- Usually, one gives an introduction to SW…

  …and then, questions are asked
- But this audience already knows the introduction…

  …so let us move to questions right away!

So...

Questions?

Well, Let Me Help You...

- Some questions come up regularly, so I collected them ☺

1. Is the Semantic Web AI on the Web?
2. Where is the "Web" in SW?
3. Isn't the RDF Model way too complex?
4. Why should I use RDF?
5. With *huge* ontologies on the Web, does this scale?
6. Where does the metadata come from?
7. Isn't This Research Only?
8. Does SW Replace Web Services?
9. Are we done?
10. What can I read to understand?

Is the Semantic Web AI on the Web?

NO!!!

- RDF and OWL are very simple things (compared to AI, that is...)
- They offer:
  - a simple way to express and store metadata
  - a way to "structure" and characterize the terms
  - means to make some simple inference

  and that is it!

RDF (Resource Description Framework)

- RDF is a set of *statements*
- Statements can be modeled (mathematically) with:
  - Resources: an element, a URI, a literal, ...
  - Properties: *directed* relations between *two* resources
  - Statements: "triples" of two resources bound by a property
    - usual terminology: (s,p,o) for subject, property, object
- *RDF* is a general model for such statements

OWL (Web Ontology Language)

- OWL refines the usage of RDF by:
  - defining the terminology used in a specific context (ontologies)
  - imposing constraints on properties
    - e.g., cardinality constraints
  - characterizing the logical characteristics of properties
    - e.g., transitivity, functionality
  - defining the equivalence of terms across ontologies
  - etc.

(to be precise: these are done by RDFS+OWL)

OWL and Logic

- OWL expresses a *small subset* of First Order Logic
  - it has a "structure" (class hierarchies, properties, datatypes...), and "axioms" can be stated within that structure only
  - i.e., OWL uses FOL to describe "traditional" ontology concepts... ...but it is *not* a general logic system per se!
  - (the same is true for RDFS, by the way)
- Inference based on OWL is *within this framework only*
  - it seems modest, but has proven to be remarkably useful...

And what about AI?

- AI requires much more complicated logical inferences
- There are *lots* of things RDF/OWL cannot express, eg:
  - the "uncle" relationship:

    $\forall x,z: ((\exists y: (y \text{ parent } x) \wedge (y \text{ brother } z)) \Rightarrow (z \text{ uncle } x))$

  - temporal and spatial reasoning
  - fuzzy logic
  - ...
- AI can be built by using OWL *and many other things*...

  ... but that is true for about everyting in IT 😉

Where is the "Web" in SW?

The "Web" is in the URI-s!

- On the SW, resources are identified by URI-s, e.g.:
  - URL-s
    - http://www.ivan-herman.net
    - ftp://ftp.cwi.nl
  - URN-s
    - urn:ISBN:0-395-36341-1
    - urn:lsid:ensembl.org:homosapiens_gene:ensg00000002016
- Anybody can create metadata on any resource on the Web
- It becomes easy to merge metadata
- *URI-s ground RDF into the Web*

Related Question...

- Q: People have misused HTML's **meta** elements... Why would that be different?
- A: The **meta** elements are *in the HTML source*
  - i.e., only the authors can set them
  - on the SW, *anybody* can define metadata
  - so one can get around misuse...

Isn’t the RDF Model way too complex?

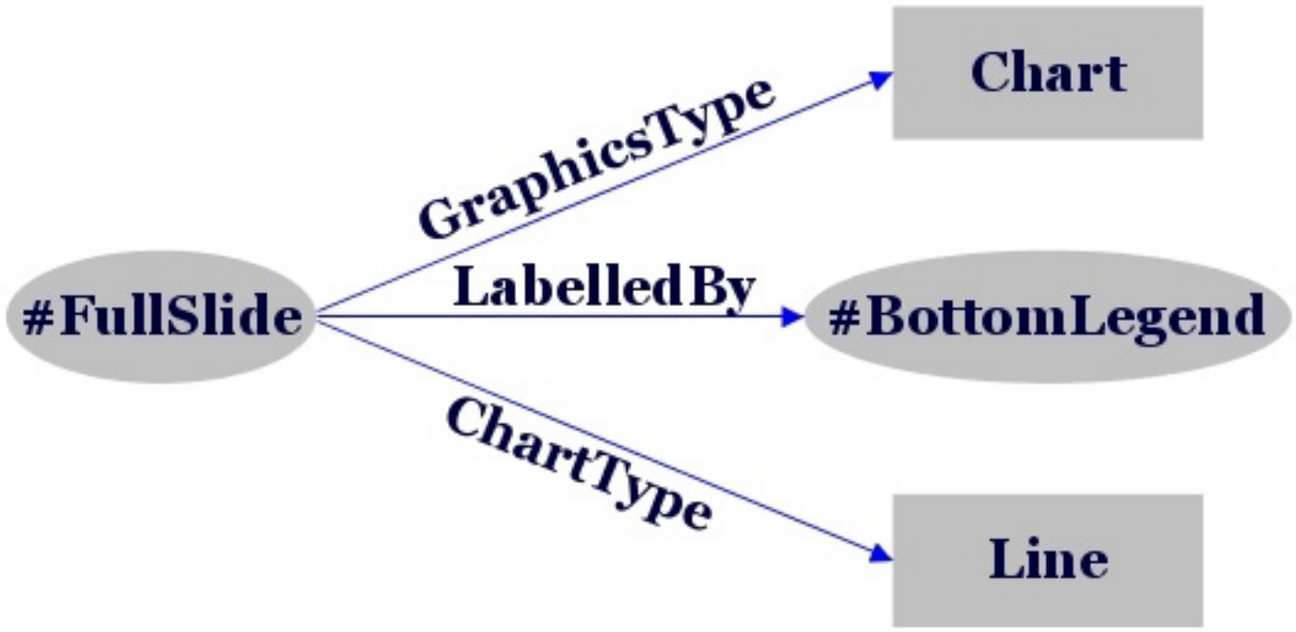
(look how complex RDF/XML is 😟)

RDF is a graph!

