ONTOLOGIES

A short tutorial with references to YAGO

Cosmina CROITORU

Unlocking the Secrets of the Past: Text Mining for Historical Documents

Blockseminar, 21.2.-11.3.2011

Presentation's structure

- Ontologies
- Yago Generalities.
- Yago Model.
- Yago Formal (Semantics).
- Yago Query.
- Yago Construction.
- Yago Outlook.

What is an Ontology? (1)

- An ontology is a formal explicit description of
  - concepts in a domain of discourse (classes),
  - properties of each concept describing features and attributes of the concept (slots, roles or properties),
  - restrictions on slots (facets or role restrictions).
- An ontology together with a set of individual instances of classes constitutes a knowledge base.

There is a fine line where the ontology ends and the knowledge base begins.

What is an Ontology? (2)

- Classes describe concepts in the domain.
- A class can have subclasses that represent concepts that are more specific than the superclass.
- Slots describe properties of classes and instances.
- In practical terms, developing an ontology includes:
  - defining classes in the ontology,
  - arranging the classes in a taxonomic hierarchy,
  - defining slots and describing their allowed values,
  - filling in the values for slots for instances.

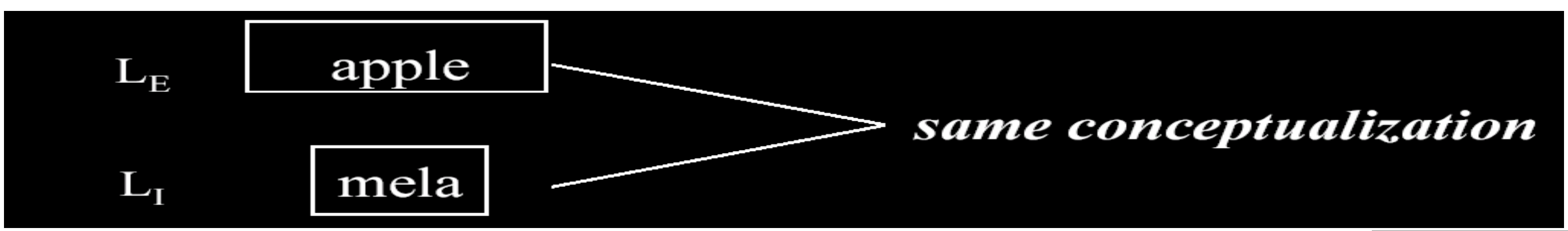
What is an Ontology? (3)

- A specific **artifact** designed with the purpose of expressing the **intended meaning** of a (shared) **vocabulary.**
- A shared vocabulary plus a specification (**characterization**) of its intended meaning.
  An ontology is a specification of a conceptualization**,** [Gruber 95].

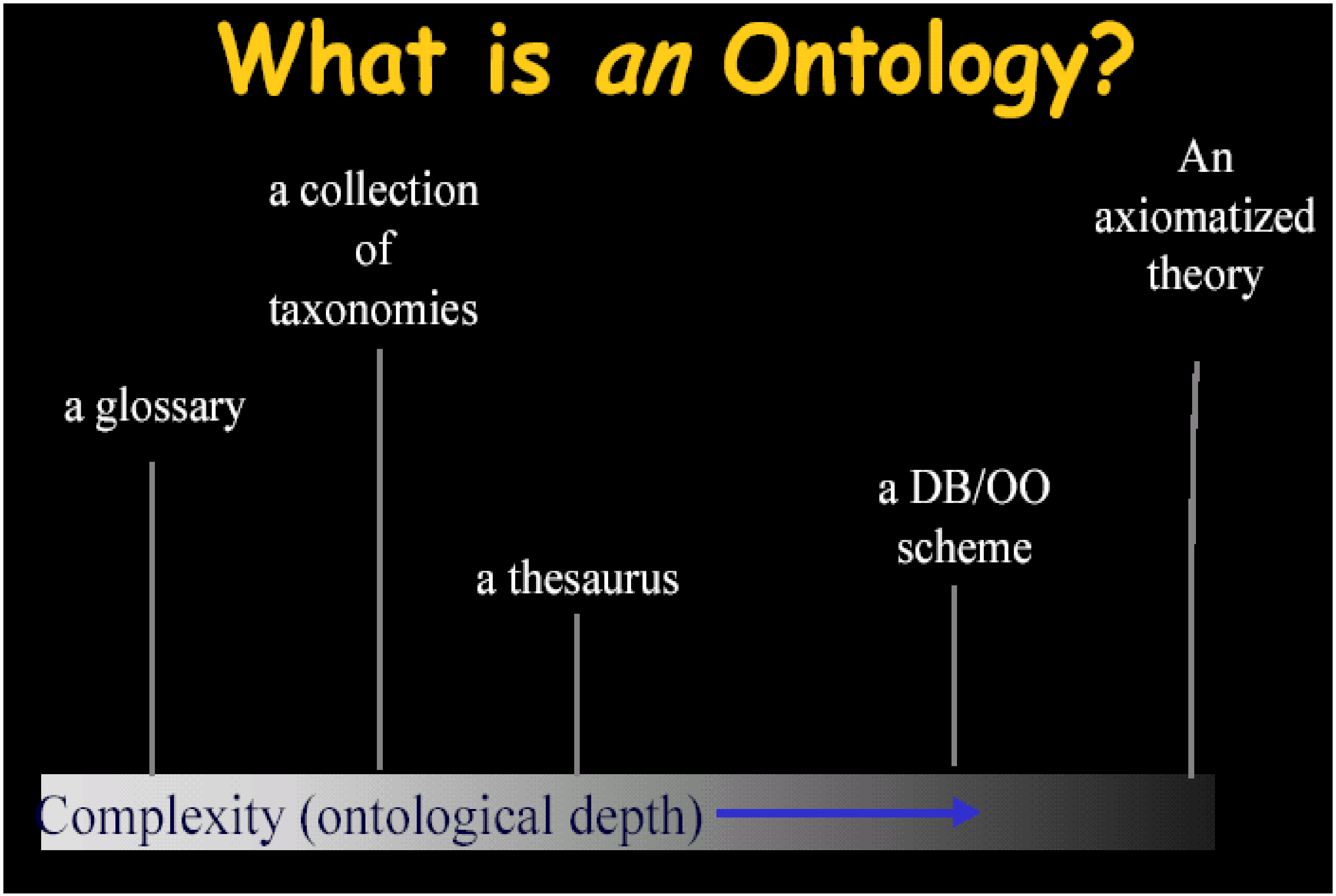
...i.e., an ontology accounts for the **commitment** of a language to a certain **conceptualization.**

What is a conceptualization?

- Formal structure of (a piece of) reality as perceived and organized by an agent, independently of:
  - the **vocabulary** used
  - the actual occurence of a specific **situation.**
- Different situations involving same objects, described by different vocabularies, may share the same conceptualization.



