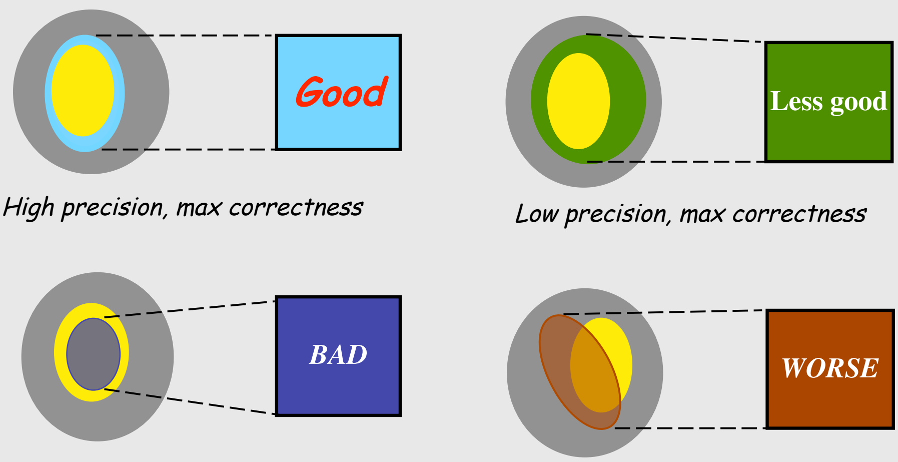
The Ontological Level

Ontology Quality: Precision and Correctness



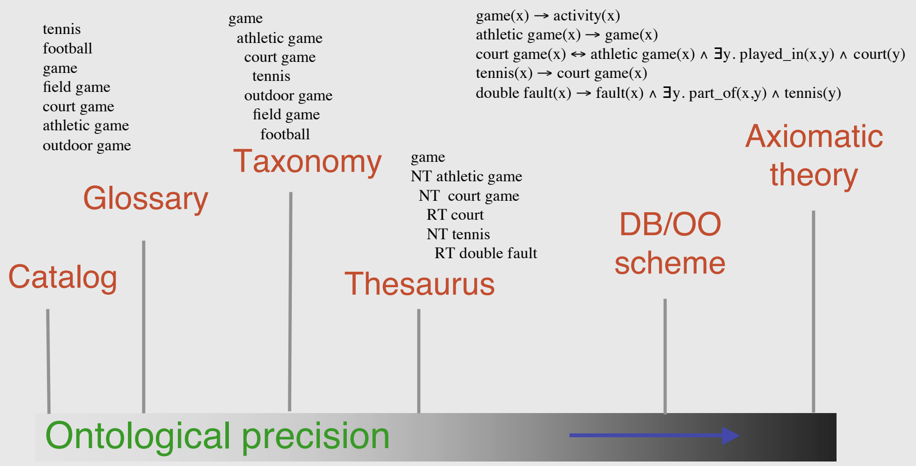
High precision, max correctness

Low precision, max correctness

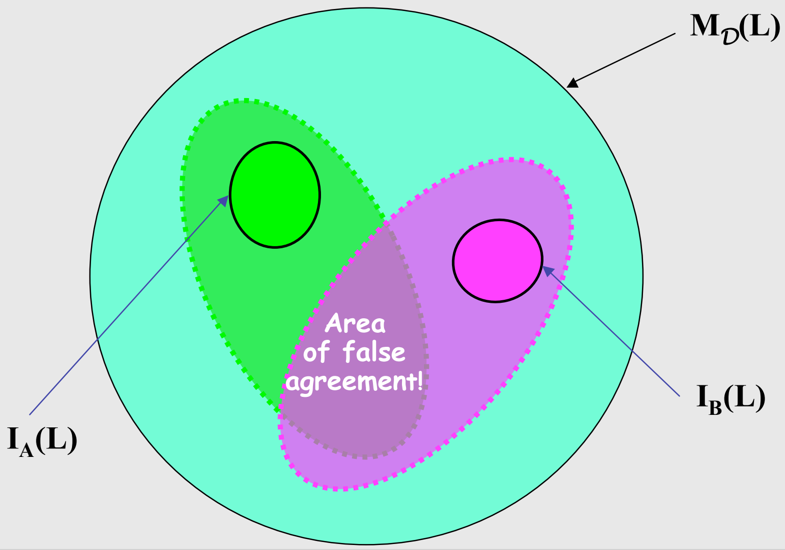
Max precision, low correctness

Low precision, low correctness

Levels of Ontological Precision



Why precision is important



When precision is not enough

Only one binary predicate in the language: on

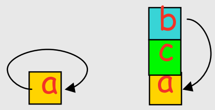
Only three blocks in the domain: a, b, c.

