
Meta-Model, Schema, Ontology, Glossary, Taxonomy, Thesaurus, Controlled Vocabulary, Data Dictionary, Data Model, Object Model, etc
- A way for a community to agree on **common terms** for **capturing meaning** or **representing knowledge** in some domain.
- All have been called '**ontologies**'

Differences

- What **notation or language** is used to specify the meaning?
- How is **meaning** specified?
- What is the **purpose** for representing information about a given domain?

Example: Controlled Vocabulary / Glossary

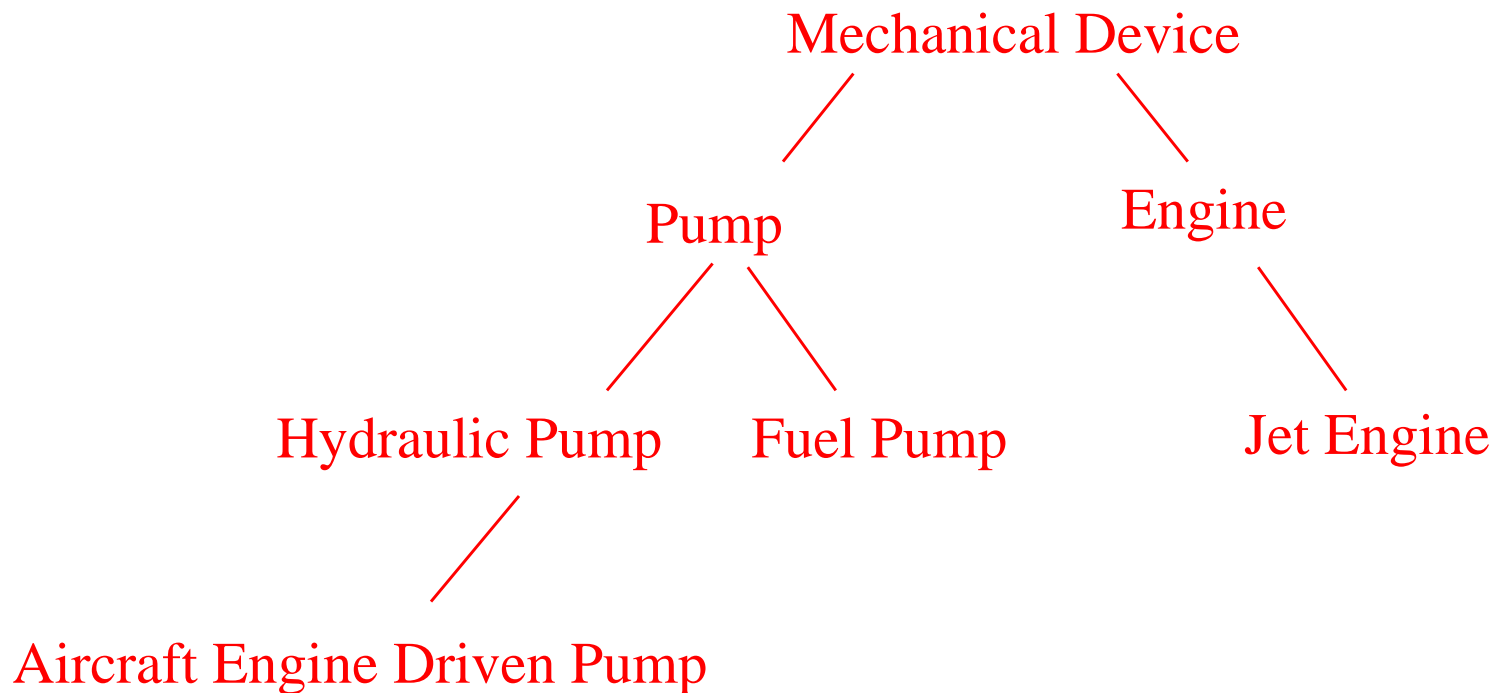
Pump: “A mechanical device for raising, compressing, or transferring fluids”

Engine: “a machine that turns energy into mechanical motion”

Mechanical Device: “a physical device with parts that move relative to each other”