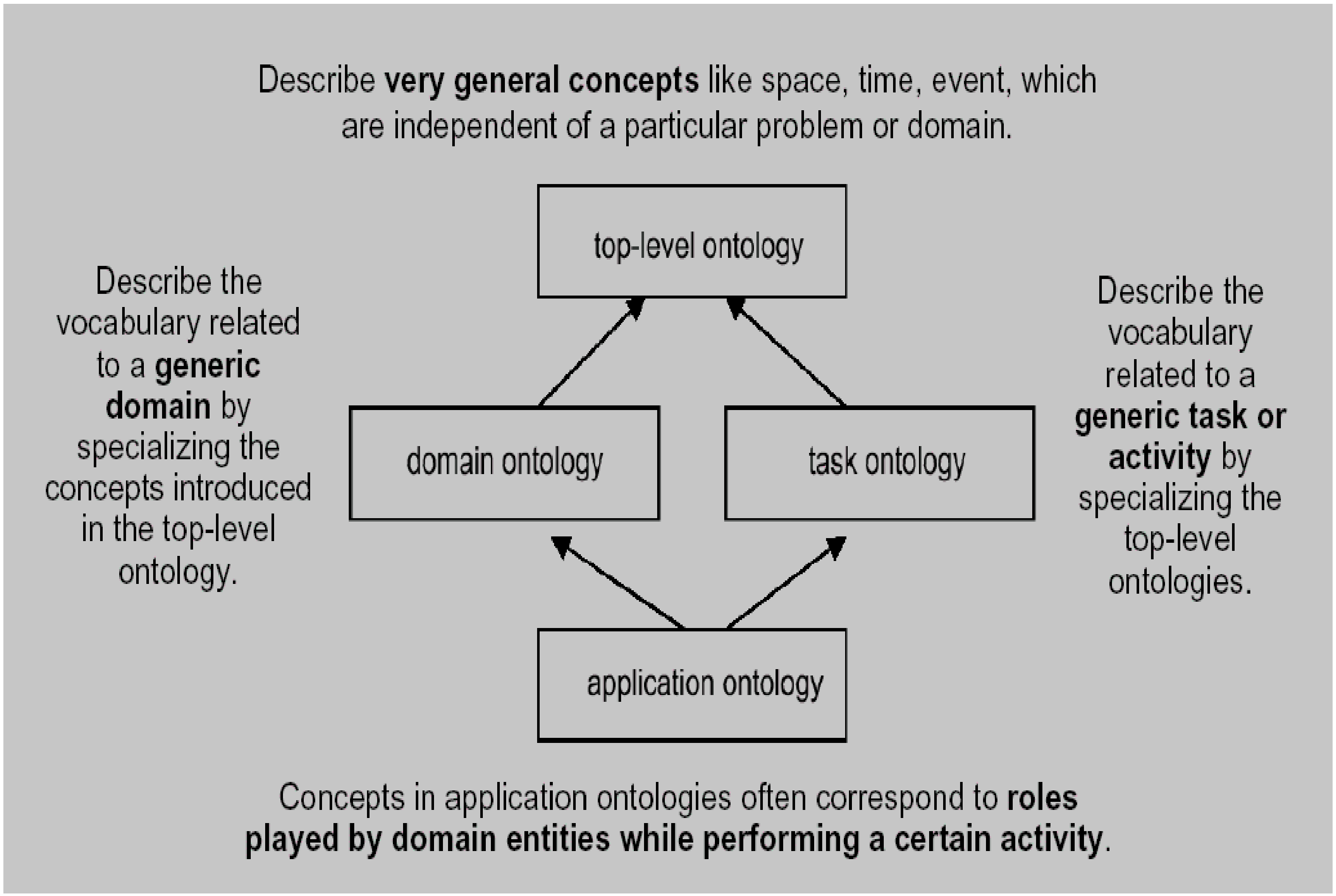
Ontologies and their relatives



Why ontologies?

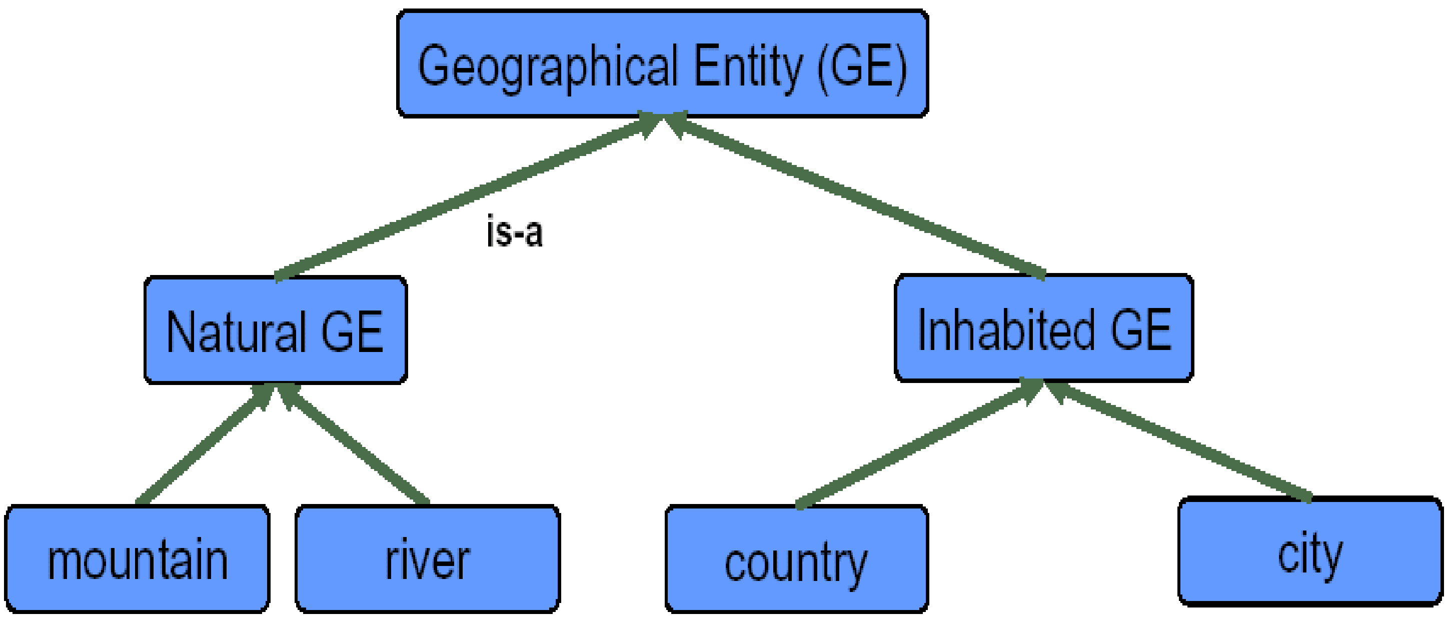
- Semantic Interoperability
  - Generalized database integration
  - Virtual Enterprises
  - e-commerce
- Information Retrieval
  - Decoupling user vocabulary from data vocabulary
  - Query answering over document sets
  - Natural Language Processing

Types of Ontologies [Guarino 98]