Axioms (for all x,y,z):

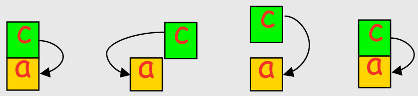
$$on(x,y) \rightarrow \neg on(y,x)$$

$$on(x,y) \rightarrow \neg \exists z\, (on(x,z) \wedge on(z,y))$$

Non-intended *models* are excluded, but the rules for the competent usage of on in different *situations* are not captured.



Excluded conceptualizations



Indistinguishable conceptualizations

The reasons for ontology inaccuracy

- In general, a single intended ***model*** may not discriminate between positive and negative ***examples*** because of a ***mismatch*** between:
  - Cognitive domain and domain of discourse: lack of ***entities***
  - Conceptual relations and ontology relations: lack of ***primitives***
- Capturing all intended models is not sufficient for a "perfect" ontology

  Precision: non-intended *models* are excluded

  Accuracy: negative *examples* are excluded

When is a precise and accurate ontology useful?

1. When *subtle distinctions* are important
2. When *recognizing disagreement* is important
3. When *general abstractions* are important
4. When *careful explanation and justification* of ontological commitment is important
5. When ***mutual understanding*** is more important than interoperability.

Kinds of ontology change
(to be suitably encoded in versioning systems!)

- Reality changes
  - Observed phenomena
- Perception system changes
  - Observed qualities (different qualia)
  - Space/time granularity
  - Quality space granularity
- Conceptualization changes
  - Changes in cognitive domain
  - Changes in conceptual relations
    - metaproperties like rigidity contribute to characterize them (OntoClean assumptions reflect a particular conceptualization)
- Logical characterization changes
  - Domain
  - Vocabulary
  - Axiomatization (Correctness and Precision)
  - Accuracy

A quantitative metric for ontology correctness and precision

- Assumption: finite **D**, finite **W** *(examples)*

- Correctness = $card(I_k \cap O_k)/card(I_k)$
- Precision = $card(I_k \cap O_k)/card(O_k)$

Measuring ontological accuracy (wrt benchmark examples)

- *Anomalous intended models* (set A_k): those that collapse intended and non-intended situations

$$\text{Accuracy} = (card(I_k)-card(A_k))/card(I_k)$$

Ontologies vs. classifications

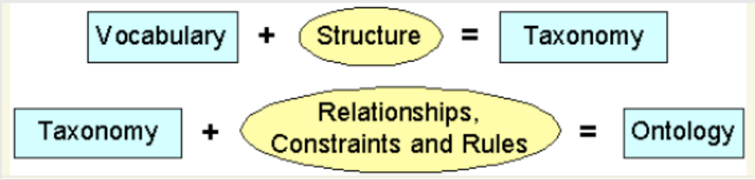
- Classifications focus on:
  - ***access***, based on pre-determined criteria (encoded by ***syntactic keys***)
- Ontologies focus on:
  - ***Meaning*** of terms
  - ***Nature*** and ***structure*** of a domain

Ontologies vs. Knowledge Bases

- Knowledge base
  - Assertional component
    - **reflects *specific (epistemic) states of affairs***
    - designed for ***problem-solving***
  - Terminological component (*ontology*)
    - ***independent*** of particular ***states of affairs***
    - Designed to support ***terminological services***