Example: Taxonomy

— = **Generalization**



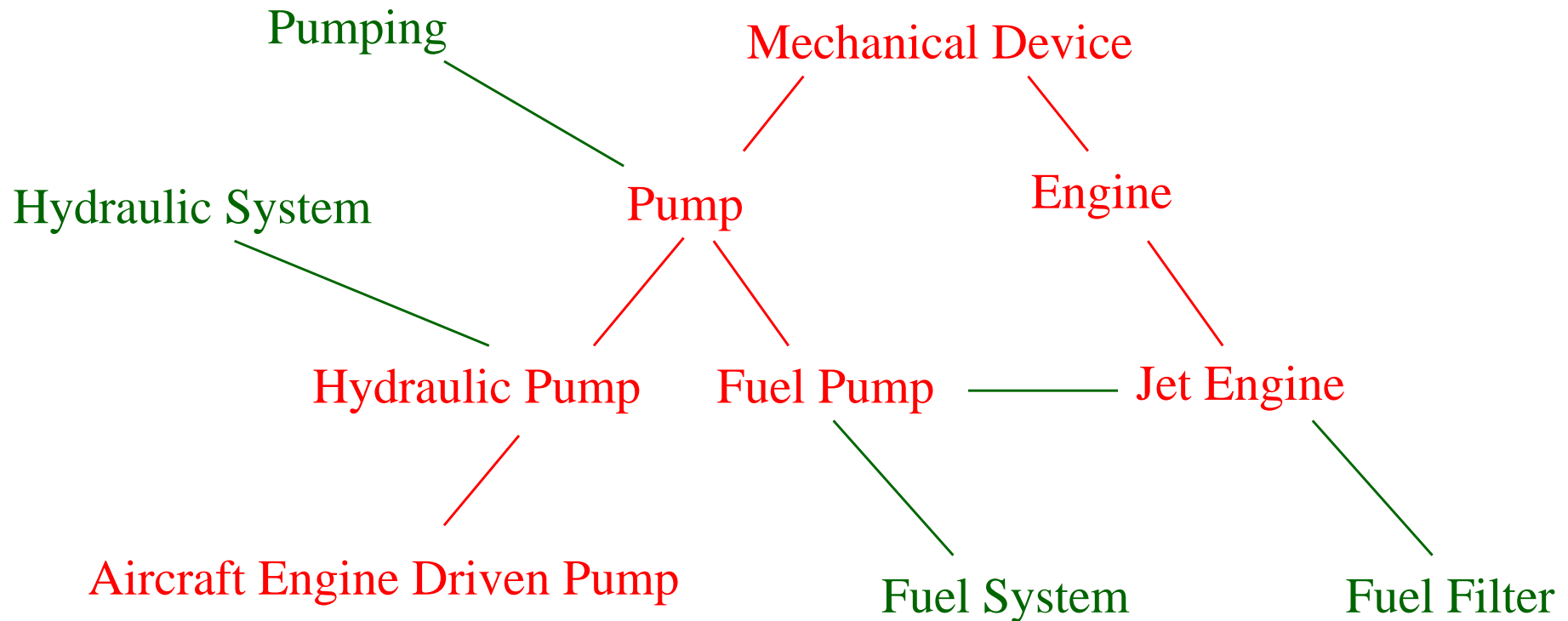
Example: Thesaurus

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

— = **Broader Term**

— = **Related Term**



Ontology: Strict Taxonomy + Formal Relationships

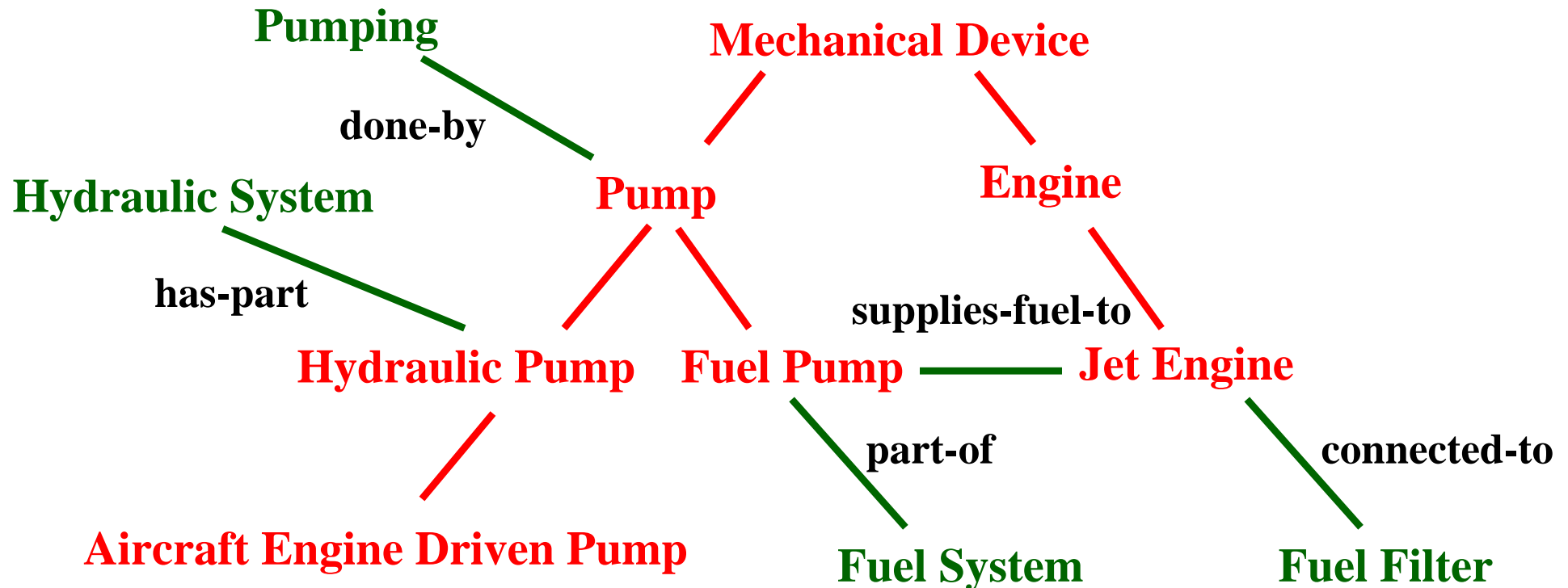
Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

- **Taxonomy with multiple link types, each with precise meaning, is usually called an “ontology”.**

