Semantic Web and **RDF(S)**

A Biased Introduction

Atanas Kiryakov

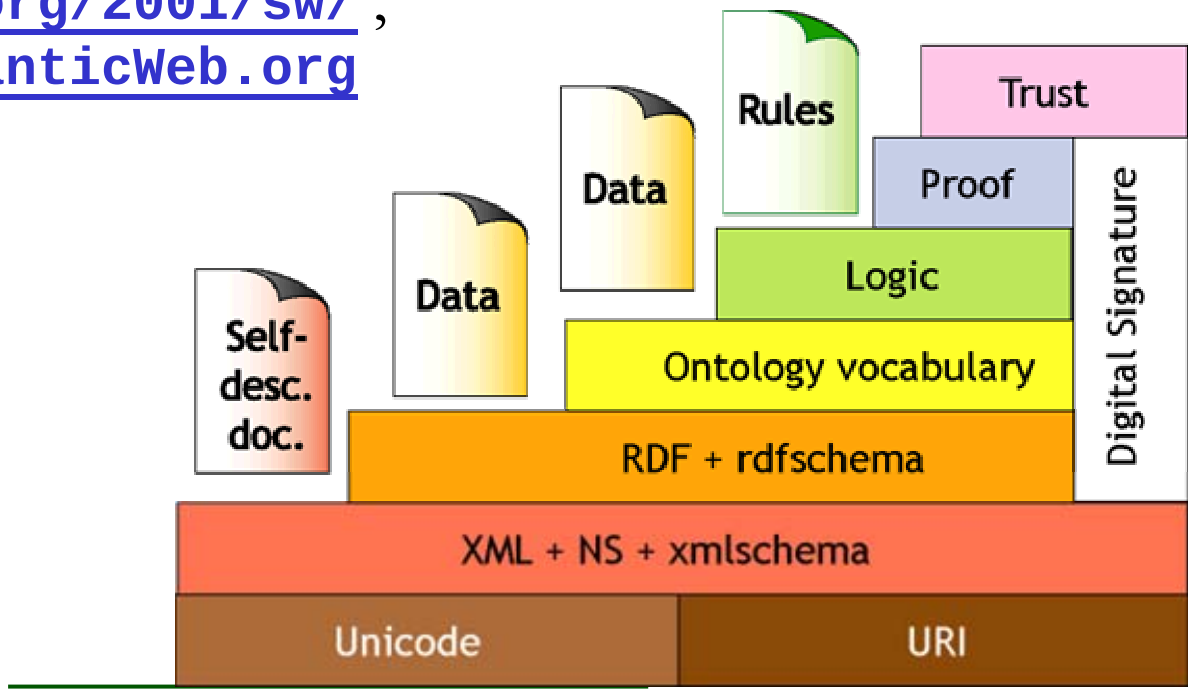
11/6/2003

Outline

- **Semantic Web**
- RDF(S) – data-model and representation
- Querying
- Conclusion

Semantic Web

- The Semantic Web is the **abstract representation of data** on the WWW, based on the RDF and other standards
- SW is being **developed by the W3C**, in collaboration with a large number of researchers and industrial partners
- http://www.w3.org/2001/sw/ , http://www.SemanticWeb.org

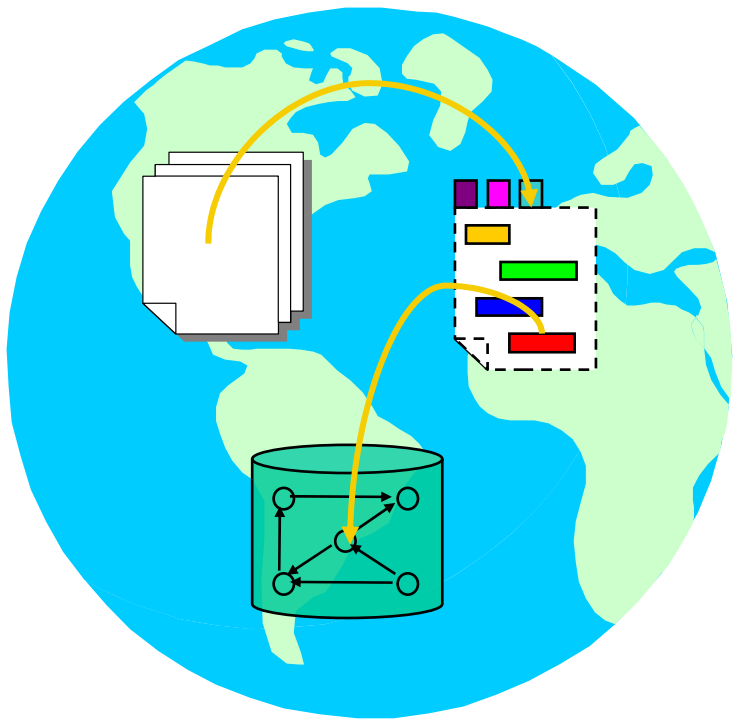


Semantic Web (II)