Ontological formulas are (assumed to be)
invariant, necessary information

Ontologies and taxonomies



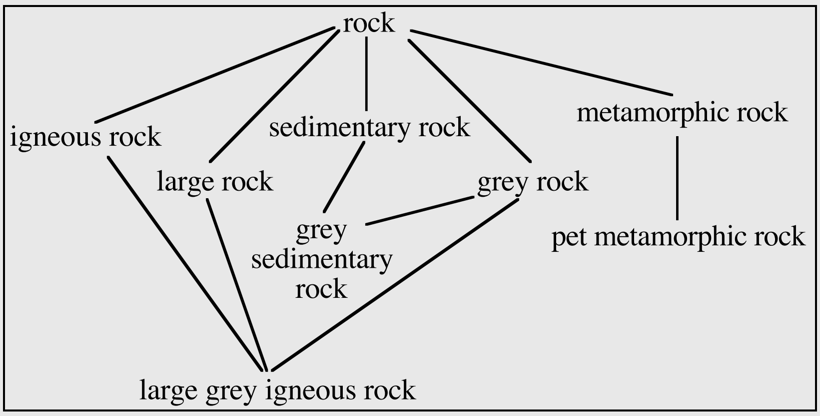
Ontologies vs. Database Schemas

- Database schemas:
  - Constraints focus on ***data integrity***
  - Relationships and attribute values out of the DoD
  - Typically ***non-executable***
- Ontologies:
  - Constraints focus on ***intended meaning***
  - Relationships and attribute values first class citizens
  - Typically ***executable***

The Ontological Level

Kinds, roles, attributions

How many rock kinds are there?



[From Brachman, R ., R. F ikes, et al. 1983. "Krypton: A Functional Approach to Knowledge Representation", *IEEE Computer*]

The answer

- According to Brachman & Fikes 83:
  - It's a dangerous question, only "safe" queries about analytical relationships between terms should be asked
- In a previous paper by Brachman and Levesque on ***terminological competence*** *in knowledge representation* [AAAI 82]:
  - "an *enhancement mode transistor* (which is a *kind* of transistor) should be understood as different from a *pass transistor* (which is a *role* a transistor plays in a larger circuit)"
- These issues have been simply *given up* while striving for logical simplification and computational tractability
- The OntoClean methodology, based on formal ontological analysis, allows us to conclude: ***there are 3 kinds of rocks*** (appearing in the figure)

From the logical level to the ontological level

- Logical level *(no structure, no constrained meaning)*
  - $\exists x$ (Apple(x) $\wedge$ Red(x))
- Epistemological level *(structure, no constrained meaning)*:
  - $\exists x$*:apple* Red(x) *(many-sorted logics)*
  - ~~$\exists x$*:red* Apple(x)~~
  - a is a Apple with Color=red *(description logics)*
  - a is ~~a Red with Sha~~pe=apple
- Ontological level *(structure, constrained meaning)*
  - Some structuring choices are excluded because of ontological constraints: Apple carries an identiy condition, Red does not.

Ontology helps building "meaningful" representations

The source of all problems: (slightly) different meanings for words

- A (simple-minded) painter may intepret the words "Apple" and "Red" in a completely different way:
  - Three different reds on my palette: Orange, Appple, Cherry
- So an expression like $\exists x$:*red* Apple(x) may mean that there is an "Apple" red.
- Two different ontological assumptions behind the Red predicate:
  - adjectival interpretation: *being a red thing* doesn't carry an identity criterion (uncountable)
  - nominal interpretation: *being a red color* does carry an identity criterion (countable)