— = Generalization

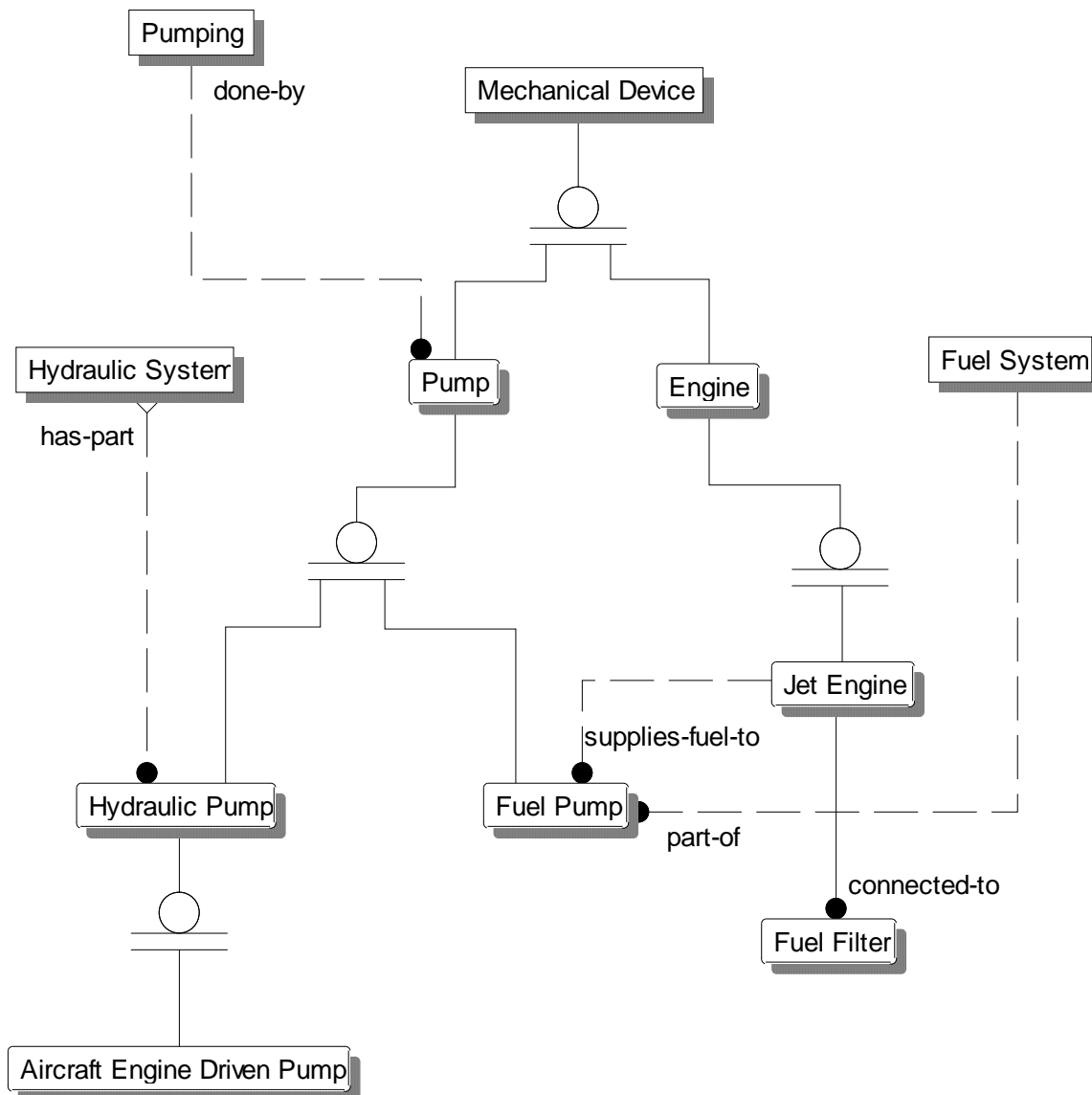
— = Other Relationships



Logical Model: IDEF1X Notation

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology



- **Just like the formal ontology**
- **Some key differences**

Database Schema vs. Ontologies

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

Focus on DATA

DB Constraints

- to ensure integrity
- may hint at meaning

No ISA hierarchy

SQL Engines

- querying, views
- data integrity

Instances Central

Data Dictionary

- separate artifact

FOCUS on Meaning

Ontology Axioms

- to specify meaning
- maybe for integrity

ISA Hierarchy is Backbone

Theorem Provers

- infer new information
- ensure consistency

Instances Optional

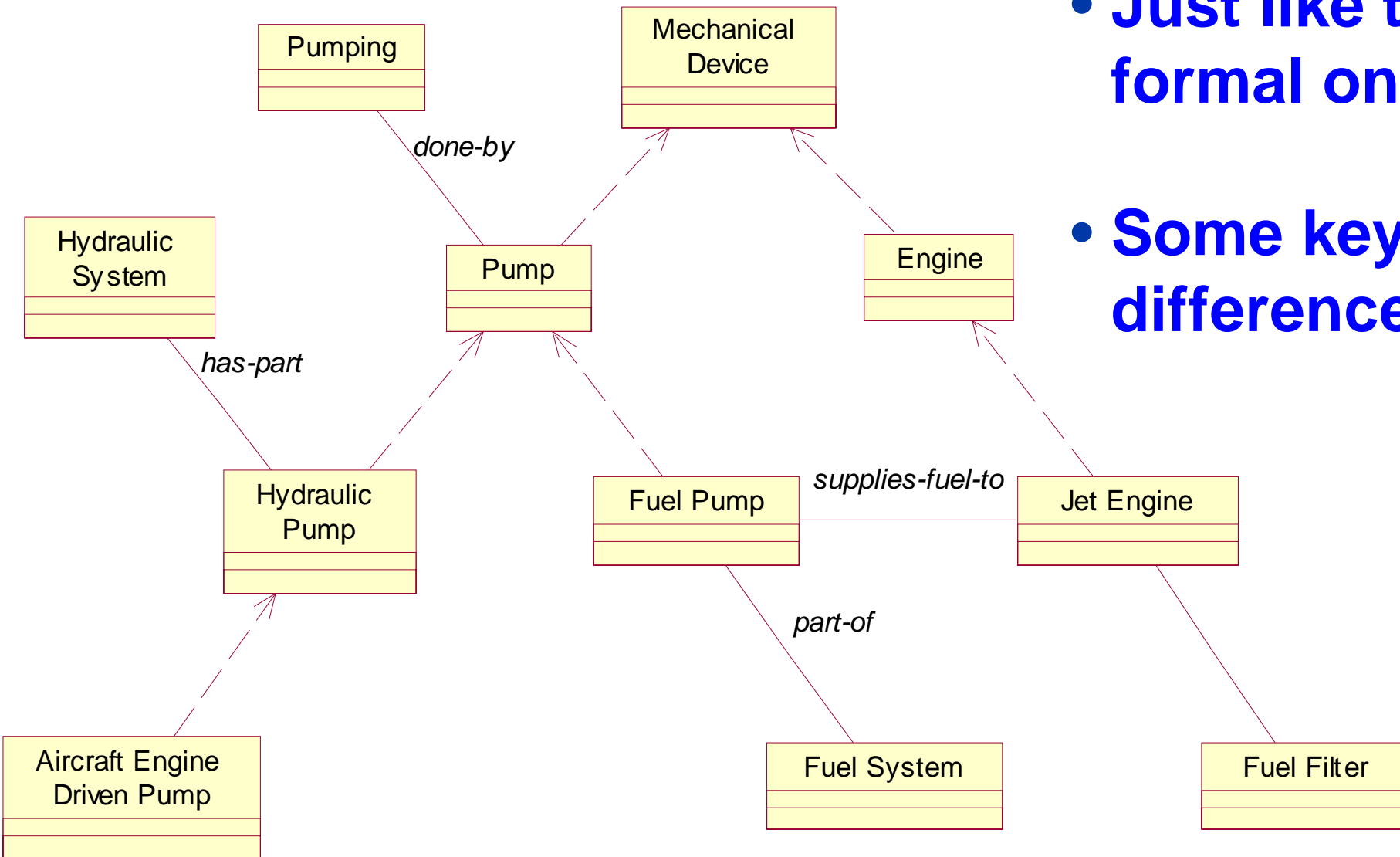
'Comments'

