

Models versus Ontologies - What's the Difference and where does it Matter?

Colin Atkinson

University of Mannheim

VORTE2006

Regal Kowloon Hotel

Hong Kong

October 16th 2006.

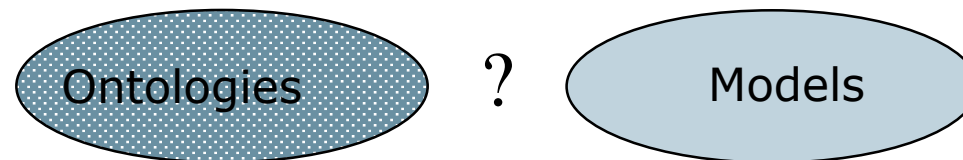
Brief History

– Ontologies

- originated from the artificial intelligence world for the purpose of precisely capturing “knowledge”
- used under an Open World Assumption (OWA)
 - new “knowledge” discovered by automated reasoning
- characterized by OWL as the flagship language
 - formal semantics (description logic)

– Models (à la MDA)

- originated from the software engineering world for the purpose of simplifying the description of software
- used under a Closed World Assumption (CWA)
 - information defined prescriptively for construction
- characterized by UML as the flagship language
 - semi-formal semantics (metamodels)



Conventional Wisdom

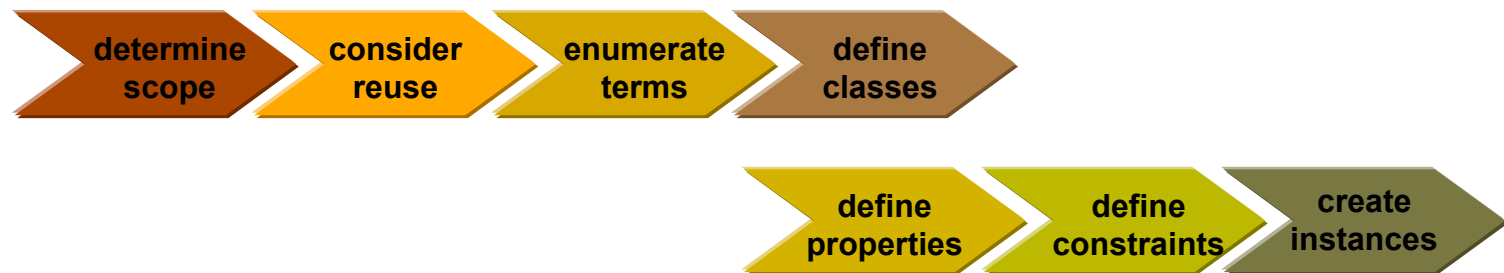
- the conventional wisdom is that the two are distinct
 - ontologies are for some things and models for others
 - you need to make a choice
- there is growing interest in how ontology-based technologies and model-based technologies overlap
 - growing number of related workshops such as SEKE, VORTE, MDSW, SWESE, ONTOSE, WoMM, ...
- numerous proposals for “bridging” or integrating the two technology spaces
 - Ontology-driven Architectures
 - Ontology-based software engineering
 - Ontology Definition Metamodel
 - Model Driven Semantic Web
- all based on the premise that they are inherently different and there are objective distinction criteria

Common Informal Distinctions

- Purpose-based distinctions
 - models focus on realization (ontologies do not)
 - A Guide to Creating your First Ontology [Noy and McGuinness 2001]
 - ontologies are for run-time knowledge exploitation (Models are not)
 - Ontology Driven Architecture [W3C06]
 - ontologies are for representing shared (e.g web based information) (Models are not)
- Property-based Distinctions
 - ontologies are formal (models are not)
 - ontologies can support reasoning (models can not)
 - models use the Close World Assumption (ontologies use the Open World Assumption)

A Guide to Creating your First Ontology (1/2)

- Highly influential paper by Noy and McGuinness describing how to create a Wine Ontology
 - Step 1: Determine the domain and scope of the ontology
 - Step 2: consider reusing existing ontologies
 - Step 3: Enumerate important terms in the ontology
 - Step 4: Define the classes and the class hierarchy
 - Step 5: Define the properties of classes - slots
 - Step 6: Define the facets of the slots
 - Step 7: create instances



A Guide to Creating your First Ontology (2/2)