- "The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation." [Berners-Lee et al. 2001]

The spirit:

- Automatically processable **metadata** regarding:
  - the structure (syntax) and
  - the meaning (semantics)
  - of the content.
- Presented in a **standard form**;
- **Dynamic interpretation for unforeseen purposes**



Semantic Web: Languages

- **RDF(S)** – the next slides
- SHOE, XOL, etc – the pioneers (now history)
- Topic Maps – a metadata language with limited impact
- **OIL** – Ontology Interchange Language, the basis of the next two
  `http://www.ontoknowledge.org/oil/`
  - Description Logics-based multilayered language
- DAML+OIL – the predecessor of OWL, not to be developed
- **OWL** – the W3C standard for Semantic Web ontology language,
  `http://www.w3.org/2001/sw/WebOnt/`
  - Extends RDF(S), but also constraints it
  - Has multiple layers (Lite, DL, Full)
  - Transitive/symmetric/etc properties, disjointness, cardinality restrictions

Semantic Web: Problems

- Critical mass of metadata is necessary
- Still lack of consensus on many issues (like query languages)
- Lack of practices at the proper scale and complexity
- Lack of robust Semantic (our days RDFS) repositories:
  - Those should has in the Semantic Web role combining those of (i) HTTP servers for the current and (ii) RDBMS for the information systems
  - Should be as flexible, multi-purpose and easy to use as HTTP servers and
  - As efficient in structured knowledge management as RDBMS

Outline

- Semantic Web
- **RDF(S) – data-model and representation**
- Querying
- Conclusion