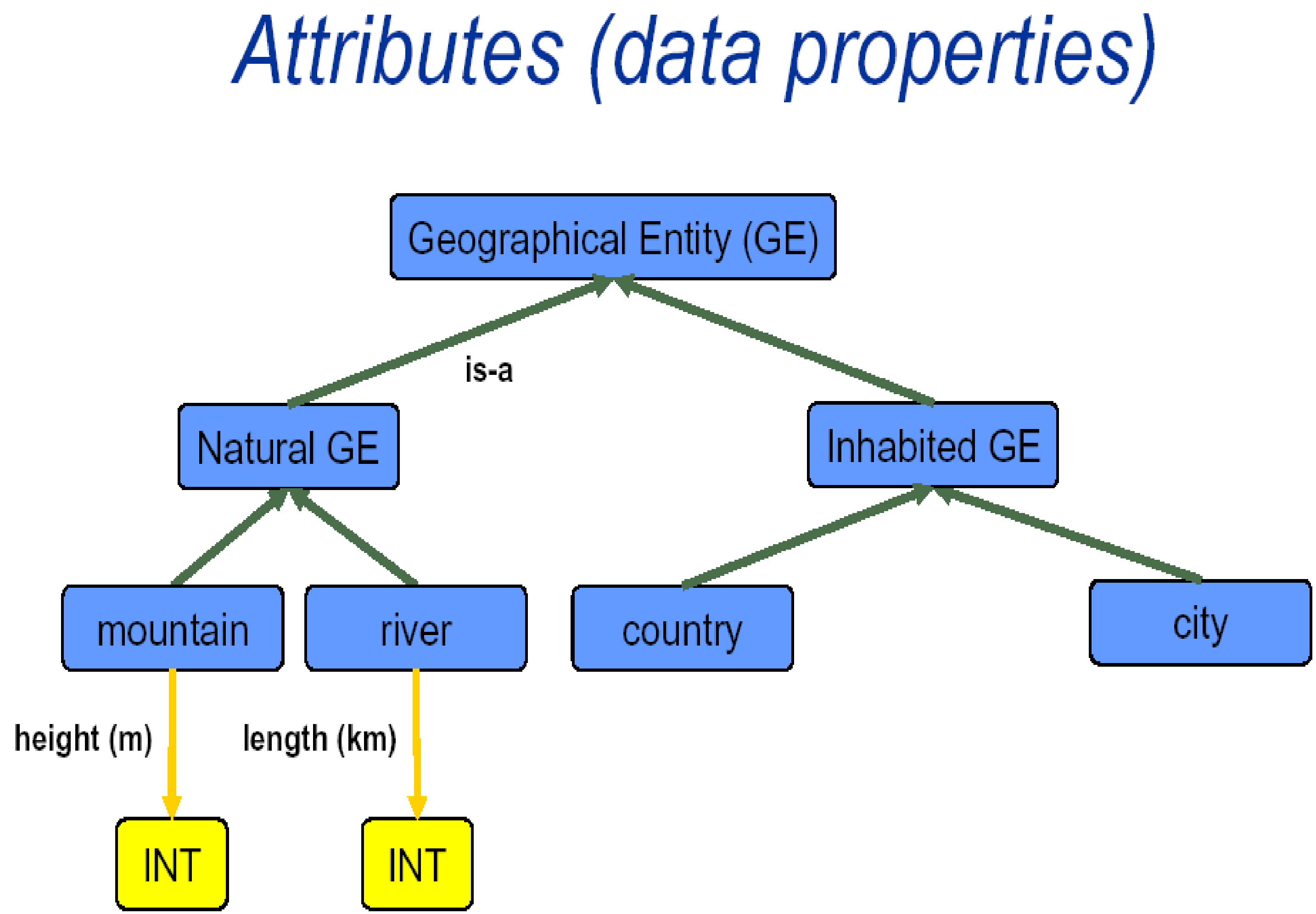


Ontology Example (1)

Classes & Taxonomy

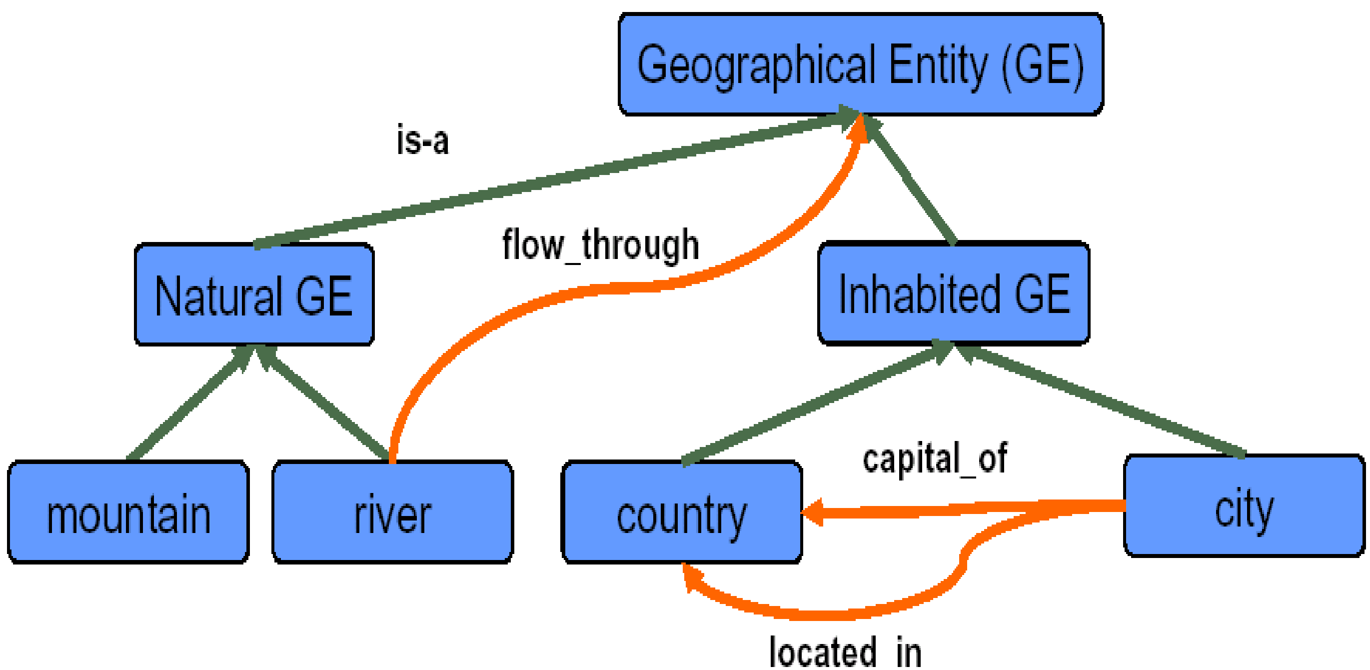


Ontology Example (2)