- part of the ontology

UML Class Diagram

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology



- **Just like the formal ontology**

- **Some key differences**

Object Models

vs.

Ontologies

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

Increasingly Formal

Static & Less Flexible

- classes fixed
- instance of same one class forever

Attributes

- defined locally to a class
- no hierarchies

Class Hierarchy

- for structuring convenience & object creation

Rigorously Formal

Dynamic & Flexible

- change/edit classes at runtime
- membership of multiple classes changes over time

Relations / Properties

- independent of a specific class
- hierarchies

Class Hierarchy

- to support inference

Summary: Comparing Ontology-Like Things

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

	Ctld. Vocab	Taxonomy	Thesaurus	Ontology	Data Models	Object Models
Definition	Defined terms, controlled	Controlled vocab. in a hierarchy.	Controlled vocab. in a network.	<i>specification of a conceptualization</i>	Specification of DB structure	Specification of a software application domain
Notation	Free text, Definition structure varies.	Strict: tree Or: multi-parent	Broader/narrower (maybe taxonomy) Gnl. association;	Logics, Taxonomy as backbone + atts. & relations.	e.g. ER diagrams Entities & Relations	Hierarchy of classes, rel's attributes & methods
Meaning	Nrl lang def's Dictionary; common usage	Nrl lang def's + meaning of link Strictness & Precision varies. Isa, partOf, similarTo ...	Nrl lang def's + meaning of links. B/N: various mng's Gnl Assoc'n: no specific meaning	Logics w/ fml. semantics. Isa hierarchy; Dom/Range constraints; cardinality. Nrl. language comments in the ontology.	Precise, not logic-based. Focus on data, not meaning (e.g. toss rel'n names). Data dictionary separate.	Increasingly formal. Isa hierarchy, Aggregation / Composition, Dom/Range constraints; cardinality.
Purpose	Human communication (HC)	HC + Structure info. base; browsing	HC + Structure digital libraries; indexing, browsing & search	Union of all the others & more.	HC + Structure (and validate) databases.	HC + Structure software systems.

Ontology: Philosophy vs. Information Technology

Similarities:

- Both require careful **study/analysis** of **what exists** i.e., **conceptual modeling**
 - **Objects**: concrete/abstract, real/ideal, [in]dependent
 - **Relationships**: dependencies, properties
- Both **document the output** of the study/analysis:
 - in 'natural language'
 - as a **logical theory**
 - for **sharing and reuse** in a larger community.

Ontology: Philosophy vs. Information Technology

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

Similarities:

- Both use **natural language** as a **source of knowledge** to reveal the objects and relationships of interest.
- Both use **formal logic** as a key **analytical tool**.

Ontology: Philosophy vs. Information Technology

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

Differences: Purpose

- Ph: **The end:** an **understanding** of the world
Why: *“because its there”*
- IT: **The beginning:** an **understanding** of the world
The middle: a **machine-sensible** model
The end: a **software application** that does something useful

Why: because it **helps the bottom line**

Range of specific purposes ...

Ontology: Philosophy vs. Information Technology

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

More Differences:

- Ph: **Grammatically, not countable** vs.
IT: Can have **one or more 'ontologies'**;
- Ph: **Subject area** extremely **general** vs.
IT: Focused on a **specific** domain (usually);
- Ph: **Main focus** is on **what is, ultimate truth** vs.
IT: What is **useful** or **important** in a specific context;
- Ph: **Source of knowledge** is the **world** itself vs.
IT: **experts, books**, etc in a given field of study.

Ontology: Philosophy vs. Information Technology

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

More Differences: Output of the study/analysis

- Ph: **Natural language output** is a **formal academic paper** vs.
IT: may be a paper, or may just be **informal documentation**;
- Ph: **Computational output**, **None** (historically) vs.
IT: **Machine-sensible model to support computer processing**
 - **browsing and search**
 - **automated reasoning and inference** .

Varying degrees of formality

Three Questions

1. What is the difference between an Ontology and a: Taxonomy? XML/DB Schema? Thesaurus? UML Model? Ontology, as a branch of philosophy?
2. When people say things like:
“Ontologies Unambiguously Represent Knowledge”
 - *What do you think they mean?*
 - *Do you believe them?*
 - *Why or Why not?*
3. Are Natural Language Comments important for Ontology Engineering?
 - *Why or Why not?*
 - *Is there a way to leverage natural language semantics?*

"Unambiguous Semantics"

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

- “The benefit of ontologies is the rich, and **unambiguous, semantics** they define.
[Folktologies: Beyond the Folksonomy vs.. Ontology Distinction](#)
- **Unambiguous semantics** allow question answerers to infer that objects are the same, objects are related, objects have certain restrictions, ...
[SWANS: Substance of the Semantic Web](#)
- “OWL-S is a ... core set of markup language constructs for describing the properties and capabilities of ... Web services in **unambiguous, computer-interpretable form.**”
[OWL-S 1.1 Release](#)
- Ontologies can provide [a] way of explicitly specifying the **semantics** for each terminology in an **unambiguous** fashion.
[Ontology and Semantic Interoperability](#)

Example: Semantics and Ambiguity

**Using the following different languages/notations, express the meaning of the terms:
pump; fuel pump & has part**

- **a glossary or controlled vocabulary**
- **a taxonomy**
- **a thesaurus**
- **UML**
- **OWL**

What can we say about which alternatives are more or less "semantically unambiguous"? Why?