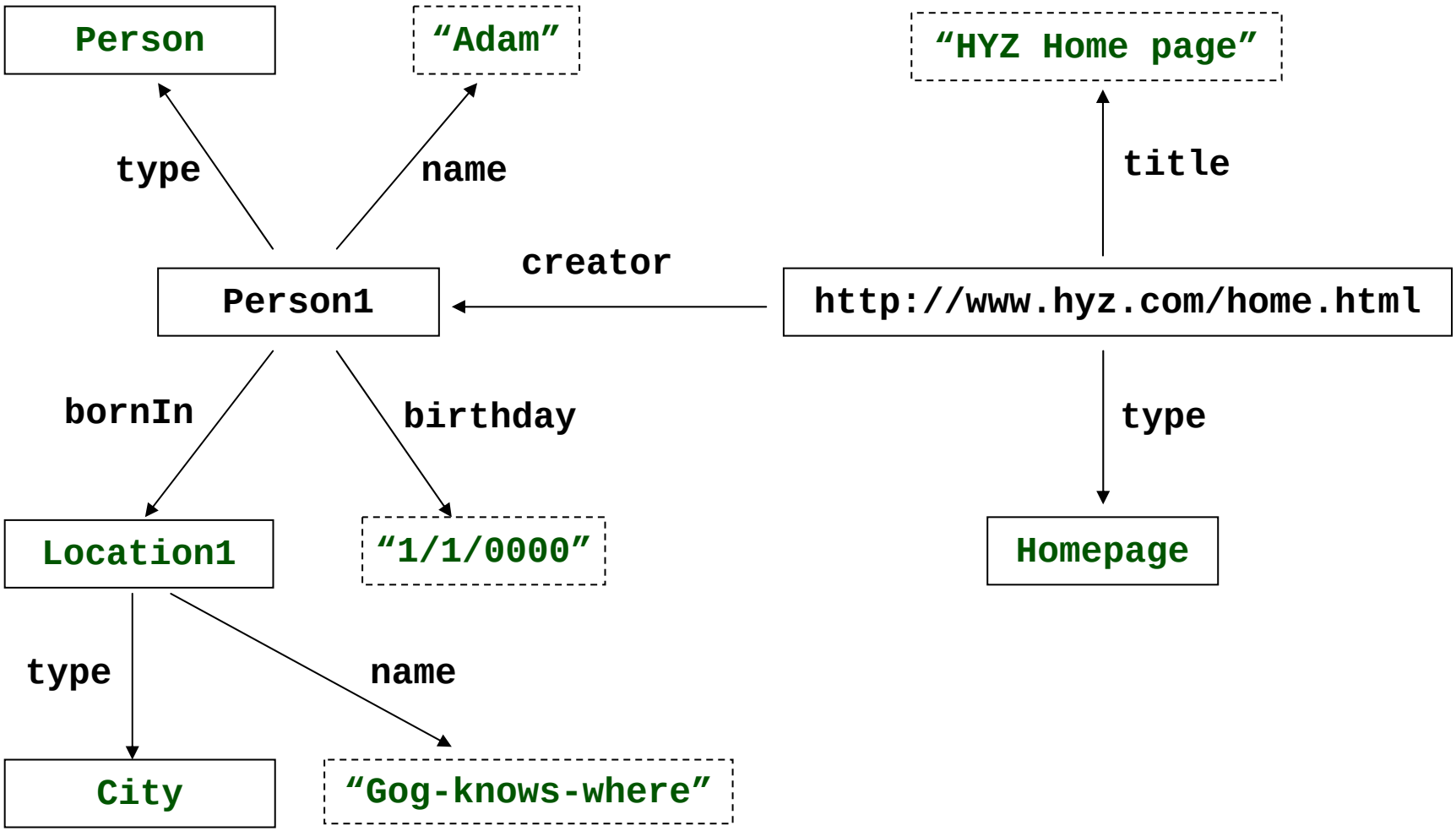
RDF

- Resource Description Framework
- A W3C standard, check **http://www.w3.org/RDF/**
- RDF – metadata description language
- **Metadata** can be associated with Resources
- **Resource** could be anything, from this presentation to Jupiter
- Resources are identified by **URIs** ("unique res. identifiers")
  - For instance URLs, but also ISBN and the social security #
- **Literals** are not resources, rather … just literals
  - out of the XML data types: "abc", "1/1/1970"
  - Does are always leaves in the graph
  - They cannot be described, or addressed again

RDF (II)

- Resources could be instances of/belong to **Classes**
  - Single resource can belong to multiple classes
  - The classes are resources themselves
- The resources are described via **statements**: triples which link the resource to another one or to a literal via named property:
  - **< Resource1, Property, Resource2 >**
  - **< Resource1, Property, Literal >**
- So, the basic data-model is **directed labeled graph**

RDF: Piece of graph



RDF vs. XML

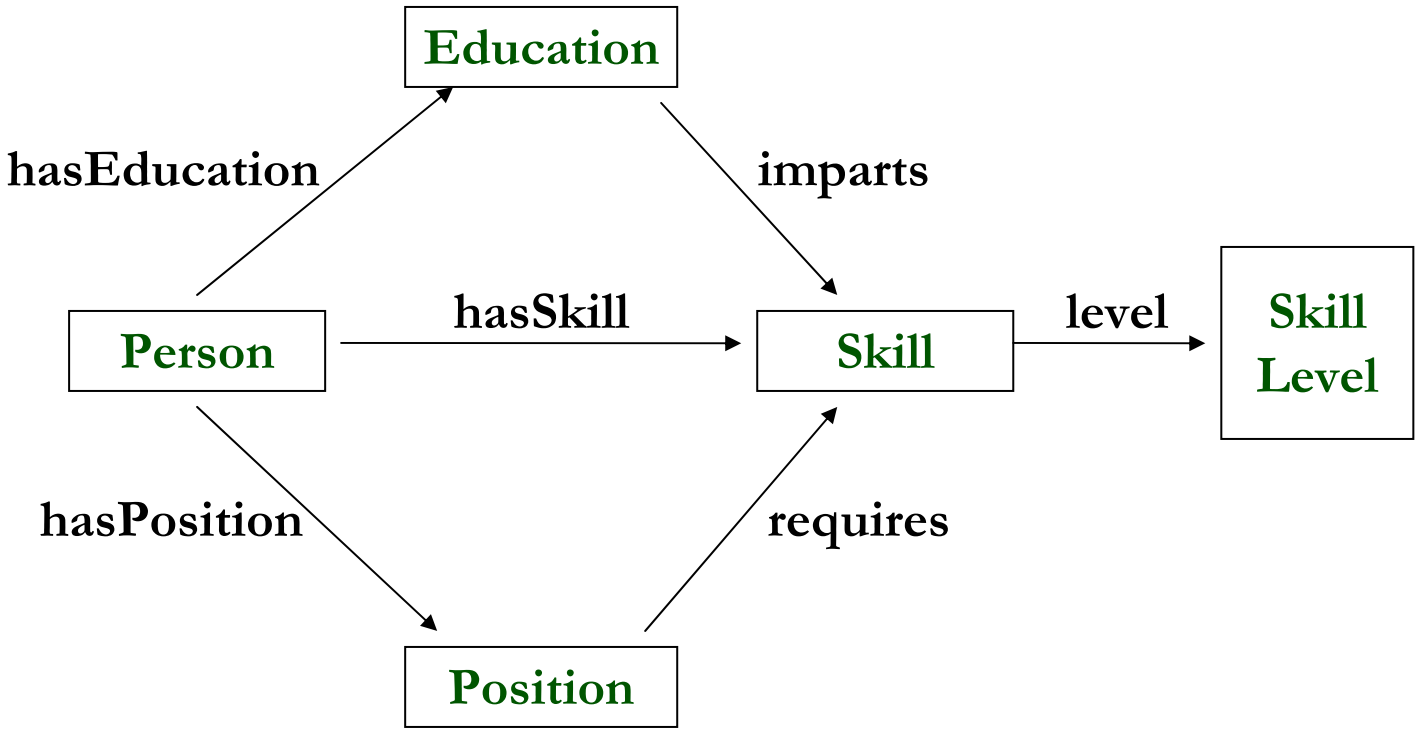
- Major difference in the data-model
  - RDF(S): graph of resources
  - XML: tree of elements