- An (s,p,o) triple can be viewed as a labelled edge in a graph
  - i.e., a set of RDF statements is a *directed, labelled graph*
    - both "objects" and "subjects" are the graph nodes
    - "properties" are the edges
  - the formal semantics of RDF is also described using graphs
- One should "think" in terms of graphs, and…

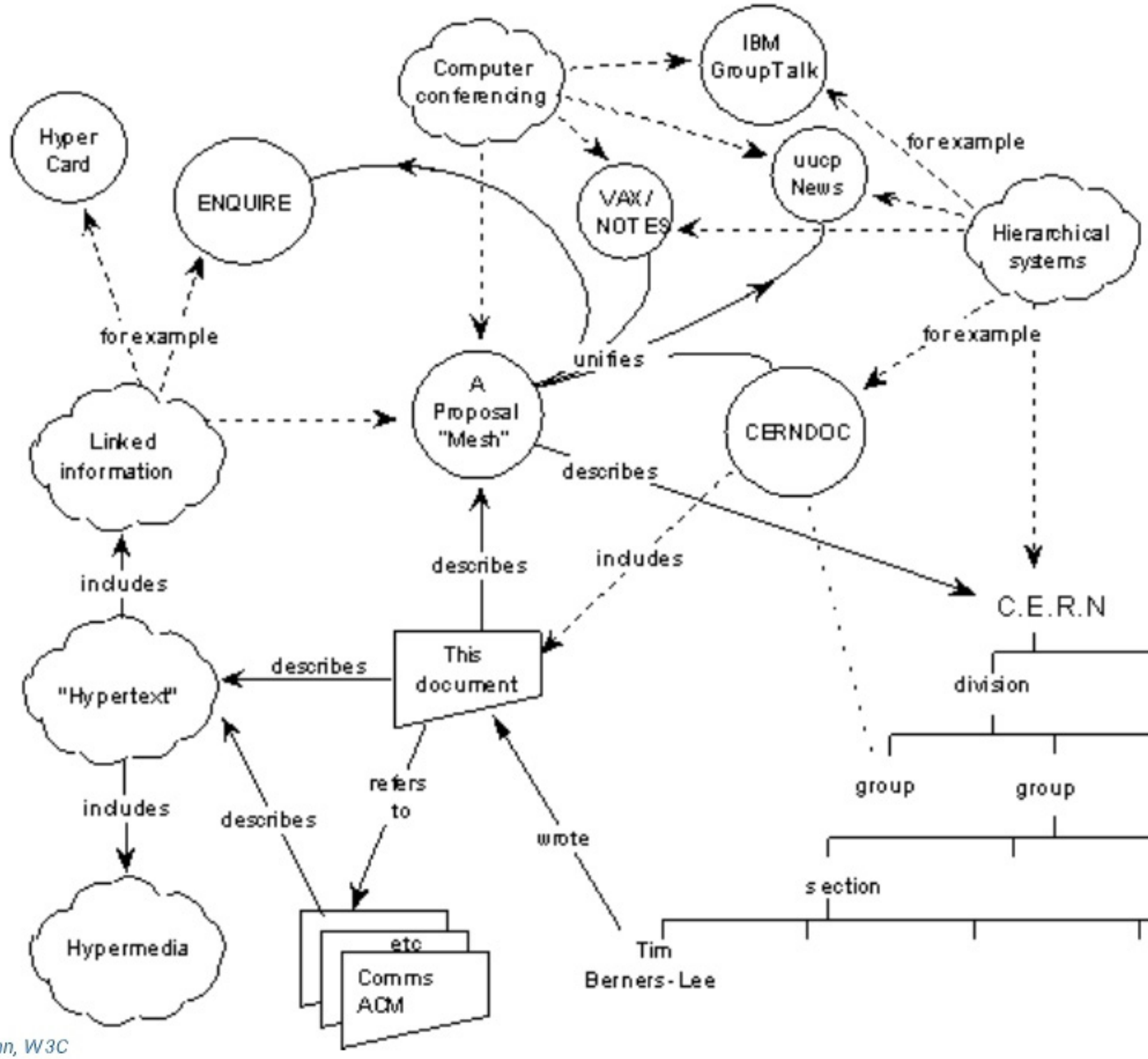
  …RDF/XML is only a tool for practical usage!
- RDF authoring tools often work with graphs, too (XML is done "behind the scenes")
- If one thinks in graphs, things become simple!

A Simple RDF Example




```
<rdf:Description rdf:about="#FullSlide">
   <axsvg:GraphicsType>Chart</axsvg:GraphicsType>
   <axsvg:LabelledBy rdf:resource="#BottomLegend"/>
   <axsvg:ChartType>Line</axsvg:ChartType>
</rdf:Description>
```

Things Are Not That New...



RDF/XML has its Problems

- RDF/XML was developed in the "prehistory" of XML
  - e.g., even namespaces did not exist!
- Coordination was not perfect, leading to problems
  - the syntax cannot be checked with XML DTD-s
  - XML schemas are also a problem
  - encoding is verbose and complex
    - (e.g., simplifications lead to confusions)

  but there is too much legacy code 😟
- Don't be influenced (and set back...) by the XML format
  - the important point is the *model*, XML is just syntax
  - other "serialization" methods may come to the fore

Other Encodings Examples...