Meaning and Ambiguity

We've see most of this already, except for OWL.

```
Pump a owl:Class ;  
  rdfs:subClassOf uo:PhysObj.  
  rdfs:subClassOf [a owl:Restriction;  
    owl:onProperty hasPart;  
    owl:someValuesFrom Piston].
```

```
FuelPump a owl:Class;  
  rdfs:subClassOf Pump.
```

```
hasPart a owl:TransitiveProperty, owl:ObjectProperty;  
  rdfs:domain uo:PhysObj;  
  rdfs:range uo:PhysObj.
```

What do we Know from this Simple Ontology?

From the formal semantics of OWL, we know that:

- **Every Fuel Pump is also a Pump.**

IF FuelPump(?X) **THEN** Pump(?X).

- **Every Pump has at least one Piston as a Part.**

IF Pump(?X)

THEN PhysObj(?X) &

(Exists ?P) Piston(?X) & hasPart(?X, ?P).

- **The hasPart relationship is transitive**

IF hasPart(?X, ?Y) & hasPart(?Y, ?Z)

THEN hasPart(?X, ?Z).

What do we Know from this Simple Ontology?

From the formal semantics of OWL, we know that:

- **Both instances in any hasPart relation are Physical Objects**

IF `hasPart(?X,?Y)`

THEN `PhysObj(?X) & PhysObj(Y)`.



Ooops...

For Example

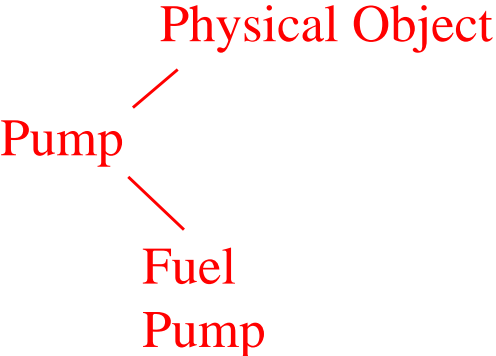
IF `hasPart(university_of_edinburgh,
ai_department)`

THEN `PhysObj(ai_department) &
PhysObj(university_of_edinburgh)`.

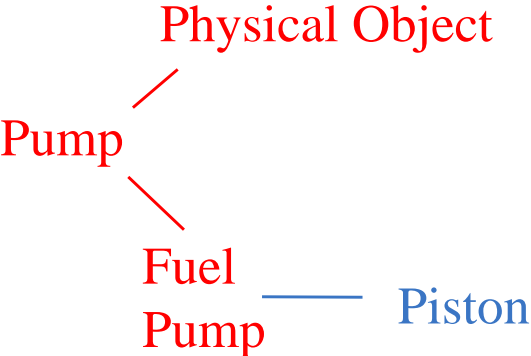
- Some additional things about 'PhysObj' from the UO ontology.
- Little else....

Which are More/Less "Semantically Ambiguous?"

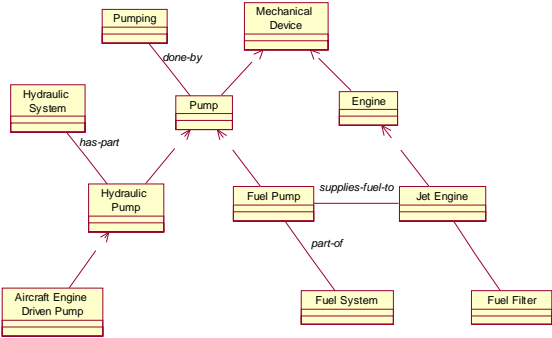
Controlled Vocabulary: A mechanical device for raising, compressing, or transferring fluids.