- The first and most popular syntax of RDF is in XML
  - But this is *just* syntax
  - One cannot explore (query, etc) RDF(S) with XML tools
  - There are number of new (more efficient and readable) syntaxes from RDF, such as, N3

RDFS

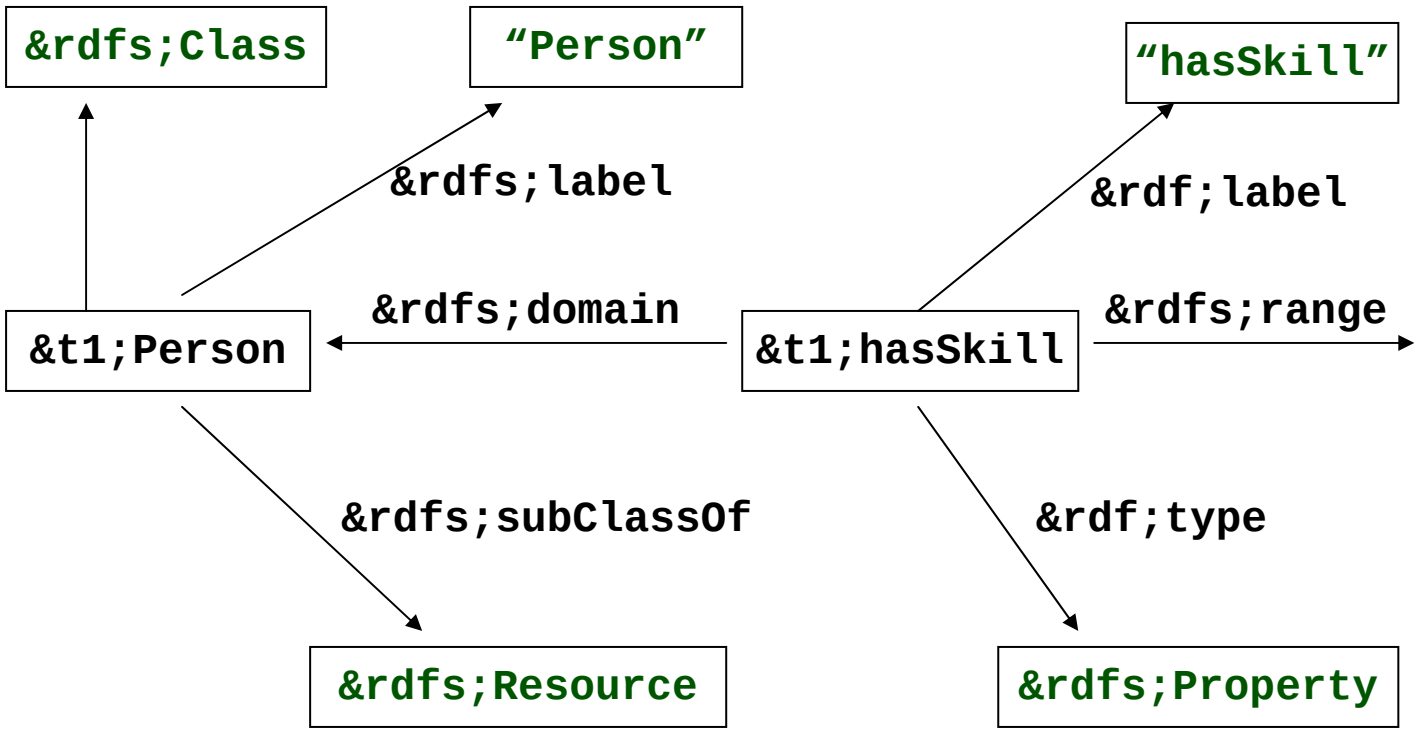
- Resource Description Framework **Schema**
- Allows definition of new classes of resources and properties
- Can be considered a simple Knowledge Representation/Ontology Language
- Features:
  - Hierarchy of classes
  - Hierarchy of properties (like, **`fatherOf`** is specialization of **`parentOf`**)
  - Domain and Range restrictions of the properties
  - No limitations in the levels of instantiation – e.g. one resource could be a class itself and instance of another class (sort of meta-class)
  - No local properties (there can not be different name properties with specific restrictions for the different classes)

