State of the Semantic Web

Ivan Herman, W3C

(Last update: 26 November 2008)

So where are we with the Semantic Web?

We have the basic technologies

- Stable specifications for the basics since 2004: RDF, OWL
- Work is being done to properly incorporate rules
- We have a standard for query since 2008: SPARQL
- We have some additional technologies to access/create RDF data: GRDDL, RDFa, POWDER, …
- Some fundamental vocabularies became pervasive (FOAF, Dublin Core,…)

Lots of Tools (*not* an exhaustive list!)

- Categories:
  - Triple Stores
  - Inference engines
  - Converters
  - Search engines
  - Middleware
  - CMS
  - Semantic Web browsers
  - Development environments
  - Semantic Wikis
  - …
- Some names:
  - Jena, AllegroGraph, Mulgara, Sesame, flickurl, …
  - TopBraid Suite, Virtuoso environment, Falcon, Drupal 7, Redland, Pellet, …
  - Disco, Oracle 11g, RacerPro, IODT, Ontobroker, OWLIM, Tallis Platform, …
  - RDF Gateway, RDFLib, Open Anzo, DartGrid, Zitgist, Ontotext, Protégé, …
  - Thetus publisher, SemanticWorks, SWI-Prolog, RDFStore…
  - …

Lots of tools (cont.)

- Significant speed, store capacity, etc; improvements are reported every day
- Some of the tools are open source, some are not; some are very mature, some are not: *it is the usual picture of software tools*, nothing special any more!
- We still need more “middleware” tools to properly *combine* what is already available…
- *Anybody can start developing RDF-based applications today*

Great community…

computas

Some good lessons

- New standards (e.g. SPARQL), proposals for standardization (e.g. SPARUL), new tools (e.g. Jena), open source (e.g. Tomcat, Apache), lack of good documentation all say **high risk!!!!**
- **However, the support and maintenance from the W3C community and open source developers (e.g. Jena team) has been impressive, the support through IRC channels, mailing lists etc has been invaluable for the project.**

Slide 19

From a presentation given by David Norheim, Computas AS, at the ESTC2008 Conference, Vienna, Austria

There is a great community

- There are *lots* of tutorials, overviews, and books around
  - again, some of them good, some of them bad, just as with any other areas…
- Active developers' communities
  - blogs, IRC channels, mailing lists, various fora: more than what one person can oversee…
- Some measures claim that there are over 10^7 Semantic Web documents on the Web

Some deployment communities

- Major communities pick the technology up: digital libraries, defense, eGovernment, energy sector, financial services, health care, oil and gas industry, life sciences …
  - Health care and life science sector is now very active
    - also at W3C, in the form of an Interest Group
- Semantic Web also appears in the "Web 2.0/Web 3.0" world (whatever that means)
  - exchange of social data
  - personal "space" applications
  - multimedia asset management (video, photos, audio, …)
  - etc

So what *is* the Semantic Web?

- There is a growing number of application patterns referring to the Semantic Web:
  - data integration using RDF, SKOS, OWL, …
  - knowledge engineering with complex ontologies
    - using, eg, OWL and/or rule based reasoning
  - better data management, archiving, cataloging, etc
    - eg, digital library applications
  - managing, coordinating, combining Web services
  - intelligent software agents
  - improving search (usually using domain specific vocabularies…)
  - etc
- But: what binds these all together?