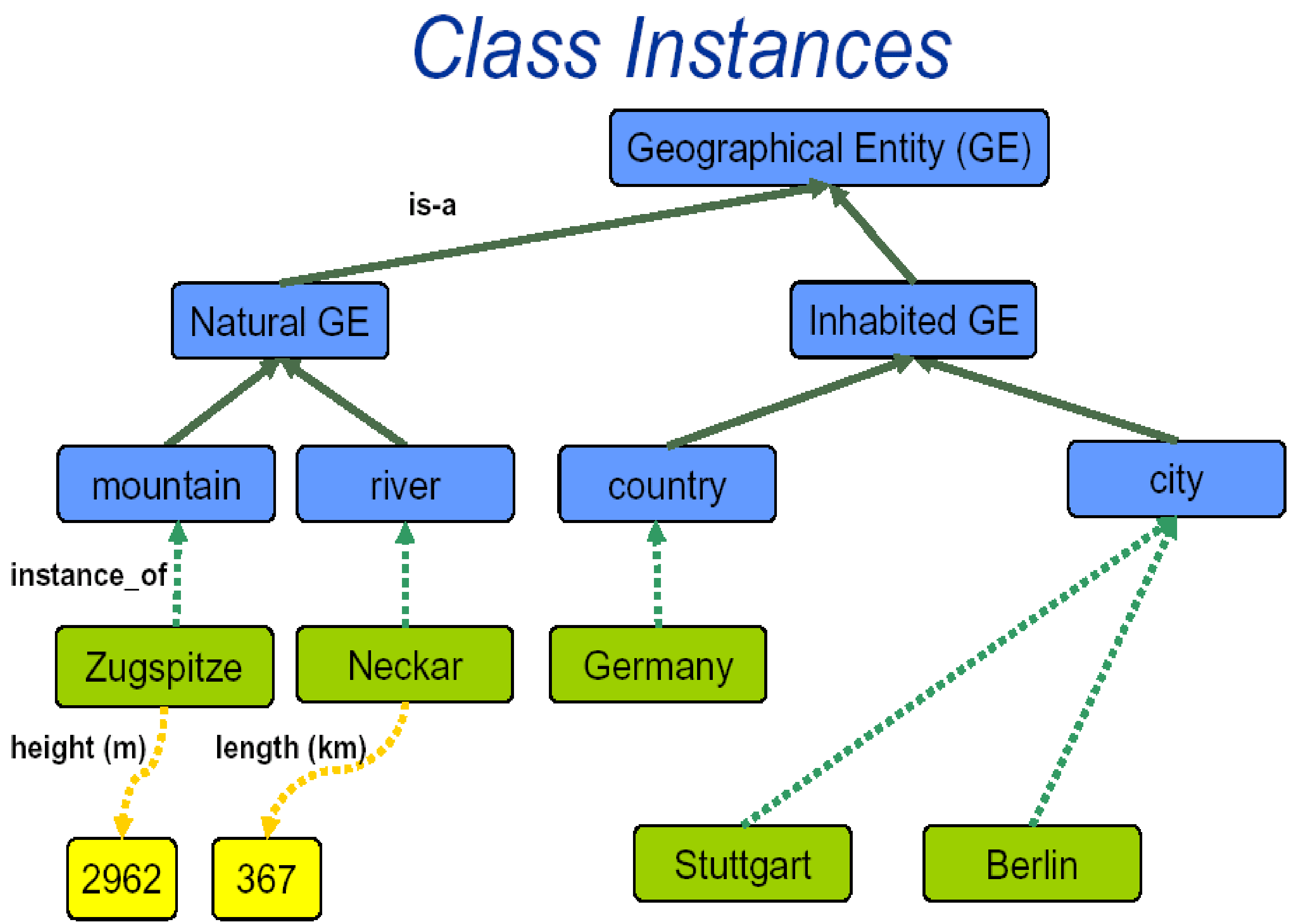


Ontology Example (3)

Relations (object properties)

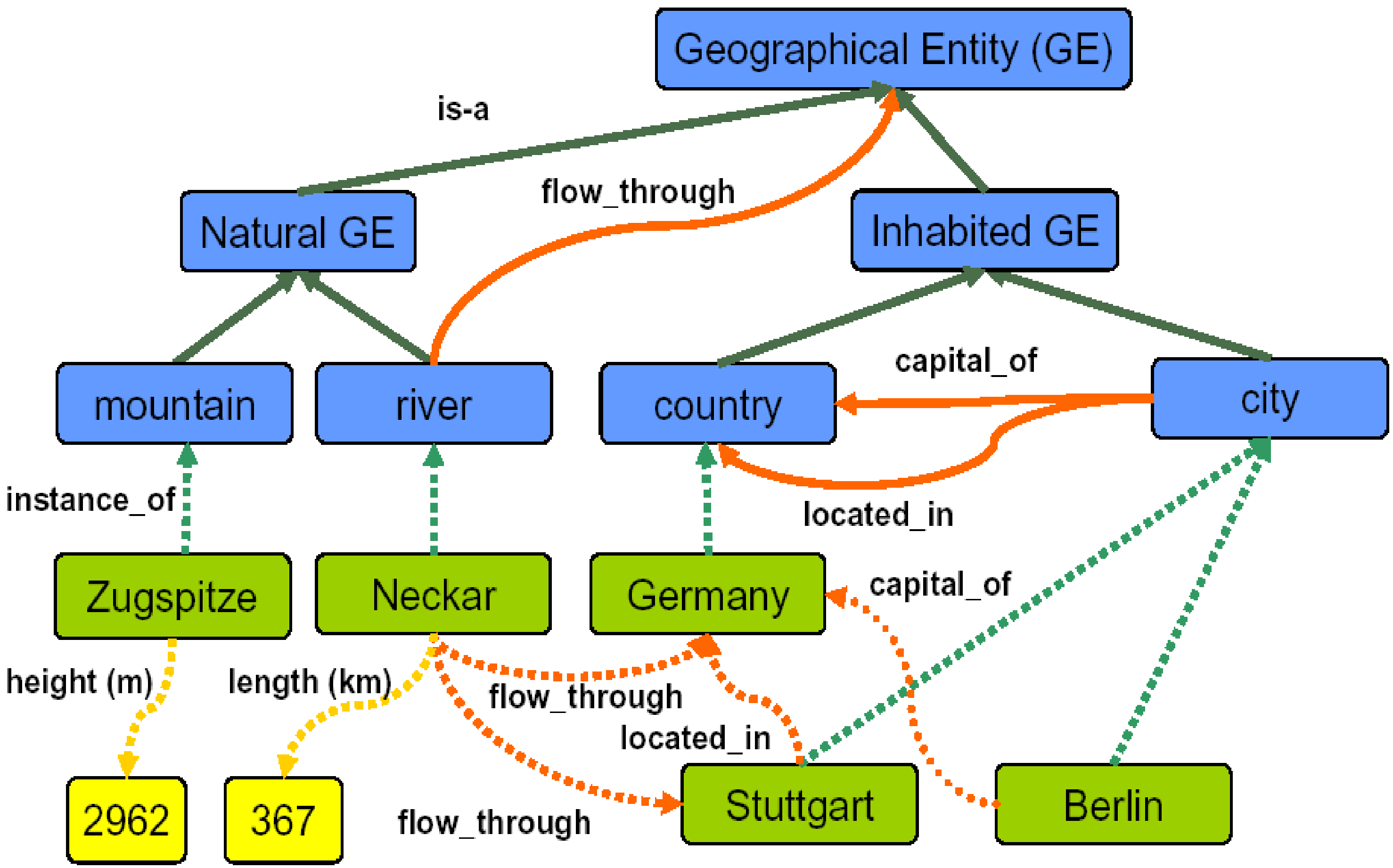


Ontology & Knowledge base Example (4)



Ontology & Knowledge base Example (5)