RDF: The schema as ER-graph

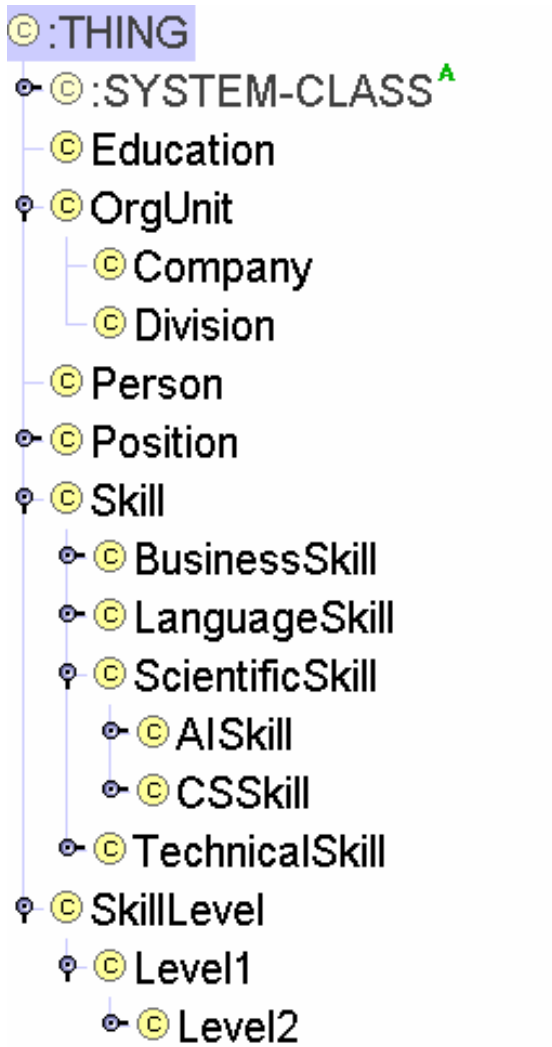


Simplified conceptual schema of a Skills Management Application

RDF: Piece of the real graph



RDF: Sample Class Hierarchy



RDF: Property Hierarchy

- What it means to have sub-properties?

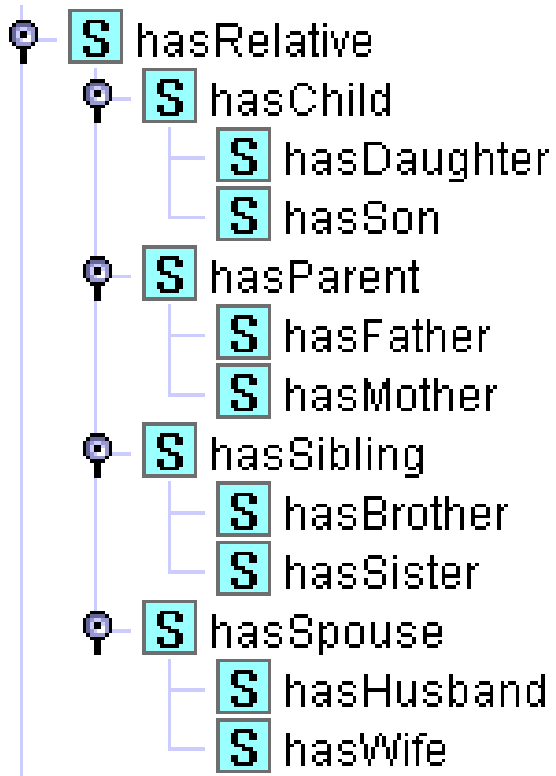
If **<p1, subProperty, p2>** and **<A, p1, B>** than **<A, p2, B>**