- Turtle, n3, N-triples (variants of one another):

```
:object :pred [
   :pred2 :val1;
   :pred3 :val2;
]
```

- RXR (Regular XML RDF):

```
<triple>
     <subject uri="..."/>
     <predicate uri="..."/>
     <object>A Literal</object>
</triple>
```

- OWL "Abstract Syntax":

```
Class(animate)
Class(animateMotion)
Class(animationEntity complete
     unionOf(animate animateMotion ...)
)
```

- Again: these are all just syntactic sugar!

Why should I use RDF?

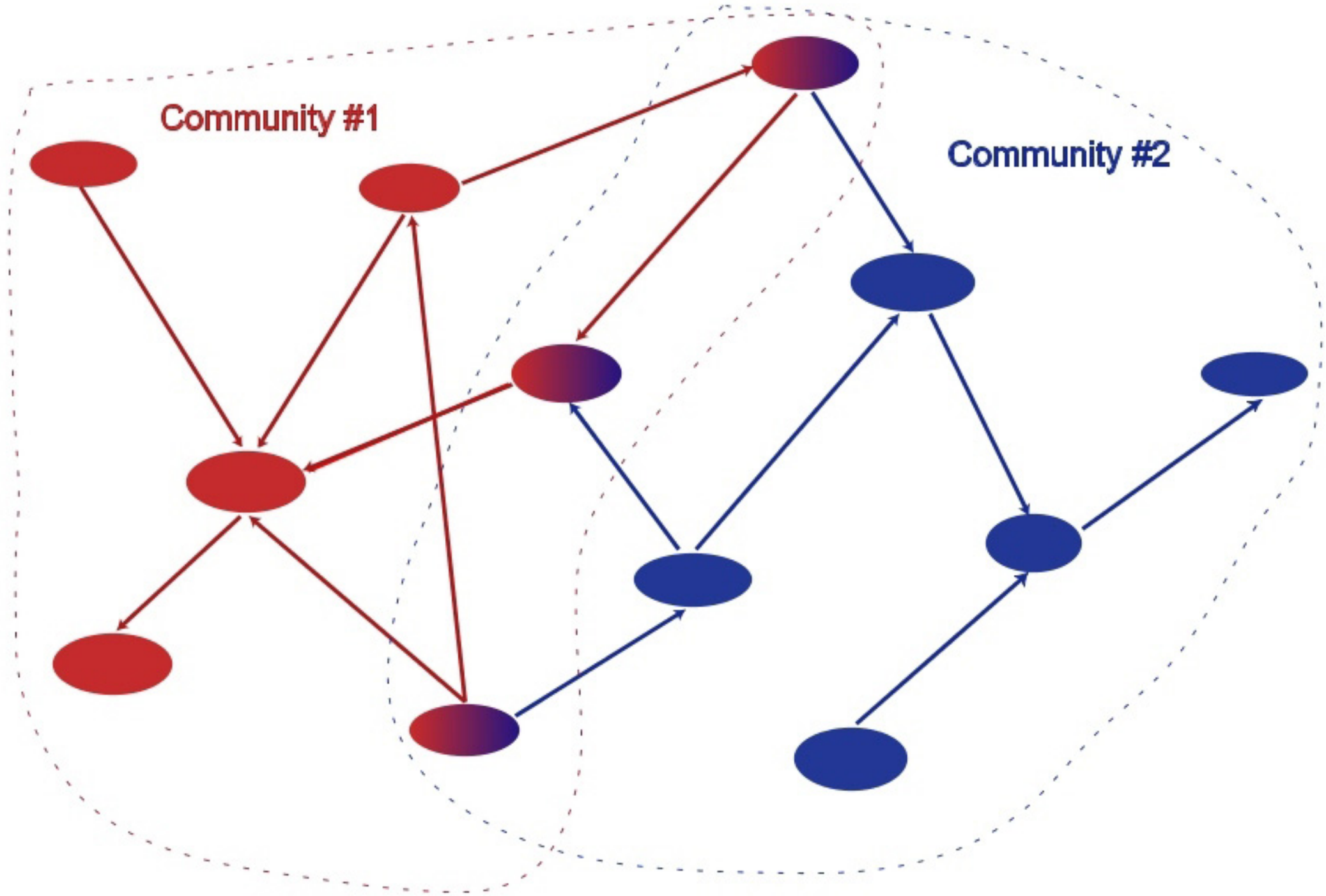
(Couldn't I simply use XML with XML Schema instead?)

It Depends...

- XML's model is
  - a tree, i.e., a strong hierarchy
  - applications may rely on hierarchy position (e.g., `li` in HTML)
  - relatively simple syntax and structure
  - not easy to *combine* trees
- RDF's model is
  - a *loose* collections of relations
  - applications may do "database"-like search
  - not easy to recover hierarchy
  - easy to combine relations in one big collection
    - great for the integration of heterogeneous information

Merge Shown as a Graph...

(note the power of the graph notation...)



"Joint" Community #1 and #2

Extra Bonus: OWL

- You may not use OWL reasoning yet…
- …but you may in future, RDF leaves the door open!

Finding New Relationships

- RDF (possibly with OWL) helps in finding *new* relationships
  - e.g., in Life Sciences:
    - most of the drug experiments are unsuccessful
    - but the information from *each* experiment may be valuable
    - by "binding" this information new insights can be gained

  (currently, life sciences are *very* excited by the prospects of the Semantic Web!)
- Sharing and aggregation of data becomes easier
  - may be determinant for future R&D, for example
  - great tool for general community building

With *huge* ontologies on the Web, does this scale?

It May Be a Problem, But...