Relation Instances



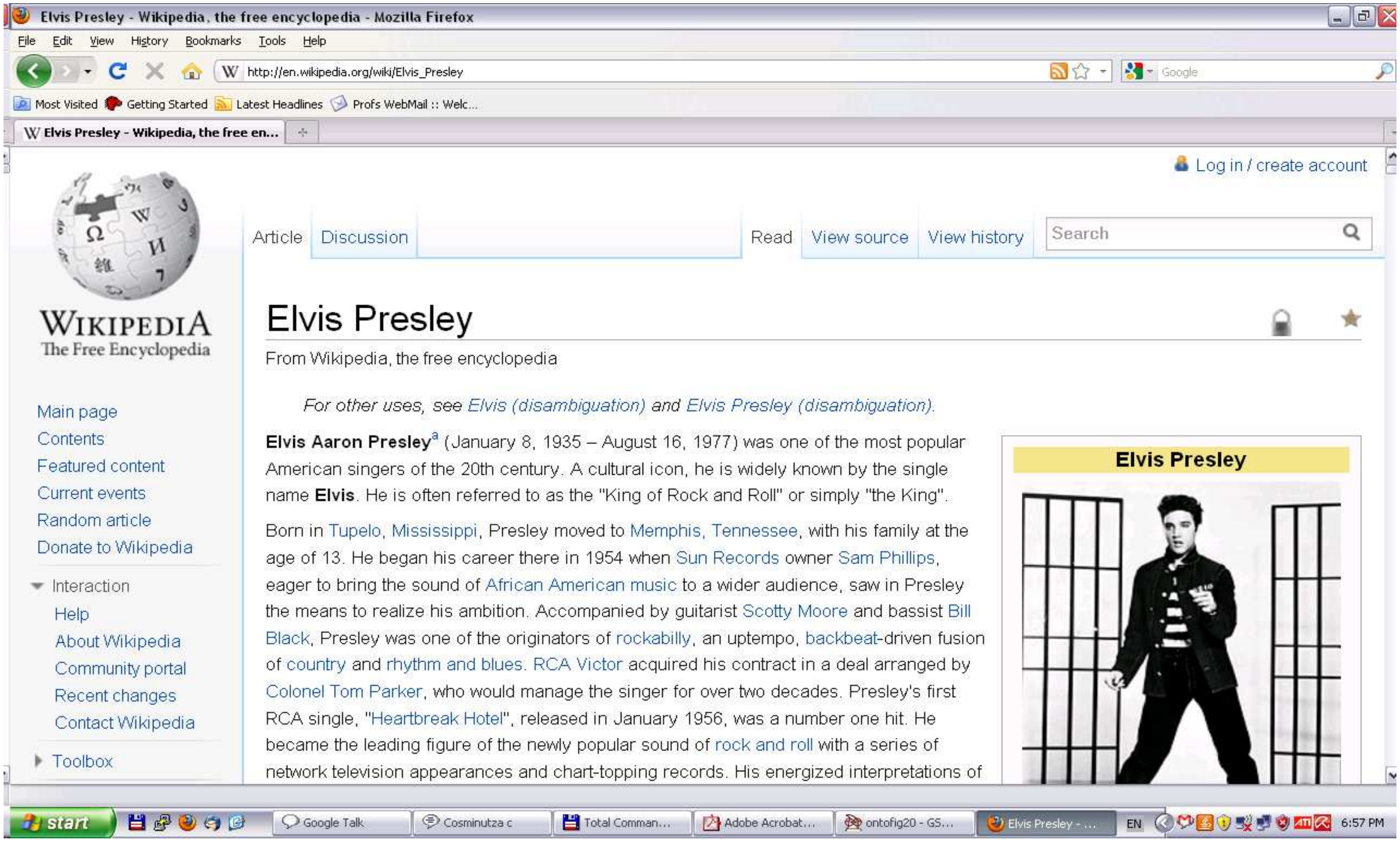
YAGO - generalities

- **Y**et **A**nother **G**reat **O**ntology: http://www.mpi-inf.mpg.de/yago-naga/
- Automatically extracted from Wikipedia; uses WordNet to structure information.
- More than 2 million *entities* (e.g. persons, cities, organizations) and 20 million *facts* about these entities.
- Builds on Wikipedia's *infoboxes* and *category pages*.
- WordNet: NLP professional *hierarchy of concepts*.
- Unification between WordNet and facts derived from Wikipedia with a *precision* of 95%.

YAGO - Using Wikipedia

- Infoboxes: tables containing basic information about the entity described in the article.
  - E.g., infoboxes for countries: name of the country, capital, size.
- Category pages: lists of articles that belong to a specific category.
  - E.g., "Elvis" is in the category of "American rock singers".
- These lists give candidates for entities, concepts, and relations.
  - E.g.: Elvis, IsA(Elvis, rockSinger), nationality(Elvis, American).

Wikipedia article



Wikipedia page about Elvis Presley

Wikipedia Infoboxes

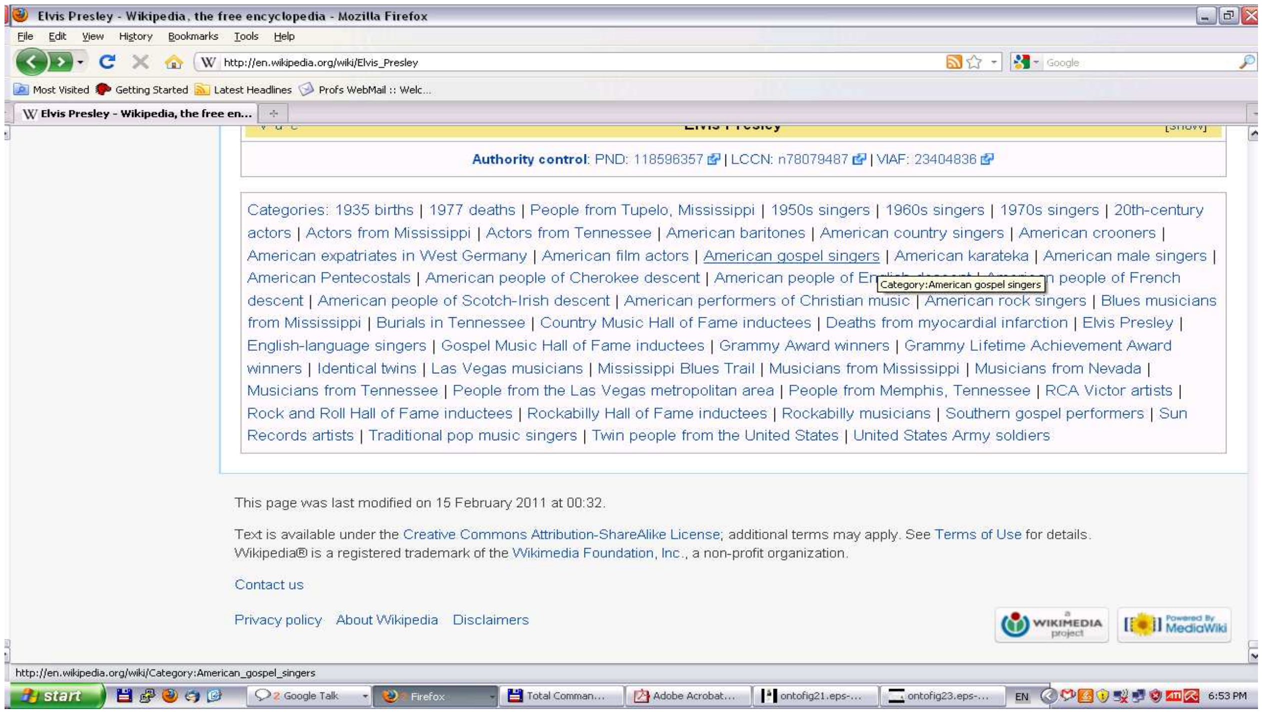
Elvis Presley

Elvis in 1970

Background information	
Birth name	Elvis Aaron Presley[1]
Also known as	Elvis
Born	January 8, 1935 Tupelo, Mississippi
Origin	Memphis, Tennessee
Died	August 16, 1977 (aged 42) Memphis, Tennessee
Genre(s)	Rockabilly, Rock and Roll, Gospel, Blues, Country
Occupation(s)	Singer, Actor
Instrument(s)	Vocals, Guitar, Piano
Years active	1954–1977
Label(s)	Sun, RCA
Website	Elvis.com

There is standardized infobox for people, which contains the birth date, the profession, and the nationality, etc.