Formal ontological distinctions help making *intended meaning* explicit

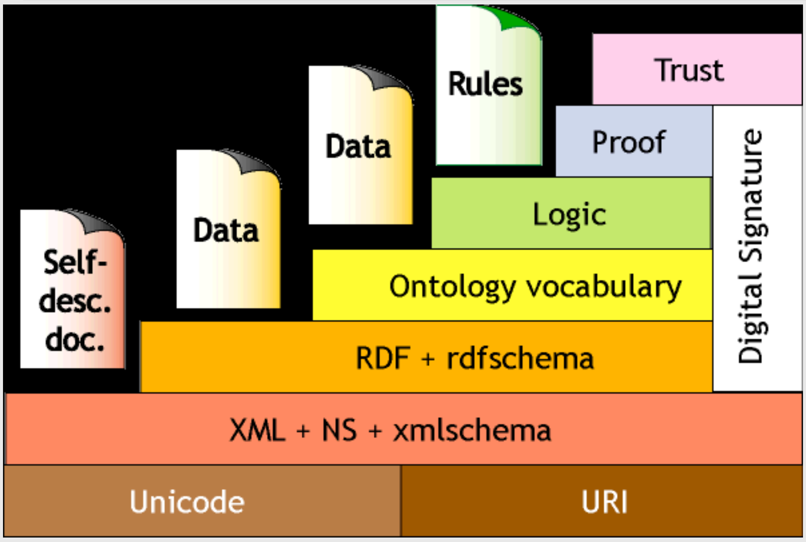
Ontological analysis can be defined as the process of ***eliciting and discovering relevant distinctions*** and relationships bound to the very nature of the entities involved in a certain domain, ***for the practical purpose of disambiguating terms*** having different interpretations in different contexts.

The Ontological Level

(Guarino 94)

Level	*Primitives*	*Interpretation*	*Main feature*
Logical	Predicates, functions	Arbitrary	Formalization
Epistemological	Structuring relations	Arbitrary	Structure
Ontological	**Ontological relations**	**Constrained** (meaning postulate s)	**Meaning**
Conceptual	Conceptual relations	Subjective	Conceptualization
Linguistic	Linguistic terms	Subjective	Language dependence

The semantic web architecture [Tim Berners Lee 2000]



Ontology-driven information systems

Ontology-Driven Information Systems

- Every IS ***has*** its own ontology (either implicit or explicit)
- The ODIS perspective: ***explicit*** ontologies play a ***central*** role, driving ***all*** aspects and components of an IS
- Two (main) dimensions to assess the role of an explicit ontology:
  - ***temporal dimension****:* development time vs. run time
  - ***structural dimension:*** impact on the various IS components:
    - ***database component***
    - ***application program***
    - ***user interface***

Temporal dimension: *development time*

- Two scenarios:
  - A pre-existing ***ontology library*** containing domain and task ontologies as “main building blocks” to be adapted and rused
    - standard IS: the ontology content is *embedded* in the standard components
    - ODIS: an *application ontology* is built by specializing domain and task ontologies taken from the library
  - Only an ***upper-level ontology*** available: not building blocks, but ***conceptual tools*** (analogous to other CASE tools)
- Two kinds of development:
  - IS ***engineering***
  - IS ***re-engineering***

Temporal dimension: *run time*

- Ontology-***aware*** IS: the IS just uses the ontology for some specific purpose
- Ontology-***driven*** IS: the ontology is a ***central component*** of the IS, cooperating at run time towards its "higher" overall goal
- Important application: ***inter-agent communication***

Structural dimension: *the database component*

- Development time:
  - support to ***requirement analysis and conceptual modelling*** (integrated with lexical resources like WordNet)
  - development of a ***global conceptual schema*** (DB integration)
- Run time:
  - mediation-based approach to ***information integration***
  - ***intensional queries***

Structural dimension:
the user-interface component

- Development time:
  - Generation of ***form-based interfaces*** (constraints checking)
- Run time:
  - Support ***quering and browsing the ontology*** itself:
    - better understanding of the vocabulary
    - queries at the desired level of specificity
  - ***Vocabulary detaching***:
    - user free to adopt his own NL terms (mapped - after disambiguation - to the IS vocabulary with the help of the ontology)

Structural dimension:
the application program component

- Development time:
  - Generation of the static part of a program (type structure)
  - Support to OO design
- Run time:
  - Explicit account of the ***ontological commitment*** of an application program
  - Increase of the ***transparency*** of application software

A single, imperialistic ontology?

- An ontology is first of all ***for understanding each other***
  - ...among people, first of all!
  - not necessarily for thinking in the same way
- A single ontology for multiple applications ***is not necessary***
  - Different applications using different ontologies can co-exist and co-operate (not necessarily inter-operate)
  - ...if linked (and compared) together **by means of a general enough basic categories and relations (*primitives*)**.
- If basic assumptions are not made explicit, any imposed, common ontology risks to be
  - seriously mis-used or misunderstood
  - opaque with respect to other ontologies