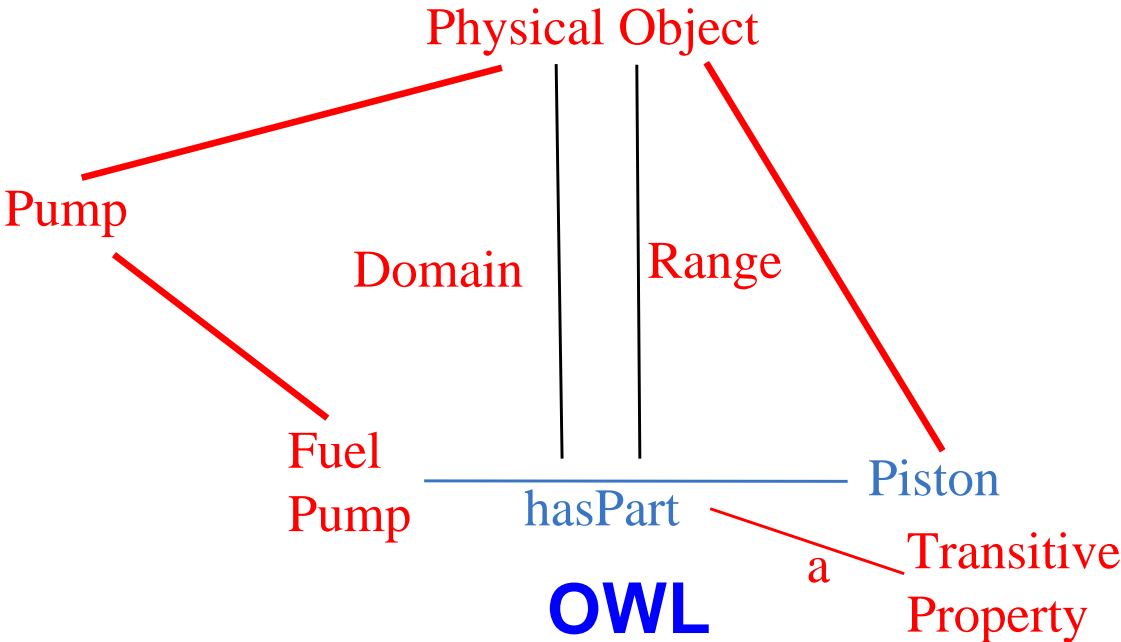
Taxonomy



Thesaurus



UML



OWL

"Semantically Unambiguous" is itself Ambiguous!

Semantics of **WHAT**?

- language constructs & inference procedures
 - e.g. `Class`, `subclassOf`
- terms & expressions in the language
 - e.g. `subclass(FuelPump, Pump)`
`transitiveRelation(hasPart)`

Unambiguous for **WHO**?

- an end user of an application?
- the computer?
- someone writing code that uses an ontology?
- semantic technology software developers?

What **KIND** of Semantics?

- model theoretic, axiomatic, real world

How Does the Machine Know What to Do?

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

Humans encode that knowledge into the machine

Either:

- **Hardwire** the meaning into the application
- Encode procedures to **discover** that meaning (at least partially)

More Semantics means Less need for hardwiring.

For Example: Finding Relevant Documents

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology



Simple Task:

Find documents about mechanical devices.

The purpose of this review is to remind operators of the

existence of the Operations Manual Bulletin 80-1, which provides

information regarding flight operations with low fuel quantities,

and to provide supplementary information regarding main tank

boost pump low pressure indications.

747 **FUEL PUMP** LOW PRESSURE INDICATIONS

When operating 747 airplanes with low fuel quantities for short

Semantics for the Computer



Hey, I know this ontology, so now I know something about Fuel Pump.

What the heck is a Fuel Pump?

The purpose of this review is to remind operators of the existence of the Operations Manual Bulletin 80-1, which provides and to provide supplementary information regarding main tank boost pump low pressure indications.747 <concept id=fuel-pump>FUEL PUMP</concept> LOW PRESSURE INDICATIONS
When operating 747 airplanes with low fuel quantities for short

```
Shared Hydraulics Repository (SHR)
Pump
a owl:Class ;
rdfs:comment "A mechanical device for raising, compressing, or transferring fluids.";
rdfs:subClassOf PhysicalObject;
rdfs:subClassOf
[ a owl:Restriction ;
owl:hasValue Piston ;
owl:onProperty hasPart
].
```

Semantic Annotation

<concept id=fuel-pump>FUEL PUMP</concept>

fuel-pump

```
a owl:class;
rdfs:subClassOf SHR: pump
```

What do they really mean by "Unambiguous Semantics"?

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

The **language** of OWL has an explicit formal semantics.

Major benefit related to **inferencing**.

Enables the **human** to encode inference engines that meet the language specification.

Helps to ensure correctness and consistency among different implementations.

BUT: don't be fooled...

The Same Axioms Can Mean Different Things

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

```
Pump a owl:Class ;  
rdfs:subClassOf uo:PhysObj.  
rdfs:subClassOf [a owl:Restriction;  
                 owl:onProperty hasPart;  
                 owl:someValuesFrom Piston].
```

```
FuelPump a owl:Class;  
         rdfs:subClassOf Pump.
```

```
hasPart a owl:TransitiveProperty, owl:ObjectProperty;  
        rdfs:domain uo:PhysObj;  
        rdfs:range  uo:PhysObj.
```

The Same Axioms Can Mean Different Things

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

```

 $\exists^* \neq \checkmark$  a owl:Class ;
  rdfs:subClassOf uo: $\exists^* \sim \dots \acute{\alpha} \dots \text{\$}$  .
  rdfs:subClassOf [ a owl:Restriction;
    owl:onProperty  $\exists^* \sim \dots \acute{\alpha} \sim \text{\$}$  ;
    owl:someValuesFrom  $\exists^* X \dots \text{\$}$  ] .

```

```

 $\exists^* \cup \neq \exists^* \neq \checkmark$  a owl:Class;
  rdfs:subClassOf  $\exists^* \neq \checkmark$  .

```

```

 $\exists^* \sim \dots \acute{\alpha} \sim \text{\$}$  a owl:TransitiveProperty, owl:ObjectProperty;
  rdfs:domain uo: $\exists^* \sim \dots \acute{\alpha} \dots \text{\$}$  ;
  rdfs:range uo: $\exists^* \sim \dots \acute{\alpha} \dots \text{\$}$  .

```

The Same Axioms Can Mean Different Things

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

```
DigitalSLR a owl:Class ;  
  rdfs:subClassOf uo:DigitalCamera.  
  rdfs:subClassOf [a owl:Restriction;  
    owl:onProperty hasMoreMegapixels;  
    owl:someValuesFrom CameraPhone].
```

```
CanonDigitalSLR a owl:Class;  
  rdfs:subClassOf DigitalSLR.
```

```
hasMoreMegapixels a owl:TransitiveProperty,  
  owl:ObjectProperty;  
  rdfs:domain uo:DigitalCamera;  
  rdfs:range uo:DigitalCamera.
```

Don't Be Fooled

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

Ontology Terms are [virtually] Always Ambiguous!!!

- ... **even if there is an unambiguous formal semantics** for the ontology language.
 - e.g. DC: Creator, or DC: Date, or MyLogistics: DeliveryDate

- **Adding more axioms can remove ambiguity.**

Only add axioms if needed, ambiguity is inevitable.