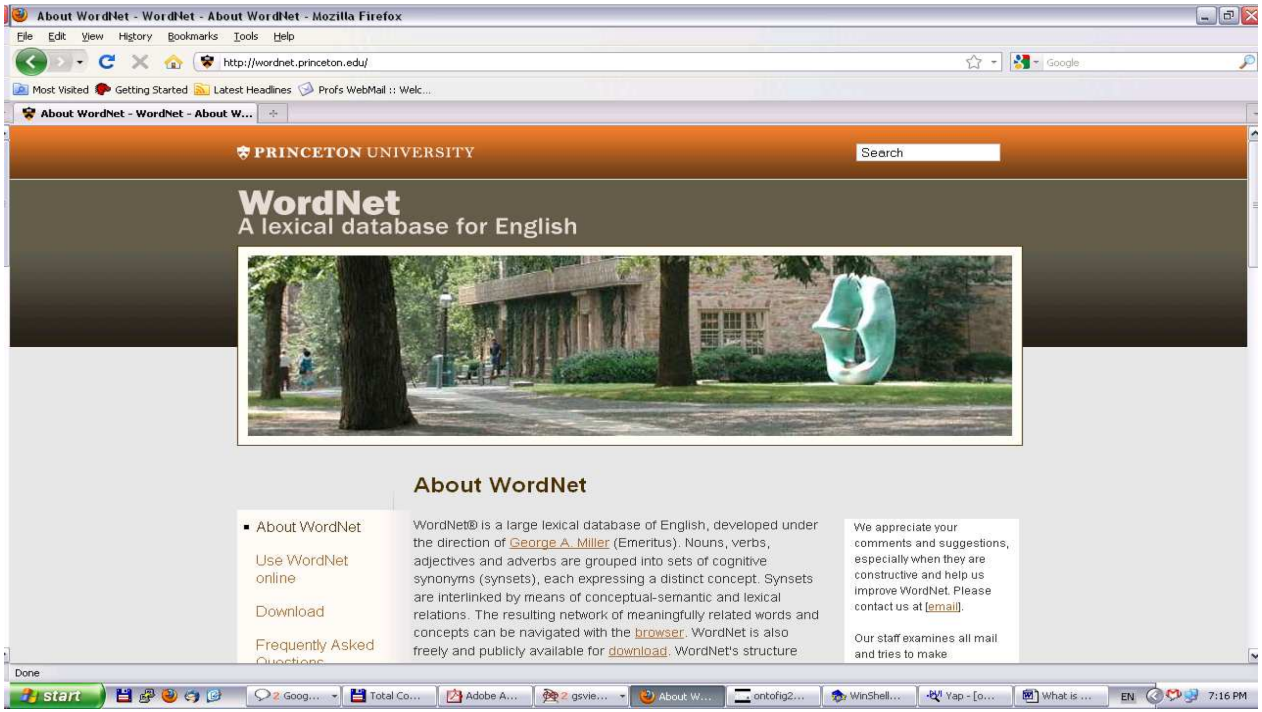
Wikipedia Categories



The majority of Wikipedia pages have been manually assigned to one or multiple categories.

The page about Elvis Presley is in the categories American rock singers, 1935 births, and 34 more.

WordNet (1)



Semantic lexicon for the English language developed at the Cognitive Science Laboratory of Princeton University. WordNet distinguishes between words as literally appearing in texts and the actual senses of the words.

WordNet (2)

- Synset - set of words that share one sense (semantic concept).
- Words with multiple meanings (ambiguous words) belong to multiple synsets.
- Version 3.0 contains 82,115 synsets for 117,798 unique nouns.
- Wordnet also includes other types of words like verbs and adjectives.
- WordNet provides relations between synsets such as hypernymy/hyponymy and holonymy/meronymy.

YAGO - Model (1)

- OWL, Web Ontology Language, state-of-the-art formalism in knowledge representation.
- OWL is based on RDFS, can express relations between facts, but provides only very primitive semantics.
  - E.g., it does not know transitivity, which is crucial for partial orders such as SUBCLASSOF or LOCATEDIN.
- YAGO model, slight extension of RDFS, can express relations between facts and relations, while being at the same time simple and decidable.

YAGO - Model (2)

- The same knowledge representation as RDFS:
- All objects (e.g. cities, people, even URLs) are represented as entities. Two entities can stand in a relation.

Fact: Elvis Presley (entity) HASWONPRIZE (relation) Grammy Award (entity)

- Numbers, dates, strings and other literals are entities. We can write: Elvis Presley BORNINYEAR 1935
- Entities are abstract ontological objects. Language uses words to refer to these entities. Words are entities as well. Expressing that a certain word (quotes!) to refers to a certain entity: "Elvis" MEANS Elvis Presley
- This allows us to deal with synonymy and ambiguity.

YAGO - Model (3)

- Similar entities are grouped into classes.
- Each entity is an instance of at least one class, by using the TYPE relation: Elvis Presley TYPE singer.
- Classes are also entities. Each class is itself an instance of the class class.
- Classes are arranged in a taxonomy, with the SUBCLASSOF relation.
- Relations are entities as well. It is possible to represent properties of relations: SubClassOf TYPE atr.
- In YAGO, facts are given a fact identifier.

YAGO - Model (4)

- Deviating from RDFS, fact identifiers are an integral part of the YAGO model.
- Each fact has a fact identifier. For example, suppose Elvis Presley BornInYear 1935 had the fact identifier #1. Then we can write: #1 FoundIn Wikipedia.
- Entities that are not facts or relations: common entities.
- Common entities that are not classes: individuals.

- In summary, *a YAGO ontology is basically a function that maps fact identifiers to fact triples.*

YAGO - Formally (1)

- A YAGO ontology can be given as a reification graph:
  - N set of nodes: common entities.
  - I set of edge identifiers: fact identifiers.
  - L set of labels: relation names.
  - The reification graph is an injective total function $G_{N,I,L} : I \rightarrow (N \cup I) \times L \times (N \cup I)$.
- A YAGO ontology over a finite set of common entities $\mathcal{C}$, a finite set of relation names $\mathcal{R}$ and a finite set of fact identifiers $\mathcal{I}$ is a reification graph

$$y : \mathcal{I} \rightarrow (\mathcal{I} \cup \mathcal{C} \cup \mathcal{R}) \times \mathcal{R} \times (\mathcal{I} \cup \mathcal{C} \cup \mathcal{R}).$$

YAGO - Formally (2)

- A YAGO ontology (any reification graph) is given as:

$$\begin{array}{llll} id_1: & arg1_1 & rel_1 & arg2_1 \\ id_2: & arg1_2 & rel_2 & arg2_2 \\ & \vdots & & \end{array}$$

- Shorthand notation:

  $id_2: \quad arg1_1\ rel_1\ arg2_1 \quad rel_2 \quad arg2_2$ to mean

$$\begin{array}{llll} id_1: & arg1_1 & rel_1 & arg2_1 \\ id_2: & id_1 & rel_2 & arg2_2 \end{array}$$

  where id_1 is a fresh identifier. Example:

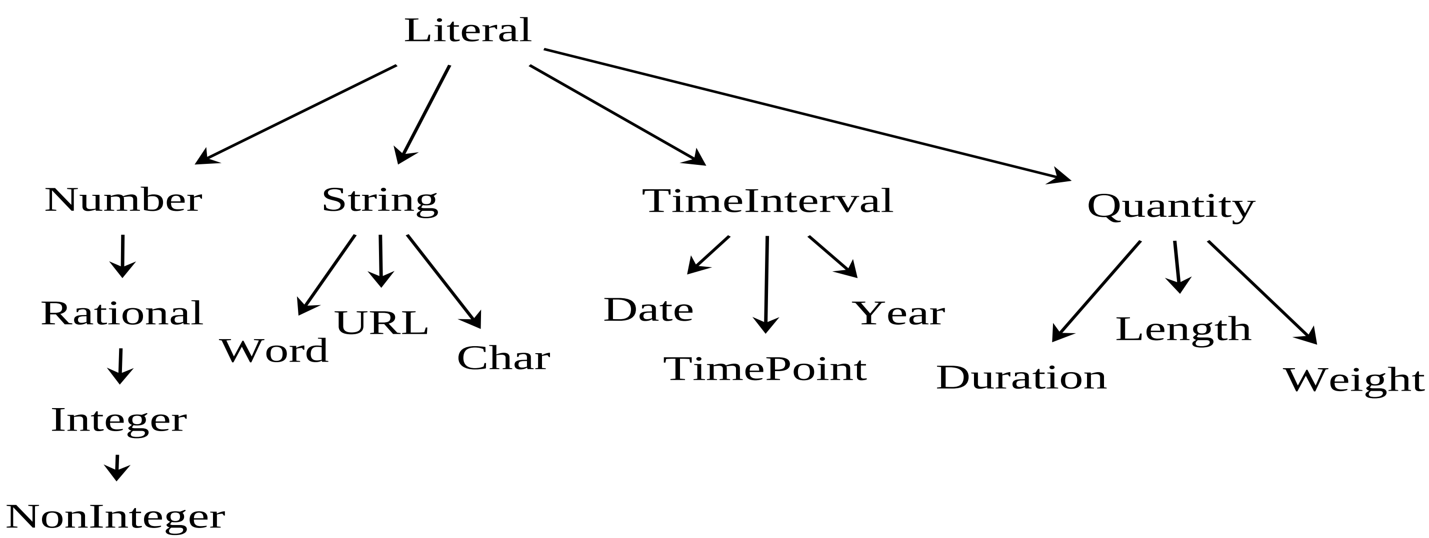
- Elvis Presley BornInYear 1935 FoundIn Wikipedia.

YAGO - n-ary relations

- Some facts require more than two arguments (e.g., the fact that Elvis got the Grammy Award in 1967).
- The YAGO model solution
  - Assumption: for each n-ary relation, a primary pair of its arguments can be identified. For relation WonPrizeInYear, $(\text{person}, \text{prize})$ is a primary pair.
  - Represent the primary pair as a binary fact:
    #1 : Elvis HasWonPrize Grammy Award
  - All other arguments: binary relations between the primary fact and the other #2 : #1 InYear 1967.
  - Elvis HasWonPrize Grammy Award InYear 1967.

YAGO - Semantics (1)

- Hypothesis. For any YAGO ontology:
  - $\mathcal{R}$ contains at least relation names: type, subClassOf, domain, range, subRelationOf.
  - $\mathcal{C}$ contains at least class entities: class, relation, atr.
  - Contain classes for all literals in figure bellow:



YAGO - Semantics (2)

- y YAGO ontology, $\mathcal{I} = dom(y)$, implicitely.
- $\mathcal{F} = (\mathcal{I} \cup \mathcal{C} \cup \mathcal{R}) \times \mathcal{R} \times (\mathcal{I} \cup \mathcal{C} \cup \mathcal{R})$ - all possible facts.
- Rewrite system $\rightarrow \subseteq 2^{\mathcal{F}} \times 2^{\mathcal{F}}$, i.e. $\rightarrow$ reduces one set of facts to another set of facts. Notation:
- $\{f_1, \ldots, f_n\} \hookrightarrow f \equiv \forall F \subseteq \mathcal{F}\ F \cup \{f_1, \ldots, f_n\} \rightarrow F \cup \{f\}$.
- Axiomatic rules:
  - $\emptyset \hookrightarrow$ (domain, RANGE, class) $\emptyset \hookrightarrow$ (domain, DOMAIN, relation)
    $\emptyset \hookrightarrow$ (range, DOMAIN, relation) $\emptyset \hookrightarrow$ (range, RANGE, class)
    $\emptyset \hookrightarrow$ (subClassOf, TYPE, atr) $\emptyset \hookrightarrow$ (subClassOf, DOMAIN, class)
    $\emptyset \hookrightarrow$ (subClassOf, RANGE, class) $\emptyset \hookrightarrow$ (type, RANGE, class)
    $\emptyset \hookrightarrow$ (subRelationOf, TYPE, atr) $\emptyset \hookrightarrow$ (subRelationOf, DOMAIN, relation)
    $\emptyset \hookrightarrow$ (subRelationOf, RANGE, relation)

YAGO - Semantics (3)

- Axiomatic rules (continued):
  - For each edge $X \rightarrow Y$, in the literal diagram, we have the rule $\emptyset \hookrightarrow$ (X, SUBCLASSOF, Y)
  - $\forall r, r_1, r_2 \in \mathcal{R}$, $\forall x, y, c, c_1, c_2 \in \mathcal{I} \cup \mathcal{C} \cup \mathcal{R}$, $r_1 \neq$ TYPE, $r \neq$ TYPE, $r_2 \neq$ SUBRELATIONOF, $r \neq$ SUBRELATIONOF, $c \neq$ atr, $c_2 \neq$ atr :
    $\{(r_1, \text{SUBRELATIONOF}, r_2), (x, r_1, y)\} \hookrightarrow (x, r_2, y)$
    $\{(r, \text{TYPE}, \text{atr}), (x, r, y), (y, r, z)\} \hookrightarrow (x, r, z)$
    $\{(r, \text{DOMAIN}, c), (x, r, y)\} \hookrightarrow (x, \text{TYPE}, c)$
    $\{(r, \text{RANGE}, c), (x, r, y)\} \hookrightarrow (y, \text{TYPE}, c)$
    $\{(x, \text{TYPE}, c_1), (c_1, \text{SUBCLASSOF}, c_2)\} \hookrightarrow (x, \text{TYPE}, c_2)$
- Theorem *Given a set of facts $F \subseteq \mathcal{F}$, the largest set S with $F \rightarrow^* S$ is finite and unique.*
- y YAGO ontology, applying $\rightarrow$ to its facts $range(y)$, gives the set of derivable facts of y, $D(y)$.