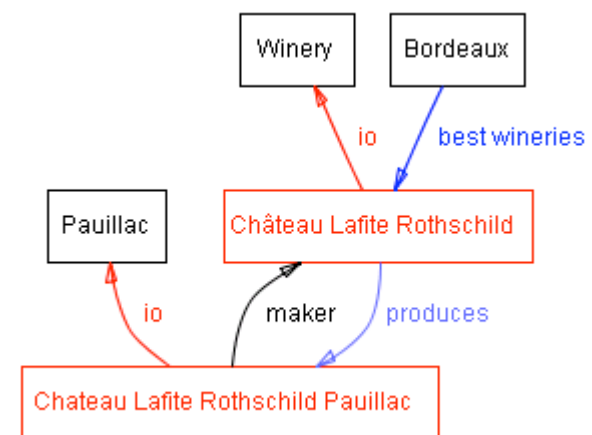
- although the authors acknowledge that some of the ontology design ideas originated from object-oriented **design** (Booch and Rumbaugh) they make the following distinction :

Ontology development

- reflects structure of the world
- about structure of concepts
- actual physical representation is not an issue
- the described process is 100% the same as that for creating a domain (conceptual) class diagram
- the stored information is 100% the same as that which would be stored in a domain class diagram

Object-oriented **programming**

- reflects structure of data and code
- is usually about behavior (methods)
- describes physical representation of data (long int, char, etc.)



Domain/Conceptual Modeling

- “A model of a system is a description or specification of that system and its environment for some certain purpose”

(MDA Guide)

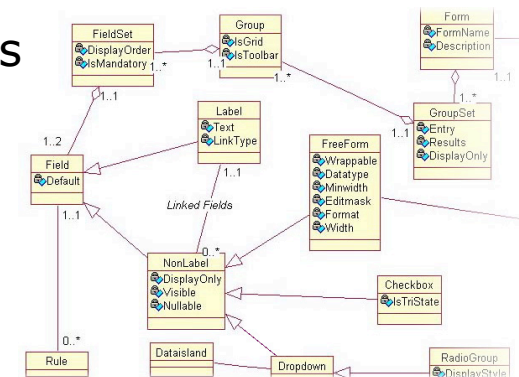
- “A domain model can be thought of as the conceptual model of the system.....The domain model is created to understand the key concept of the system and to familiarize with the vocabulary of the system ”

(Wikipedia)

- A domain model is an explicit description of a domain in terms of:

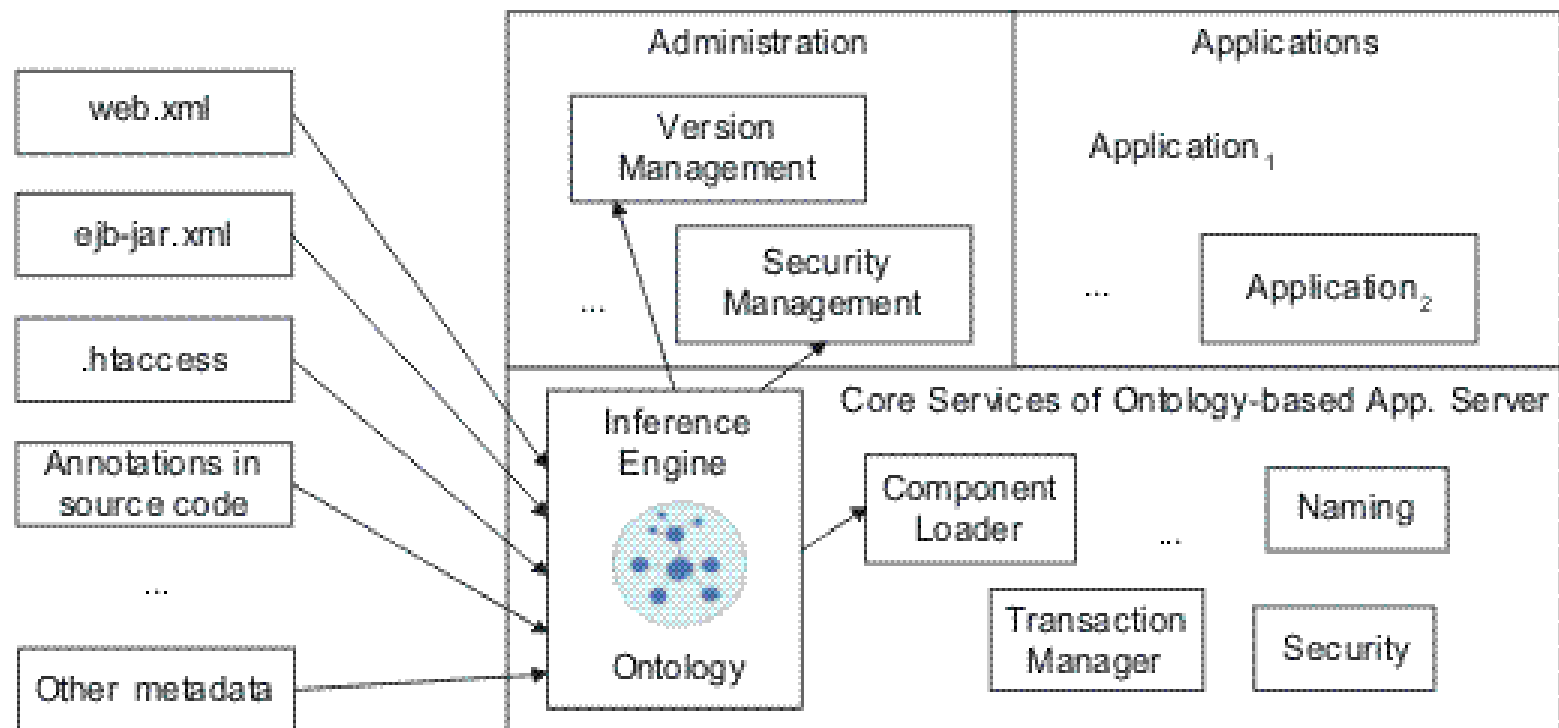
- concepts
- properties and attributes of concepts
- constraints on properties and attributes
- individuals