- **All meaning bottoms out in natural language & common usage.**
- **Fully automated discovery of meaning – impossible?**

Three Questions

1. **What is the difference between an Ontology and a: Taxonomy? XML/DB Schema? Thesaurus? UML Model? Ontology, as a branch of philosophy?**
2. **When people say things like: “Ontologies Unambiguously Represent Knowledge”**
 - *What do you think they mean?*
 - *Do you believe them?*
 - *Why or Why not?*
3. **Are Natural Language Comments important for Ontology Engineering?**
 - *Why or Why not?*
 - *Is there a way to leverage natural language semantics?*

The Role of Comments in Ontology Engineering

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

Many people think

- "they're just comments" and don't pay that much attention.

BUT comments are valuable, they:

- can be a **vehicle for reaching agreement**
- can **specify** and/or **document** the formalization (i.e. logical theory)
- fill in many [semantic] blanks in the logical theory, [which in turn]
- helps ensure a **shared understanding** [that is]
- critical for **collaborative** ontology engineering [and]
- very helpful for ontology **evolution & maintenance**

Therefore, it is important:

- to **have** comments,
- of **high quality!**

Comments / Documentation / Annotations

- **Overall Goals:**
 - convey **intended meaning**
"to provide a human-readable description of a resource" [W3C]
 - sometimes extra-logical instructions for machine processing
- **Factual knowledge critical to understanding a concept**
(if you're lucky):
 - **intrinsic properties & distinguishing characteristics**
 - **examples:** both positive and negative
- **Other background information**
 - references, dates, authors, ...historical information
 - things that may **not** be appropriate **for formalization**
 - **whatever you wish...**

Approaches / Guidelines for "Good" Definitions

Ontology Engineering:

- **Enterprise Ontology Experiences** – Uschold et al., 1998
Informal general guidelines, some early ideas, small part of larger effort, no computer implementation.
- **Glosses** – Jarrar, 2006
Informal general guidelines, part of a larger ontology engineering effort, exploit linguistic resources; existing prototype.

Data Dictionaries:

- **Database Lexicography** – Coen, 2002
Highly structured guidelines, major focus: better data dictionaries & databases; exploit linguistics technology & resources; implementation – industrial use.

Good Definitions and "Glosses"

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

- **Start with the principle supertype;**
e.g. Invoice: a business document that ...
- **Focus on intrinsic characteristics, distinguishing properties & examples;**
- **Refer to other defined terms;**
(prefer clarity to 'nice' text)
- **Ensure consistency among all definitions;**
- **Use existing terms before** inventing new ones –ensures understandability and future interoperability;
- **Accurately specify/reflect axioms** of the logical theory.

Opportunities & Challenges

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

- **Automatic generation of text from logical definitions**
 - easy to do poorly...hard to do well.
- **Automatic generation of logical definitions from carefully crafted text.**
 - major thrust of research, knowledge acquisition
- **"Semantic synch" between logical axioms and text definitions**
 - is anyone doing this?
- **Semantic dependency analysis & consistency checking of text definitions (+ maintenance guidance)**
 - **database lexicography**

Database Lexicography

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

Lexicography: art or craft of compiling, writing and editing dictionaries.

Database Lexicography:

- lexical dependency analysis to improve data design.

Why?

- data model design is often the most expensive, difficult and important phase of software development.
- better maintenance: avoid fossilized data designs

How?

- tool support for building better data models
- ensure **semantic consistency**
- **design for change**

Database Lexicography: Basic Approach

- Treat **data dictionary** as a **first class object**

Rules for "good definitions"

- Identify and measure **semantic dependencies**,

This supports ability to:

- Detect **circularity**
- Measure "lexical stability"
- Detect "dependency inversions"
- Predict impact of changes

RESULTS:

- **Guides** the data designer in how to correct problems;
- Improved **data design & evolution.**

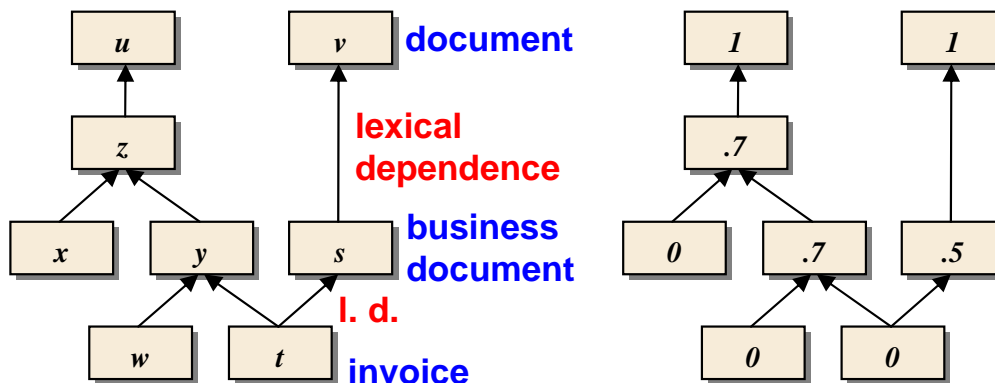
Lexical/Semantic Dependency

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

- **Data dictionary entry: <term, definition>**
e.g. <"invoice", "a **business document** that ...">
e.g. <"**business document**", "a document that ...">
- Entry <t,d> is **lexically dependent** on <t',d'> IF t' is contained in d (i.e. substring).
- Formal metrics to determine 'desirability' of semantic dependencies.

i.e. "defined in terms of"