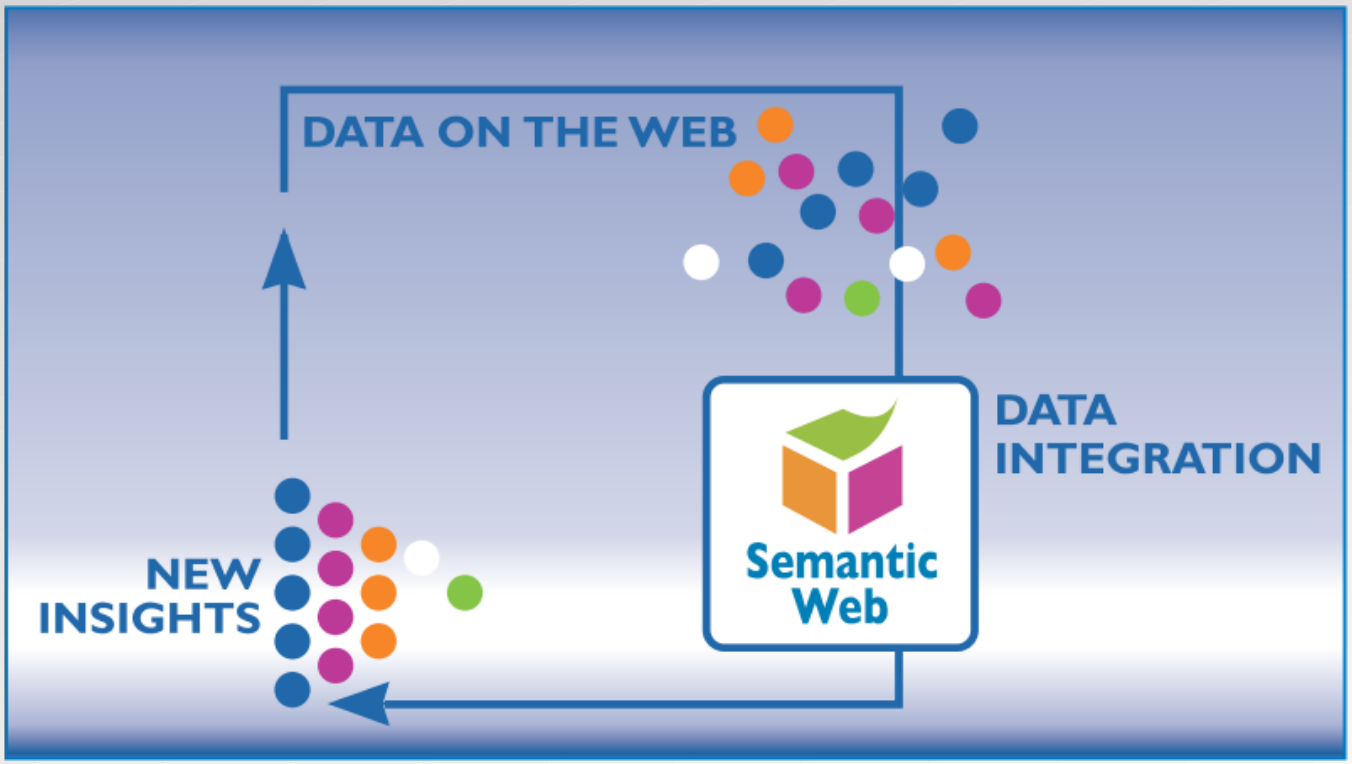
Is this where we are?

So what *is* the Semantic Web? (cont.)

- Maybe, but being an elephant is not necessary bad!
  - it shows that the Semantic Web is a mature technology
  - that there is lots of interest, applications
  - various application areas pick what they need…
    - e.g., some need sophisticated knowledge management, so they go for complex ontologies…
    - some concentrate on semantically simpler vocabularies but large volume of data
  - …and that is fine, there is room for many!

So what *is* the Semantic Web? (cont.)

- But it is good to (re-)emphasize some principles
- The Semantic Web:
  - extends the principles of the Web from documents to data; create a *Web of data*



So what *is* the Semantic Web? (cont.)

- It is the Semantic *Web*, and not only Semantics!
  - Data, ontologies, vocabularies, etc, can (and should!) be shared, reused, potentially on Web scale
    - *the "network effect" on data*
  - one can use the Web infrastructure to denote "things"…
    - Eg: `http://www.ivan-herman/me` denotes, well, *me* (not my home page, not my foaf file, but *me*!)
  - … and add relationships for those, too!

A few words about “newer” technologies

Querying RDF: SPARQL

- Querying RDF graphs is essential (can you imagine Relational Databases without SQL?)
- SPARQL is
  - a query language based on graph patterns
  - a protocol layer to use SPARQL over, eg, HTTP
  - an XML return format for the query results
- Is a W3C Standard (since January 2008)
- Numerous implementations are already available (eg, built in triple stores)

SPARQL (cont.)

- There are also SPARQL “endpoints” on the Web:
  - send a query and a reference to data over HTTP GET, receive the result in XML or JSON
    - some of those can be easily installed on any machine
  - big datasets often offer “SPARQL endpoints” to their local data
  - *applications may not need any direct RDF programming any more, just use a SPARQL processor*
- SPARQL can also be used to *construct* graphs!

The power of CONSTRUCT

```
CONSTRUCT {
    <http://dbpedia.org/resource/Amitav_Ghosh> ?p1 ?o1.
    ?s2 ?p2 <http://dbpedia.org/resource/Amitav_Ghosh>.
}
WHERE {
    <http://dbpedia.org/resource/Amitav_Ghosh> ?p1 ?o1.
    ?s2 ?p2 <http://dbpedia.org/resource/Amitav_Ghosh>.
}
```

- SPARQL endpoint
- returns RDF/XML


```
SELECT *
FROM <http://dbpedia.org/sparql/?query=CONSTRUCT+%7B++...>
WHERE {
  ?author_of dbpedia:author res:Amitav_Ghosh.
  res:Amitav_Ghosh  dbpedia:reference ?homepage;
                    rdf:type          ?type;
                    foaf:name         ?foaf_name.
  FILTER regex(str(?type),"foaf")
}
```

- Data reused in a query elsewhere…

A word of warning on SPARQL…

- Some features are missing
  - control and/or description on the entailment regimes of the triple store (RDFS? OWL-DL? OWL-Lite? …)
  - modify the triple store
  - querying collections or containers may be complicated
  - no functions for sum, average, min, max, …
  - ways of aggregating queries
  - …
- Delayed for a next version…

But: how do you get the data on the SW?