- A domain model defines
 - a common vocabulary
 - a shared understanding



Ontology Driven Architectures

- Quote from "Ontology Driven Architectures and Potential Uses of the Semantic Web in Systems and Software Engineering" by the W3C

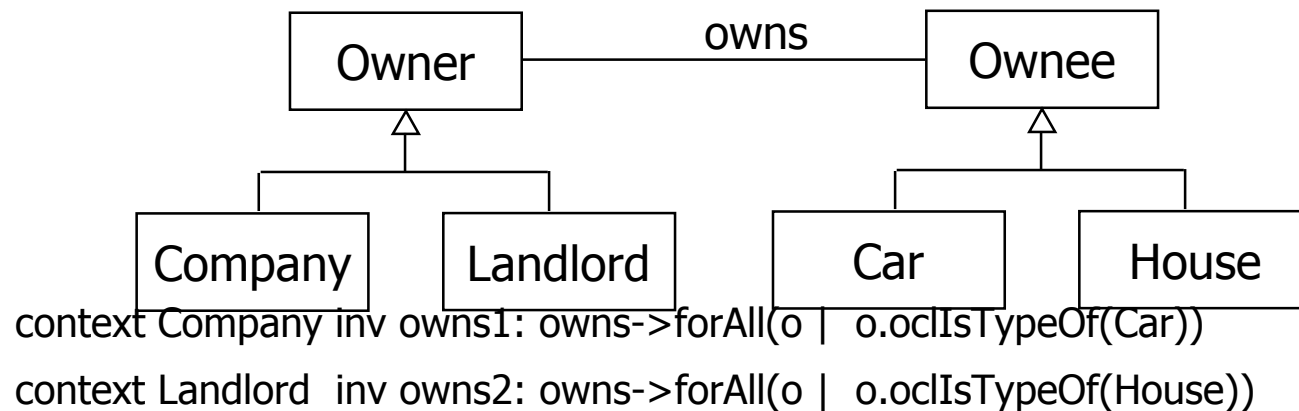


The Role of OCL

- an often quoted shortcoming of the UML for the purpose of ontology representation is the lack of global properties
 - “it is not possible to show that different incarnations of a relationship (e.g. “owns”) are somehow the same”



- but although association inheritance is not directly supported, it can be simulated with the help of OCL



Summary of Current Situation

- no consensus on the relationship between “models” or “modeling” and “ontologies” or “ontology engineering
- old distinctions based on “purpose” becoming obsolete
 - people shift criteria to suite their purpose and artificially maintain the separation
- most uses of UML as a visual notation for ontologies override (ignore) UML’s default semantics
 - metaphore overload
- why it matters
 - general confusion, arbitrariness and ubiquitous use of semantics-free statements
 - very early choice between OW and CW technologies
 - developers locked to one technology due to migration barrier
 - unnecessary weaknesses in both technologies
 - second order notions (OWL-S, CCM, UML Components ..)