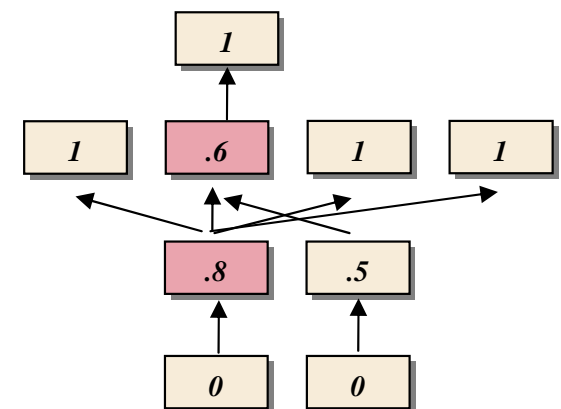
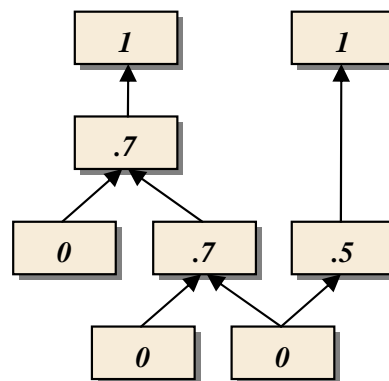
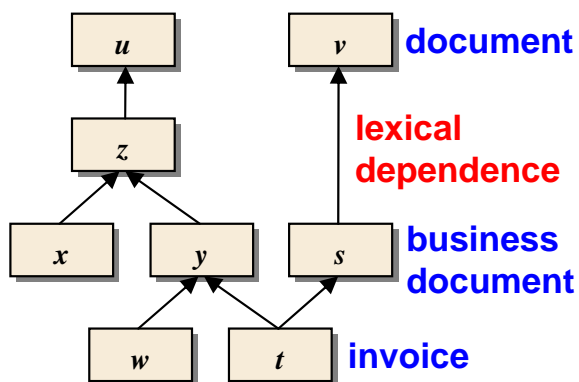


Lexical/Semantic Dependency

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

- **Lexical Stability**: measures the difficulty of modifying the semantics of a dictionary entry.
 - calculated from lexical dependencies
 - more stable means more difficult
- **Dependency Inversion**: a more stable definition depends on a less stable one
 - evidence of fossilization (not good!).



Rules for Data Dictionary Entries

General:

- Succinct and precise;
- Every lexical dependency must be reflected in the physical or logical model;

Attributes and Domains

- Every domain is defined in terms of a primitive data type, or an exhaustive enumeration of a set of typed values;

Inheritance & Aggregation:

- Each holonym/hypernym lexical dependency is reflected in the logical or physical model and vice versa.

Rules for Data Dictionary Entries

Existential Entailment

- Existence dependence in the model requires lexical dependence in the definitions;

Interface

- Definitions relating things at the physical / logical model interface must be lexically dependent.

Collectively: these rules guarantee that the **data dictionary** presents an **integrated representation** of the **logical** and **physical data models**. I.e. "semantic synch"

Managing Change

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

- **Compute lexical stability & find dependency inversions**
- **Predict the impact of changes**
- **Suggest preferred changes to minimize impact**
- **Significant cost reductions in the long term**
- **No Free Lunch!**
- **Analogous to OntoClean?**

Database Lexicography: Summary

- **All meaning is grounded in natural language semantics.**
- **There is a lot of meaning in a Data dictionary that is not exploited** (e.g. in ER modeling tools).
- **Data dictionary *should* define elements of data models.**
- **Lexer**: a tool for managing knowledge in data dictionaries.

So What Does all this Have to Do with Ontologies?

- **Leverage linguistic knowledge to reduce ambiguity.**
- **Perhaps something analogous can be done to support ontology engineering, and evolution?**
- **Age of "Ontology-Driven Software Engineering" where Ontologies replace Database Schema?**
- **Open research question.**
- **With much potential for practical benefit.**

Takeaways

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

- **Many ways to express meaning**
 - much overlap
 - many important differences: purpose; culture, formality
- **Formal semantics not the be all and end all**
- **Take comments seriously!**
 - Use principled guidelines for defining things.
 - Leverage natural language semantics
 - You can do serious analysis on comments & achieve major benefits
 - Maintain "semantic synch"
 - **Build Plug-Ins for Protégé?**

Acknowledgements

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

- **Woody Pidcock for material comparing different ‘ontology-like things’**
- **Dave VanRossum & Janet Jones for preparing data model & UML diagrams**
- **Chris Menzel in preparing Philosophy vs. IT slides**
- **Peter Clark & other Boeing colleagues for reviewing this talk.**

References

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

- **Vocabularies, Taxonomies, Thesauri, Meta-models**

<http://www.metamodel.com/article.php?story=20030115211223271>

<http://www.ontopia.net/topicmaps/materials/tm-vs-thesauri.html>

- **XML Schema and Ontologies**

<http://www.cs.man.ac.uk/~horrocks/Publications/download/2001/etai01.pdf>

- **Introduction to Ontologies with Semantic Integration Focus**

<http://www.sigmod.org/sigmod/record/issues/0412/12.uschold-9.pdf>

References

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

- **Unambiguous Semantics**

- Uschold, M. *Where are the Semantics on the Semantic Web?* AI Magazine, 24(3), pp 25-36. Fall 2003.

- **Glosses & Definitions for Ontology Engineering**

- Jarrar, M. Towards the notion of gloss, and the adoption of linguistic resources in formal ontology engineering. *Proceeding of the 15th International World Wide Web Conference, WWW2006*. Edinburgh, Scotland. May 2006. ACM, 2006
- Michael, J. and Mejino, Jose L V and Rosse, Cornelius (2001) The Role of Definitions in Biomedical Concept Representation. In *Proceedings, American Medical Informatics Association Fall Symposium*, pages pp. 463-467.
<http://sigpubs.biostr.washington.edu/archive/00000148/>

- **Database Lexicography**

- Coen, G. 2002. Database lexicography. *Data & Knowledge Engineering* 42, 3 (Sep. 2002), pp 293-314

Uncle SEM Wants You!

Boeing Technology | Phantom Works

E&IT | Mathematics and Computing Technology

Buy Ontologies.

Loose Ontologies Sink Apps.

Questions?



Artwork by James Montgomery Flagg.