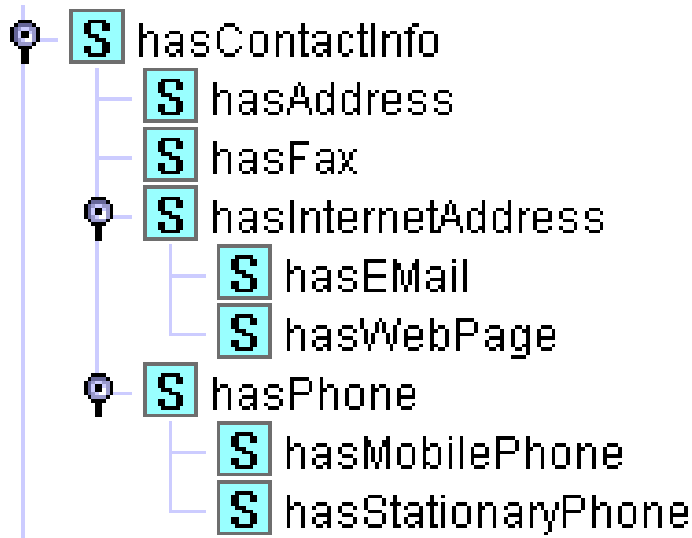
- Who really need this?
  - It is in order to allow you manipulation of the knowledge at different levels of generality
  - This is what the hierarchies/taxonomies are used for, actually
  - To be able to store specific information and answer both specific and general questions.

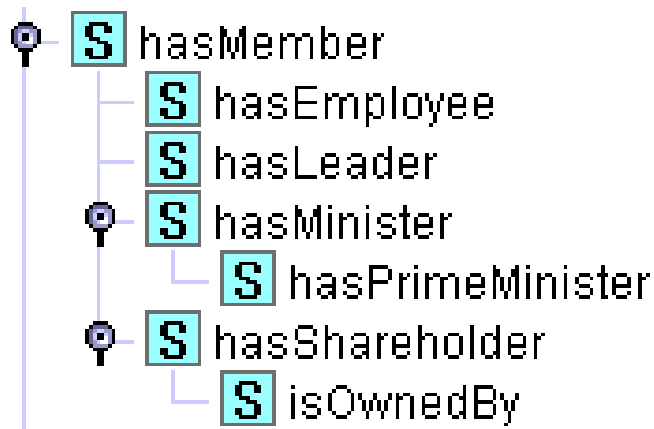
Imagine, you have in the KB that **<Peter, sonOf, John>**

Should John match: **<Peter, hasRelative, X>**

RDF: Property Hierarchy (II)