- Yes, reasoning over *huge* ontologies may be a problem
  - combination of ontologies may lead to this
  - DL systems shown to work for ≈100k concepts already
    - albeit with a simple structure
  - there are already applications with large ontologies (see later)
  - lots of R&D is happening here...

    but it is indeed still a challenge
- But: "a little semantics can take you far" (Jim Hendler)
  - i.e., small OWL ontologies may lead to useful applications
  - applications may use ontology "islands", and loosely bind them
    - OWL gives you tools for that

The "Ideal" Pattern

- The ideal application pattern:
  - $\mathscr{R}$(RDFS,OWL),Ω; where:
    - $\mathscr{R}$(RDFS,OWL) is an off-the-shelf OWL reasoner
    - Ω is an ontology based on RDFS+OWL
  - an application means:
    - define metadata $\mathscr{M}_{\Omega}$, based on Ω
    - use $\mathscr{R}$ to deduce facts, answer questions
- But that does not always work...

The "Pragmatic" Pattern

- The alternative (more realistic?) pattern:
  - $\wp_{\Omega}(\mathscr{R}(\text{RDFS,OWL}),\Omega)$; where:
    - $\mathscr{R}$(RDFS,OWL) is an off-the-shelf OWL reasoner
    - Ω is an ontology based on RDFS+OWL
    - $\wp_{\Omega}$ is an extra application layer
      - it (may) use a general reasoner
      - it "knows" about Ω, and extends reasoning accordingly
  - an application means:
    - define metadata $\mathscr{M}_{\Omega}$, based on Ω
    - use $\wp_{\Omega}$ to deduce facts, answer questions

Other Patterns

- Further alternatives:
  - $\wp_{\Omega}(\mathscr{R}(\text{RDFS},\text{OWL}_{\text{DL}}),\Omega)$
  - $\wp_{\Omega}(\mathscr{R}(\text{RDFS},\text{OWL}_{\text{Lite}}),\Omega)$
  - $\wp_{\Omega}(\mathscr{R}(\text{RDFS}),\Omega)$
- In plain English: one can develop
  - a specialized terminology framework for an application area
    - based or not on OWL
  - a specialized application for this area
  - you are not *oblidged* to use OWL to be a good SW citizen 😉
- Examples: SKOS, CC/PP, RSS, …

The SKOS Example

- "Simple Knowledge Organisation Systems"
- Framework for building, e.g., thesauri, glossaries, …
  - the document refers to "concept schemes"
  - simpler than OWL
  - does not have the same expressive power
  - no strictly defined semantics
  - simpler to build SKOS applications
- Still a draft
  - http://www.w3.org/2001/sw/Europe/reports/thes/1.0/guide/
  - not in the "regular" W3C recommendation track

SKOS example

```
<skos:Concept rdf:about="http:/example.com/Concept/0001">
 <skos:prefLabel>English cuisine</skos:prefLabel>
 <skos:altLabel>English dishes</skos:altLabel>
 <skos:altLabel xml:lang="fr">Cuisine anglaise</skos:altL
 <skos:inScheme rdf:resource="http:/example.com/thesaurus
 <skos:broader rdf:resource="http:/example.com/Concept/00
 <skos:related rdf:resource="http:/example.com/Concept/00
</skos:Concept>
```

Where does the metadata come from?

(should we really expect the author to type in all this metadata?)

It May Be Around Already...

- Part of the metadata information is present in tools...

  ...but thrown away at output 😢
  - e.g., a business chart can be generated by a tool...

    ...it "knows" the structure, the classification, etc. of the chart

    ...but, usually, this information is lost

    ...storing it in metadata would be easy!
- "SW-aware" authoring tools will be of a great help
- Tools can also extract some metadata from HTML files (with little help)
- Ontologies also exist already
  - albeit mostly in propriatery formats
  - but conversion tools exist!

Isn't This Research Only?

(or: does this have *any* industrial relevance whatsoever?)

Not Any More...

- SW has indeed a strong foundation in research results...
- ...but we see more and more companies embracing it!
- Remember:
  1. the Web was born at CERN...
  2. ...was first picked up by high energy physicists...
  3. ...then by academia at large...
  4. ...then by small businesses and start-ups...
  5. "big business" came only later!

  network effect kicked in early...
- Semantic Web is now at #4, and moving to #5!

Lots of Tools

- (Graphical) Editors:
  - IsaViz (Xerox Research/W3C), RDFAuthor (Univ. of Bristol), Protege 2000 (Stanford Univ.), SWOOP (Univ. of Maryland) Orient (IBM)
- Programming Environments:
  - Jena (for Java, includes OWL reasoning), RDFLib (for Python), Redland (in C, with interfaces to Tcl, Java, PHP, Perl,...), SWI-Prolog, IBM's Semantic Toolkit, ...
- Triple based database systems:
  - Kowari, Tucana, RDF Gateway, Sesame
- RDF and OWL validators:
  - http://www.w3.org/RDF/Validator/, http://owl.bbn.com/validator/
- Ontology converter (to OWL):
  - http://www.mindswap.org/2002/owl.html
- The list gets longer every day! You can always start with:
  - http://www.w3.org/RDF/#developers

SW Applications

- Large number of applications emerge
  - some applications use RDF only
  - others begin to use ontologies, too
    - huge number of ontologies exist already, with proprietary formats
    - converting them to RDF/OWL will be a major task

      (but there are converters)
    - but it will be worth it!
- See, for example, on WSIndex
  - portal on "Web Services and Semantic Web Resources"