YAGO - Semantics (4)

- y YAGO ontology, its deductive closure is $y^* = y \cup \{(f_{r,a,b}, (a,r,b)) | (a,r,b) \in D(y) \setminus range(y)\}$.
- A structure for a YAGO ontology y is a triple $\langle \mathcal{U}, \mathcal{D}, \mathcal{E} \rangle$:
  - $\mathcal{U}$ is a set, the universe.
  - $\mathcal{D}$ is a function, $\mathcal{D} : \mathcal{I} \cup \mathcal{C} \cup \mathcal{R} \rightarrow \mathcal{U}$, the denotation.
  - $\mathcal{E} : \mathcal{D}(\mathcal{R}) \rightarrow \mathcal{U} \times \mathcal{U}$, the extension function.
- Interpretation Ψ of y with respect to a structure $\langle \mathcal{U}, \mathcal{D}, \mathcal{E} \rangle$ is the following relation: $\Psi = \{(e_1, r, e_2) | (\mathcal{D}(e_1), \mathcal{D}(e_2)) \in \mathcal{E}(\mathcal{D}(r))\}$.
- a fact (e_1, r, e_2) is true in a structure, if it belongs Ψ.

YAGO - Semantics (5)

- A model of a YAGO ontology y is a structure such that
  - all facts of y^* are true in the structure,
  - if $\Psi(x, \text{TYPE}, \text{string})$ for some x, then $\mathcal{D}(x) = x$,
  - if $\Psi(r, \text{TYPE}, \text{atr})$ for some r, then $\nexists x$ s.t. $\Psi(r, x, r)$.
- Ontology y is consistent if there exists a model for it.
- *The consistency of a YAGO ontology is decidable.*

YAGO - Query Language

- A pattern for a reification graph $G_{N,I,L}$ over a set of variables V, $V \cap (N \cup I \cup L) = \emptyset$, is any reification graph $G_{N \cup V, I \cup V, L \cup V}$.
- Variables - symbols with a question mark (e.g., $?x$).
- A matching of a pattern P for a graph G is a substitution $\sigma : V \rightarrow N \cup I \cup L$, such that $\sigma(P) \subset G$. $\sigma(P)$ is called a match.
- The query "When did Elvis win the Grammy Award?" can be formulated as Elvis HasWonPrize Grammy Award InYear $?x$.

YAGO - Query Engine

- A simple query engine on top of the database version of YAGO was implemented.
- A "demo" could be found at http://www.mpi-inf.mpg.de/yago-naga/yago/demo.html

YAGO - Construction (1)

- The construction of the YAGO ontology takes place in two stages:
  - Different heuristics are applied to Wikipedia to extract candidate entities and candidate facts. This stage also establishes the connection between Wikipedia and WordNet.
  - Quality control techniques are applied: Type Checking and Canonicalization.

YAGO - Construction (2)

- Each row of the infobox will generate one fact: its first argument is the article entity, its relation is determined by attribute and its second argument is the value of the attribute.
- Only the leaf conceptual categories of Wikipedia are considered and ignore all higher categories.
- WordNet is used to establish the hierarchy of classes, by the taxonomy of synsets.
- Wikipedia and WordNet yield also word meaning.

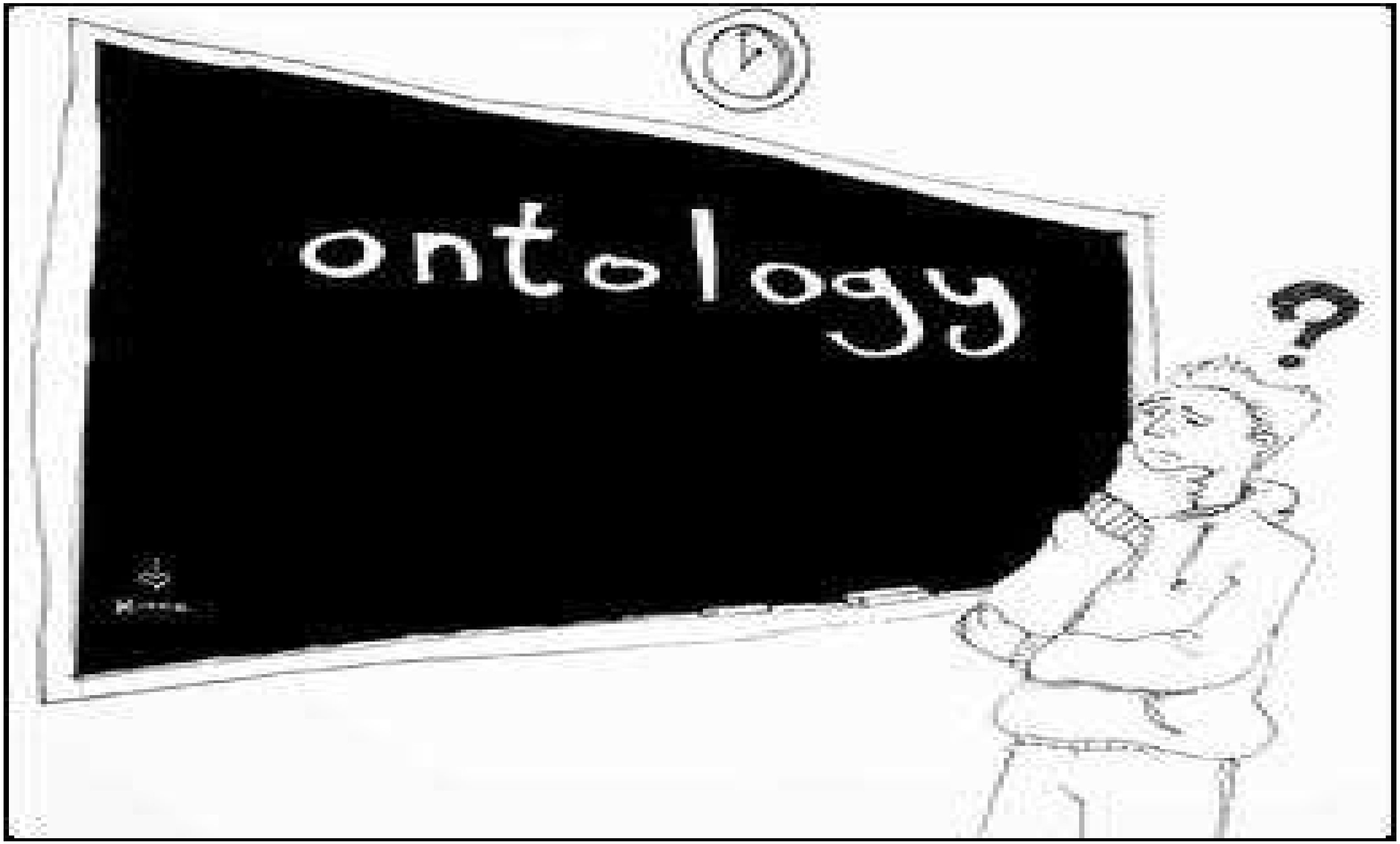
YAGO - Outlook

- YAGO opens up new opportunities and challenges.
- A positive feedback loop, in which the addition of knowledge helps the extraction of new knowledge.
- YAGO can be freely downloaded from the Web site http://www.mpii.de/yago.
- Availability of a huge, clean, and high quality ontology can give new impulses to the Semantic Web vision.

Bibliography

1. Nicola Guarino, Ontology-Driven Conceptual Modeling, Tutorial given at ER2002.
2. Paul Buitelaar, Philipp Cimiano, Ontologies and Lexical Semantics in Natural Language understanding, Tutorial given at 19th European Summer School on Logic, Language and Computation, ESSLLI 2007.
3. Fabian M. Suchanek, Gjergji Kasneci, Gerhard Weikum, YAGO:A Large Ontology from Wikipedia and WordNet, *Web Semantics: Science, Services and Agents on the World Wide Web*, Volume 6 Issue 3, September, 2008

Questions ?



THANKS!