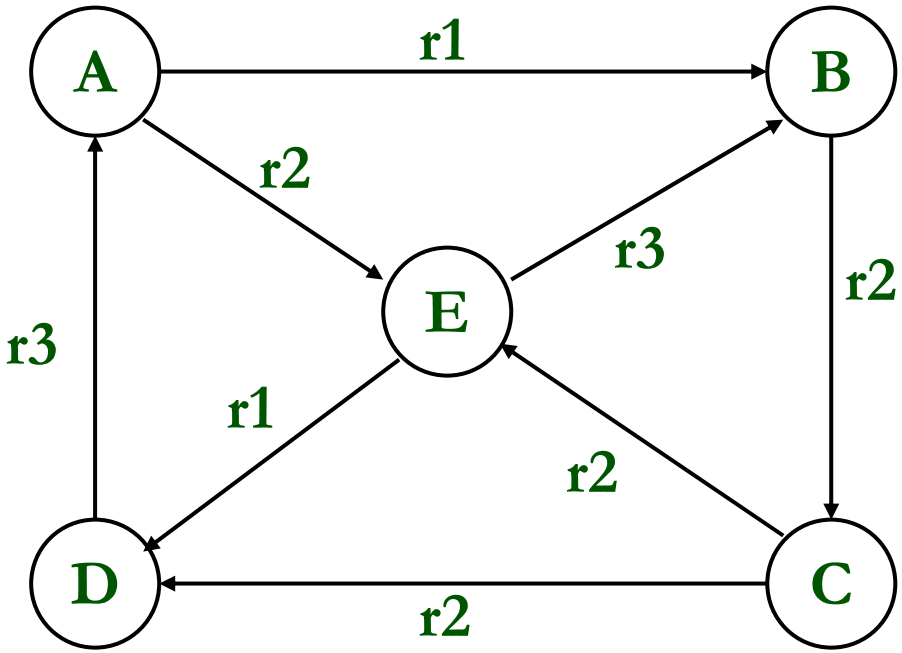


Examples from KIMO (the KIM Ontology)
www.ontotext.com/KIM/kimo.rdfs

Plan

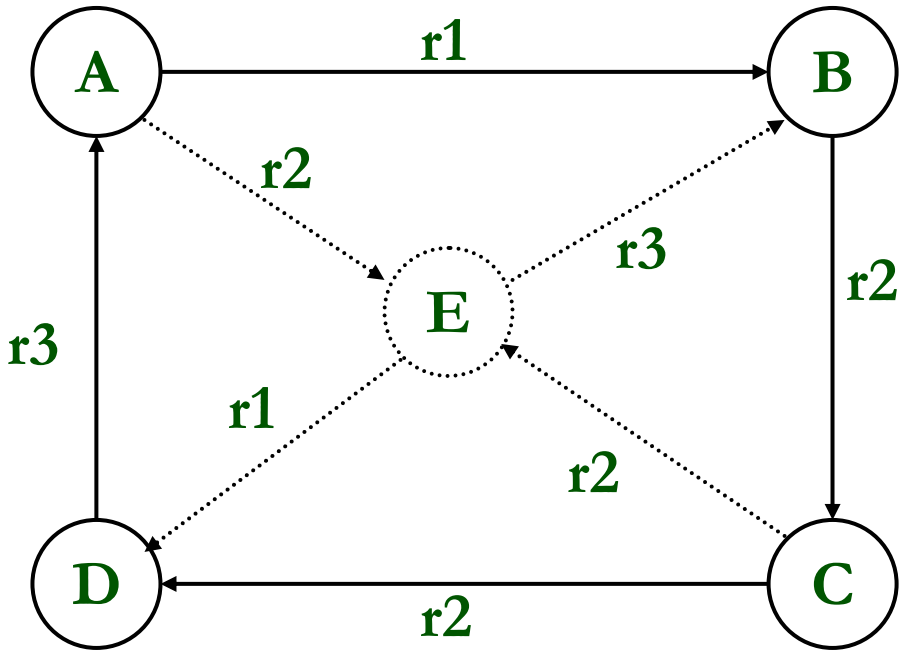
- Semantic Web
- RDF(S) – data-model and representation
- **Querying**
- Conclusion

RDF: Querying



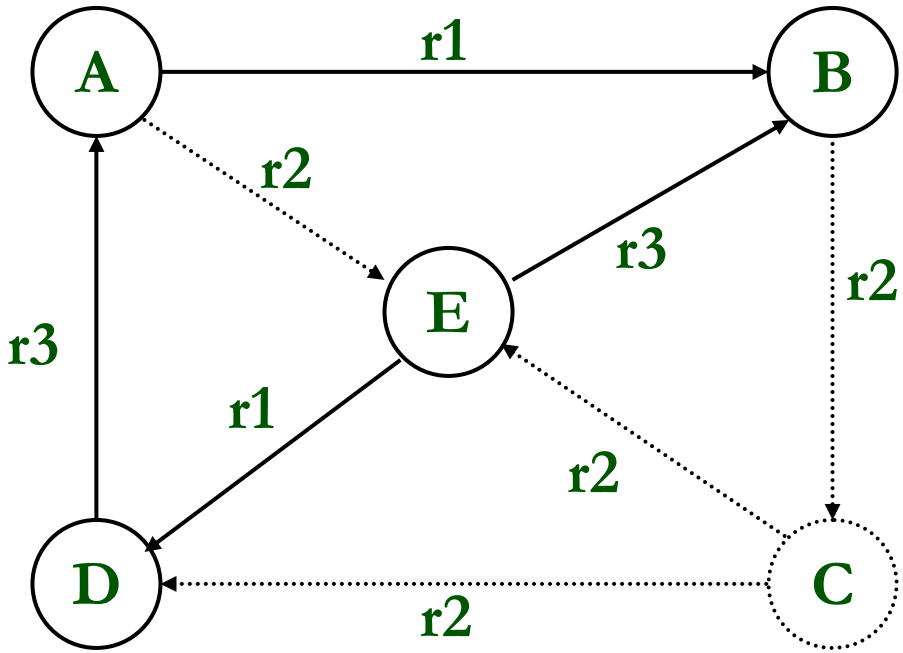
An RDF repository represented as graph. The following slides will present the result of few RQL queries over it.