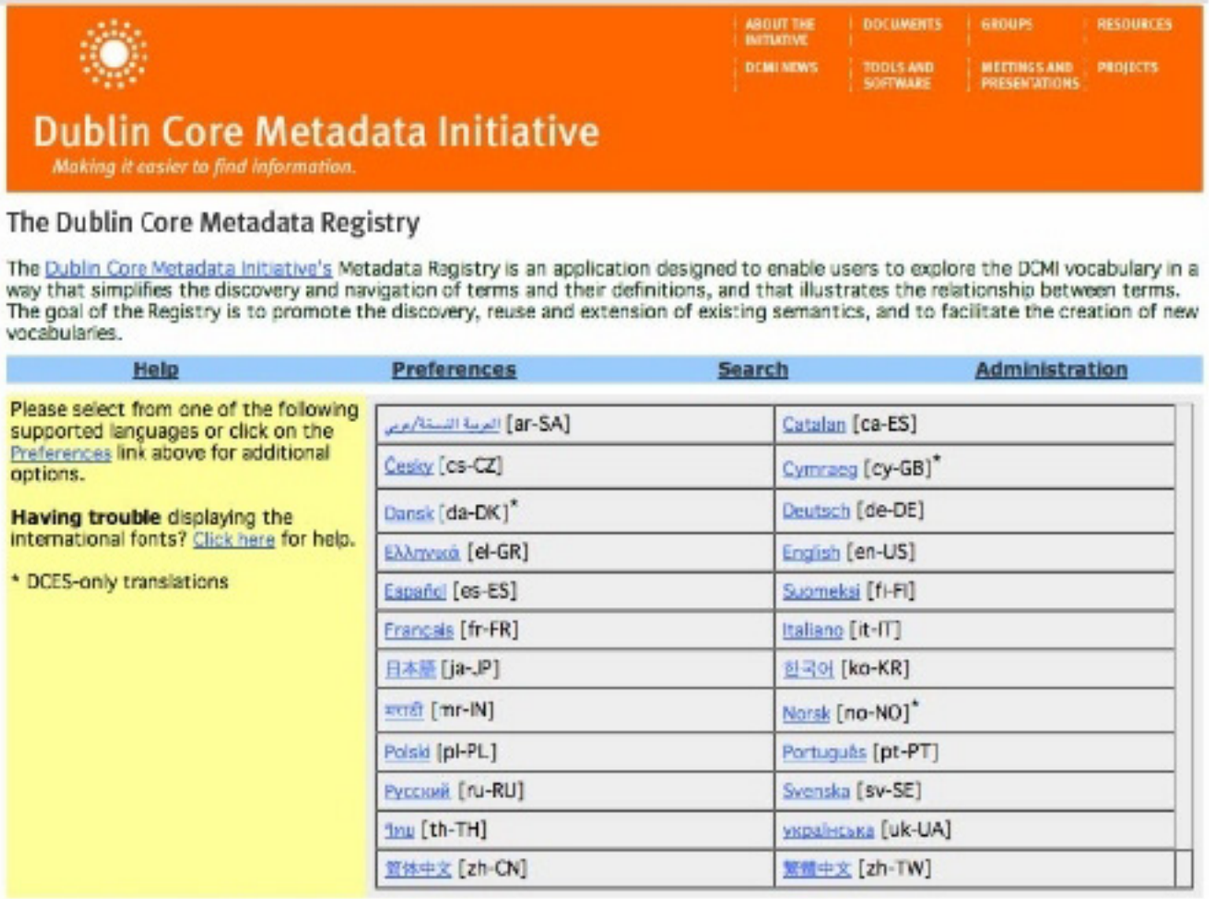
    with a separate page for SW applications

  …or the SWAD-Europe survey
  - lists more than 50 applications in 12 categories…
  - and is already more than a years old!

SW Application Examples

Dublin Core

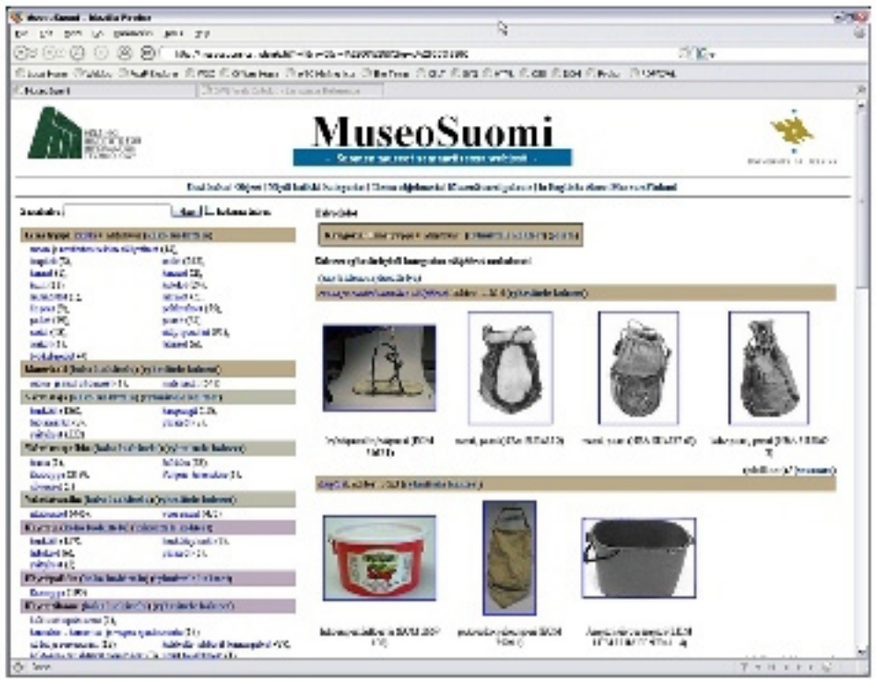
- vocabularies for distributed Digital Libraries
- one of the first metadata vocabularies in RDF
- URI: http://www.dublincore.org
- extensions exist, eg, PRISM that includes digital right tracking



SW Application Examples (cont)