Fundamental Definitions

Ontologies

DM1. "An ontology is an explicit (or implicit) mapping of a subject matter into a system of symbols."

Herbert Studer et al.

DO2. "An ontology is a formal, explicit specification of a shared conceptualization. **Conceptualization** refers to an abstract model

DM2. "A model of a system is a description or specification of that system and its environment for **Explicit** means that the type of concepts used, and the constraints on their use are explicitly defined. Formal refers to the fact that the ontology should be **machine-readable**. **Shared** reflects the notion that an ontology captures consensual knowledge, that is, it is not private to some individual, but accepted by a group."

MDA Guide
Studer et. al.

DO3. "[...] an ontology refers to an engineering artefact, constituted by a specific vocabulary used to describe a certain reality, plus a set of explicit assumptions regarding the intended meaning of the vocabulary. Usually a form of **first-order-logic** theory is used to represent these assumptions, vocabulary appears as unary and binary predicates, called concepts and relations, respectively."

Maedche

Observations

- O5. When information is presented in IRs, DMs, OWLs, or UMLs, the core definitions DO1, DO2, and DO3 are not being used, but they also help in defining requirements DM1 and DM2 for the definition of a model.
- O6. IRs in OWL that have no shared understanding contradict DO2.
- O2. Most of the requirements presented characterize IRs in OWL that are not mentioned in the core definitions.
- support for reasoning and intelligent databases are not therefore required for conformance to DO1, DO2 and DO3.
- O3. Of the informal property-oriented distinctions only formality is mentioned in the core definition DO3.
- no requirement for OWA or CWA for either models or ontologies.
- O4. There is no mention of the intended scope of a model in either of the core definitions DM1 and DM2 (i.e. there is no reference to whether they are intended for representing shared information).
- however, there is no requirement that models are restricted to the representation of private (i.e. non shared) data.

Ontology Set Membership Criteria

- the set of ontologies should be viewed as a subset of the set of models
 - not particularly controversial
- BUT, when does a model qualify as an ontology?
 - what is the ontology set membership criteria?
 - when should one refer to a model as an ontology?
- possessing the following properties is necessary, but is it sufficient?
 - conceptualization, explicit, machine readable, based on first order logic, shared
- if we say yes, then “domain models” or “computation independent models” qualify as ontologies

Proposal

- _ the term ontology be saved for models which satisfy these five criteria and are intended to be of “universal” scope
 - _ machine readable form of standard reference books
- _ Qualifying models
 - _ general ontologies
 - _ Mereology Ontology
 - _ upper-level” or “top-level ontologies
 - _ Suggested Upper Merged Ontology
- _ Non-qualifying models
 - _ limited scope models (e.g. SE domain models)
 - _ informal models
 - _ many taxonomies
- _ Metamodels are common examples of ontologies?

Choices Today

- When to use an “ontology technology” (Protégé, OWL)
 - if you need support for automated reasoning (classification and subset recognition)
 - if you want to share your information on the web
- When to use a “modeling technology” (UML Tool, UML)
 - if you want an intuitive and expressive graphical notation (but don’t need reasoning support beyond inheritance)
 - if you want to construct artifacts using MDA techniques (including database tables)
- if you want UML / MDA features and support for reasoning you have a problem
- in general, these choices are overridden by more pragmatic concerns
 - employee experience/training
 - upwardly compatibility
 - marketing

Vision (or Hope)

- unified information representation technology which supports
 - same graphical syntax for common object-oriented representation concepts (classes, relationships ..)
 - most general default assumptions
 - ability to switch between OW/CW assumptions
 - expressed as a general constraint at the appropriate meta level
 - maximum possible degree of reasoning
- consensus on when to label an IR as an ontology and as one of the different kinds of model
 - Domain/conceptual models versus ontologies?
 - move away from current arbitrariness and semantics-free use of terminology