RDF: Querying by Resource



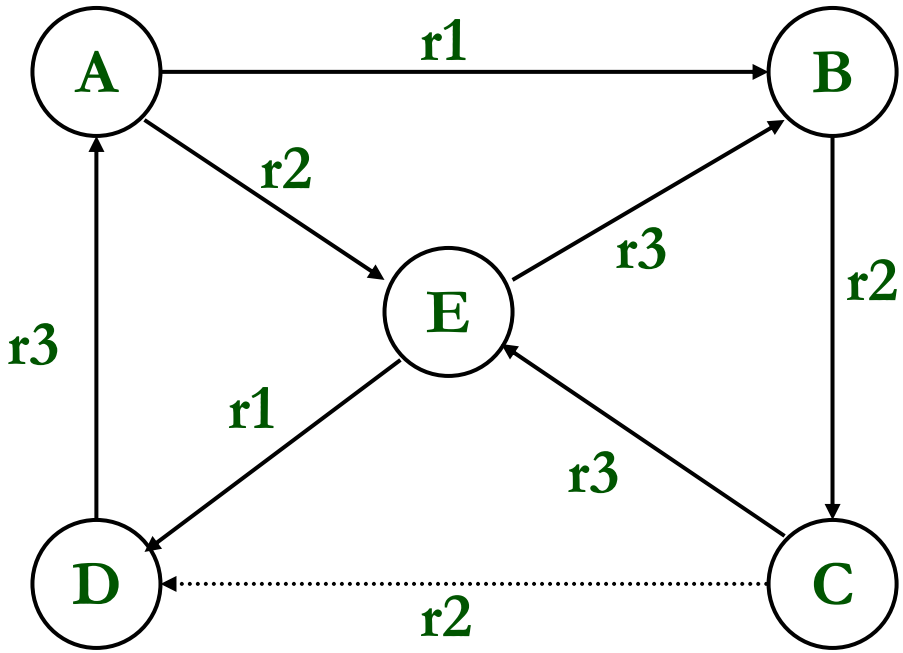

```
SELECT X, @P, Y
FROM {X} @P {Y}
WHERE X != E and Y != E
```

RDF: Querying by Property




```
SELECT X, @P, Y
FROM {X} @P {Y}
WHERE @P != r2
```

RDF: A More Complex Query




```
SELECT X, @P, Y
FROM {X} @P {Y}, {X} @Q {#E}. @R {Y}
WHERE @P = #r2
```

Plan

- Semantic Web
- RDF(S) – data-model and representation
- Querying
- **Conclusion**

RDF(S) Links

- RDF(S) specs, start from **http://www.w3.org/RDF/**
- RDF(S) repository … many, varying functionality. Try:
  - Sesame: **http://sesame.aidministrator.nl/**
  - KAON: **http://kaon.semanticweb.org/**
- Editors … number of: Protégé 2000, OilEd, OntoEdit
  - But none of them good enough to replace XMLSpy or UltraEdit

- Query languages – no consensus. Give a check to:
  - RQL, **http://sesame../publications/rql-tutorial.html**
  - SeRQL, **http://sesame../publications/users/ch05.html**
  - RDQL, **http://www.hpl.hp.com/semweb/rdql.htm**

- Sample usage, see **http://www.ontotext.com/kim** (download the IE plug-in and play)

RDF(S) Current State

- There are many tools supporting it, but still a young technology, far not as robust as the RDBMS and even OO-DBMS
- Most important is the repository (the equivalent of the DB-server). Most of those are based on relational DBs
- No much front-ends
- The power of the language to be used in a wisely!
- The front-end tools (editors, whatever) are immature
- There are new standards like OWL which allow more complex features of the representation
- Quite a lot of development can be expected in the years to come

Conclusion

- **The Semantic Web has no alternative**
  - But has problems
  - Just needs time to develop

- Most standards and tools (somehow) comply with RDF(S)
- **RDF(S) is the HTML of the Semantic Web**
  - Many complains that it allows too much or too less, but … it's in use
- **OWL is to RDF(S) like XML is to HTML**

- The major issue is to get **critical mass of metadata**
- And methods to easily produce more on demand

Thanks!

And remember, **this was a biased introduction!**