Data integration

- achieve semantic integration of corporate resources or different databases
- RDF/RDFS/OWL based vocabularies as an "interlingua" among system components
- Boeing example: http://www.cs.rutgers.edu/~shklar/www11/final_submissions/paper3.pdf
- similar approaches: Artiste project, MITRE Corp., MuseoSuomi, ...
- there are companies specializing in the area



SW Application Examples (cont)

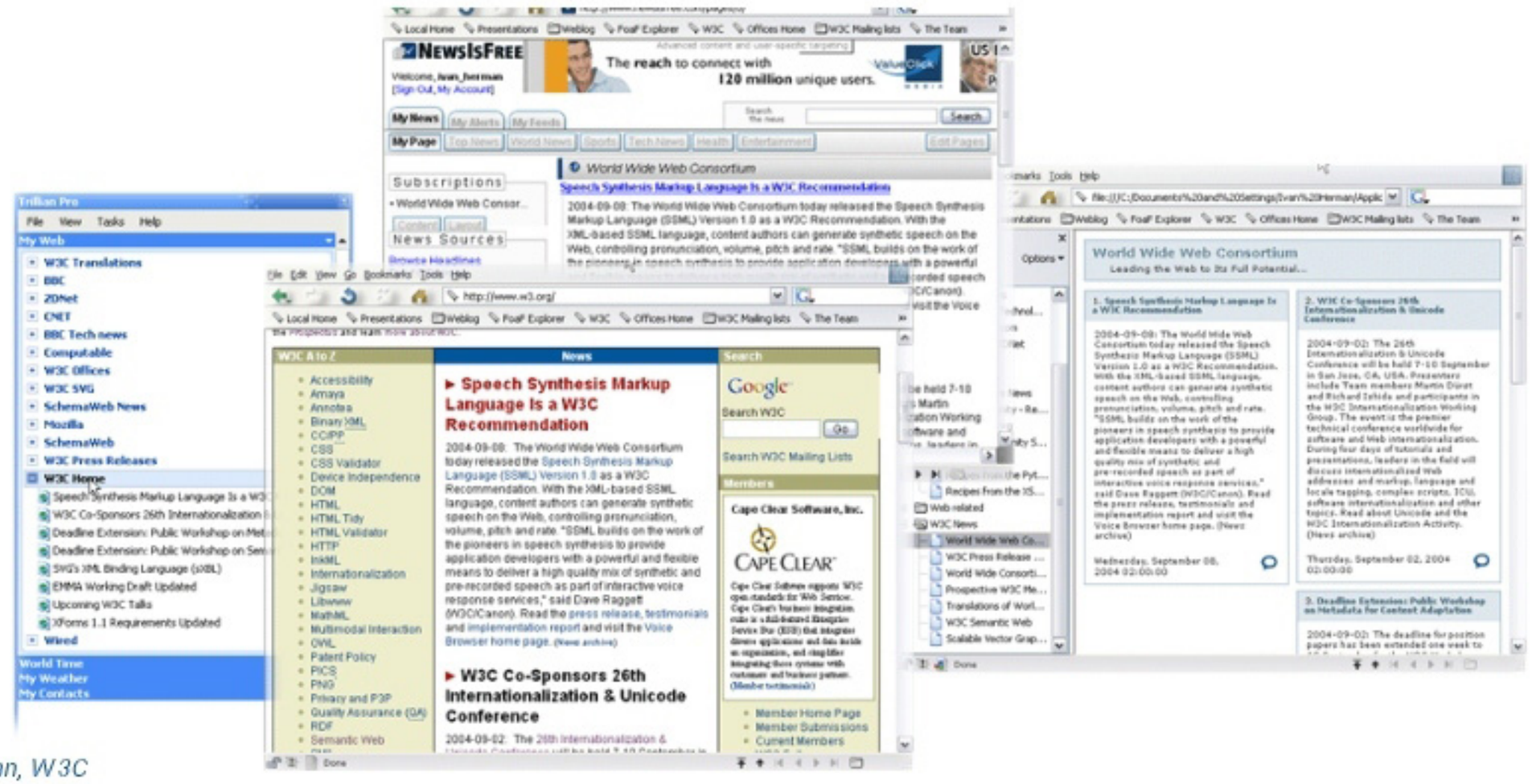
Sun's SwordFish

- Sun provides assisted support for its products, handbooks, etc
- Public queries go through an internal RDF engine for, eg:
  - Sun's White Papers collection (http://www.sun.com/servers/wp.html/)
  - Sun's System Handbooks collection (http://sunsolve.sun.com/handbook_pub/)

SW Application Examples (cont)

Web Content Syndication (RSS)

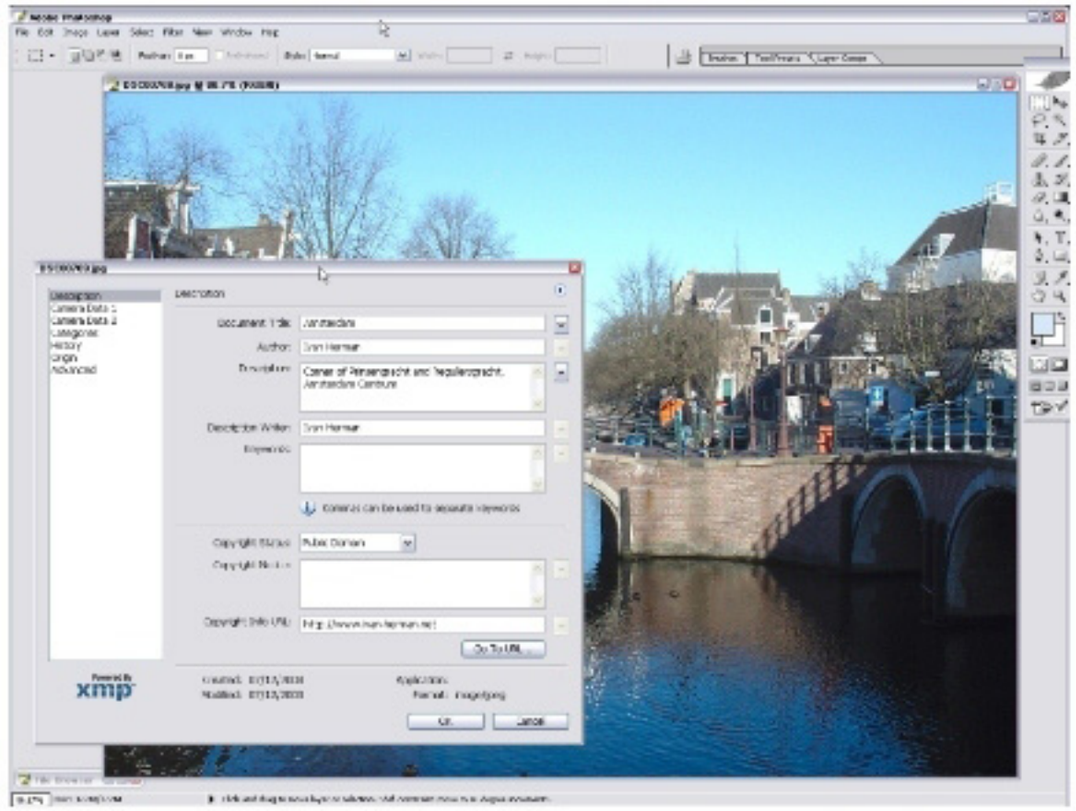
- can be used to specify the *important* content of a page
- there is a Yahoo discussion group and (non-W3C) working group
- URI: http://purl.org/rss/
- widely used in the weblog world!
- example: W3C home page syndicated



SW Application Examples (cont)

XMP

- Adobe's tool to add RDF-based metadata to *all* their file formats
  - eg, Photoshop in Creative Suite
  - millions of people use RDF without knowing it...
- the tool is available for all!
- URI: http://www.adobe.com/products/xmp/main.html



SW Application Examples (cont)

Mozilla

- internal data are stored in RDF (eg, bookmarks, conf. files)
- Mozilla's open directory project stores data in RDF
  - operates with RDF files of more than 55MB…

Creative Commons

- an environment to express rights of digital content on the Web
  - legal constraints referred to in RDF, added to pages
- there are specialized browsers, browser plugins
- more than 1,000,000 users worldwide(!)
  - without knowing that they use RDF…

Brandsoft

- entreprise Web Management
- all business models are stored in RDF
- easy to set up internal rules

Does the SW Replace Web Services?

SW and WS are Complementary

- Two facets of machine-to-machine communication
  - service based ("Web of applications")
  - metadata based ("Web of data")
- *A widely deployed Web Services infrastructure may be the most compelling business case for the Semantic Web*
- *The synergy of Semantic Web and Web Service will hugely benefit for the wide deployement of both!*

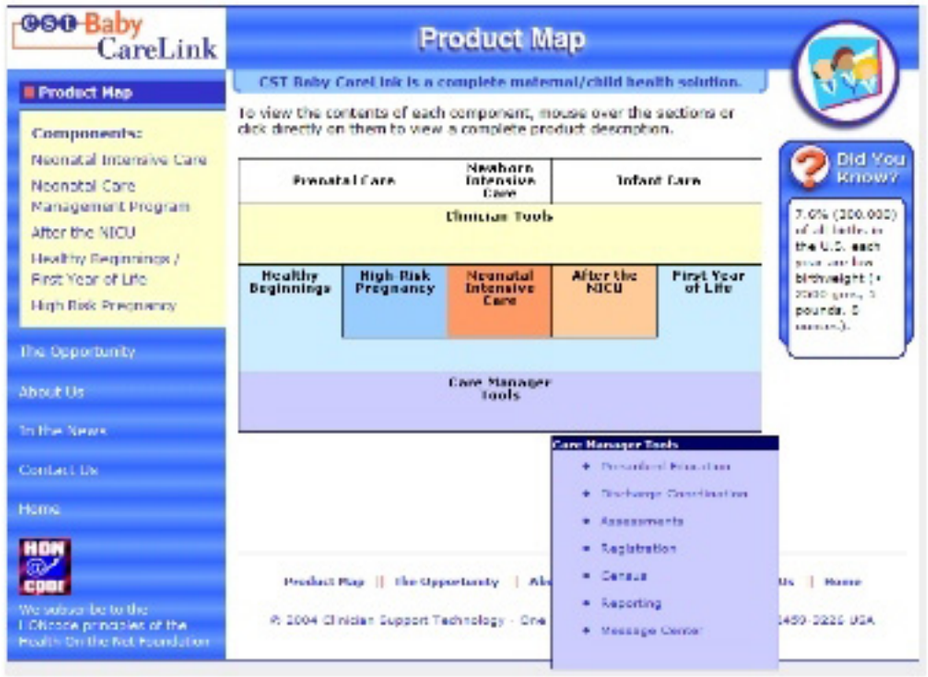
Examples for Potential Synergies

- Semantic Web based search engines for Web Services
  - search based on complex constraints
    - e.g., "find the most elegant Schrödinger equation solver"
- Ontology services
  - "provide a Web Service to make logical deductions on my behalf" (e.g., on complex metadata with an ontology)
    - find and manage equivalences
    - make logical deduction of terms
    - check SW description for validity
    - etc

SW-WS Synergy Example

Baby CareLink

- centre of information for the treatment of premature babies
- provides an OWL service *as a Web Service*
  - combines disparate vocabularies like medical, insurance, etc
  - remember: ontology is hard!
  - users can add new entries to ontologies
  - complex questions can be asked through the service



Convergence (at W3C)

- *Both areas are represented at W3C*
- Lots of discussions on convergence
  - mapping of WSDL2.0 to RDF
  - Web Choreography development in terms of RDF
    - initiatives already exist, e.g., OWL-S (formerly DAML-S)
  - discussions on WS "Features and Properties"
  - there is a "Semantic Web Services" Interest Group

Are we done?

Semantic Web Activity Phase 2

- First phase (completed): core infrastructure
- Second phase: promotion and implementation needs
  - relevant working groups
  - outreach to user communities
    - life sciences
    - geospatial information systems
    - libraries and digital repositories
    - ...
  - intersection of SW with other technologies
    - Semantic Web Services
    - privacy policies
    - ...

"Best Practices" Work

- "Semantic Web Best Practices and Deployment"
  - recommendations for practical deployment
  - engineering guidelines
  - ontology/vocabulary development practices
  - educational material
  - effective demonstrations
  - information on applications
  - etc.
- Goal is to increase awareness on SW
- W3C started work in this area recently
  - some initial drafts are already available

RDF Data Access (a.k.a. Queries)

- In Python, for example, one uses:

```
# do something with (p,o) pairs
for (p,o) in triples.predicate_objects(subject) :
    do_something(p,o)
```

  "predicate_objects" returns a *subgraph*

- Applications may want more
  - i.e., return complex subgraph with parts missing
- Very important for large and *distributed* RDF depositories

Data Access Example

- One may want something like:

```
SELECT ?a, ?b
WHERE (?x 'parent' ?a)
      (?b 'brother' ?x)
```

  (i.e., 'b is the uncle of a')
- W3C started a standardization work in this area in 2004
- Firs draft for a spec published in October 2004:
  - "SPARQL Query Language for RDF"
  - http://www.w3.org/TR/rdf-sparql-query/
- Similar facilities already implemented in Jena, RAP,...

Further SPARQL Features

- Constraining Values

```
SELECT ?a, ?b
WHERE (?x 'parent' ?a)
      (?b 'brother' ?x) AND age(?b) < 30
```

("age" is an application specific function)

- Optional Constraints:

```
SELECT ?a, ?b, ?c
WHERE    (?x 'parent' ?a)
         (?b 'brother' ?x)
OPTIONAL (?b 'age' ?c)
```

(return age if known)

- *This is still a draft!*

Rules

- OWL can be used for simple inferences
- Applications may require other type of rules, e.g.:
  - (prem-1 ∧ prem-2 ∧ ...) ⇒ (concl-1 ∧ concl-2 ∧ ...)
  - e.g.: for *any* «X», «Y» and «Z»:

    "if «Y» is a parent of «X», and «Z» is a brother of «Y»

    then «Z» is the uncle of «X»"
  - using a logic formalism:

    ∀x,z: ((∃y: (y parent x) ∧ (y brother z)) ⇒ (z uncle x))

    (Horn clauses)
- Lots of research is happening to extend RDF/OWL (Metalog, RuleML, SWRL, cwm, ...)
- W3C *may* initiate a standardization work in this area, too
  - question is whether results are "ripe" for standardization
  - and whether the necessary manpower is available

Trust

- Can I trust a metadata on the Web?
  - is the author the one who claims he/she is?
  - can I check the credentials?
  - can I trust the inference engine?
  - what about IPR of the metadata?
  - etc.
- Some of the basic building blocks are available:
  - XML Signature/Encryption
  - XML based Key Management is in preparation
- Much is missing, e.g.:
  - a "canonical" form of RDF/XML
    - necessary for unambiguous signatures
  - exhaustive tests for inference engines
  - protocols to check, for example, a signature
- It is on the "future" stack of W3C...

What can I read to understand?

Some Books

- M. Dertouzos: The Unfinished Revolution (1995)
  - an early "vision" book (not only on the Semantic Web)
- T. Berners-Lee: Weaving the Web (1999)
  - another "vision" book
- J. Davies, D. Fensel, F. van Harmelen: Towards the Semantic Web (2002)
- S. Powers: Practical RDF (2003)
- D. Fensel, J. Hendler: Spinning the Semantic Web (2003)
- G. Antoniu, F. van Harmelen: Semantic Web Primer (2004)
- …

Further Information

- Bristol University
  - http://www.ilrt.bristol.ac.uk/discovery/rdf/resources/
  - huge list of documents, publications
- Semantic Web Community Portal
  - http://www.semanticweb.org/
  - "Business model IG" (part of the portal)
  - huge set of links to documents, software, ...
- SemWeb Central
  - http://semwebcentral.org
  - Open Source development archive
- WSIndex
  - http://www.wsindex.org/Companies/Semantic_Web/index.html
  - portal on "Web Services and Semantic Web Resources"
- SWBP&D Working Group Tutorial list
  - http://www.w3.org/2001/sw/BestPractices/Tutorials

W3C Primers and Test Cases

RDF Primer

URI: http://www.w3.org/TR/rdf-primer

OWL Guide

URI: http://www.w3.org/TR/owl-guide/

RDF Test Cases

URI: http://www.w3.org/TR/rdf-testcases/

OWL Test Cases

URI: http://www.w3.org/TR/owl-test/

Public Fora at W3C

Semantic Web Interest Group

a forum for discussions on applications

URI: http://www.w3.org/RDF/Interest

RDF Logic

public (archived) mailing list for technical discussions

URI: http://lists.w3.org/Archives/Public/www-rdf-logic/

Further Information

These slides (with links)

http://www.w3.org/2004/Talks/0212-Brisbane-IH/

Australian Office of W3C

http://w3c.dstc.edu.au/

Mail me:

ivan@w3.org

Now For Real...

Other Questions?