Public datasets are accumulating

- IgentaConnect bibliographic metadata storage: over 200 million triplets
- RDFS/OWL Representation of WordNet: also downloadable as 150MB of RDF/XML
- "Département/canton/commune" structure of France published by the French Statistical Institute
- Geonames Ontology and Data: 6 million (and growing) geographical features
- "dbpedia": infobox data of Wikipedia into RDF
- Note the "Billion Triple Challenge 2008"!

How to get RDF data?

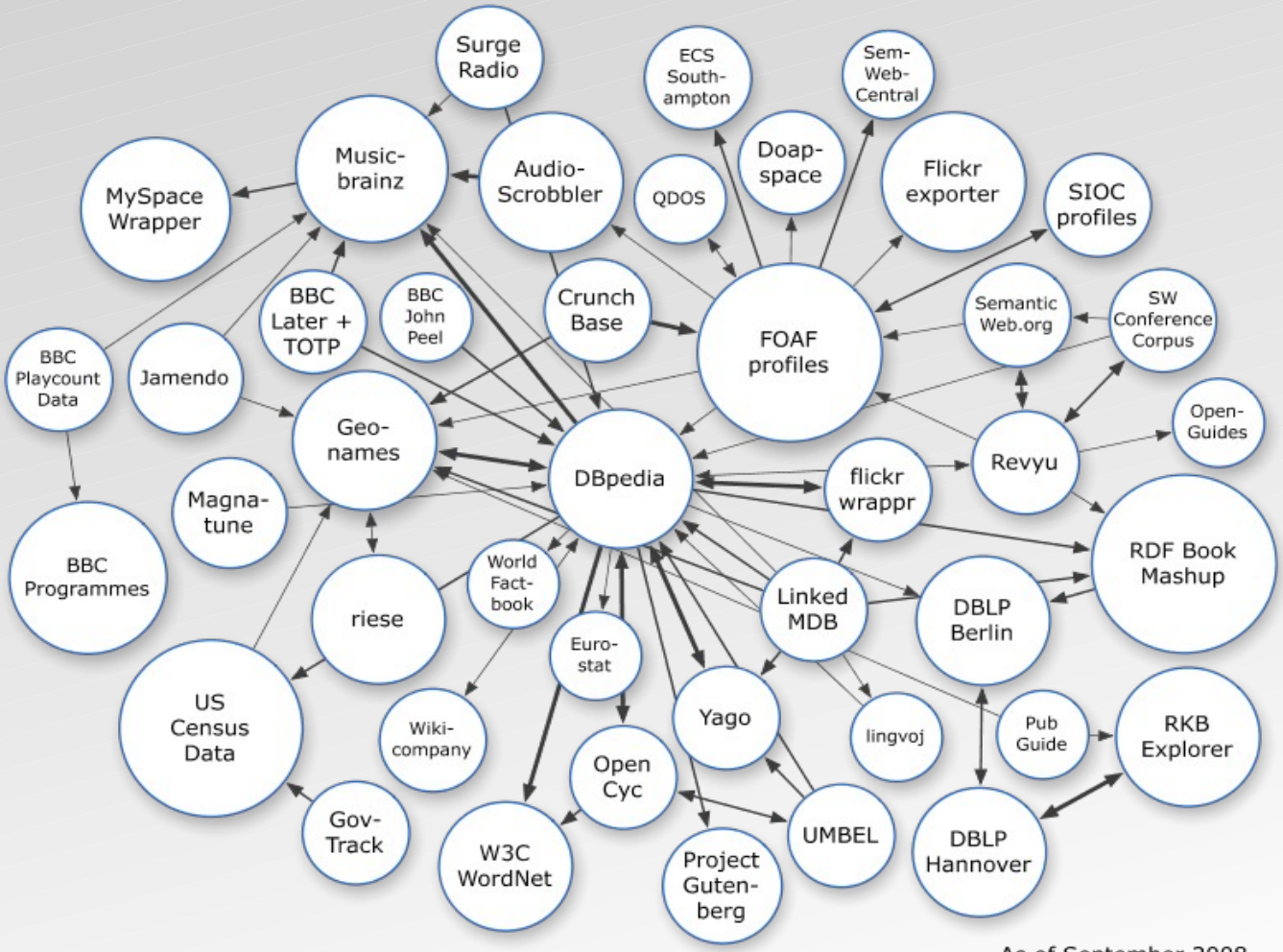
- Of course, one could create RDF data manually…
- … but that is unrealistic on a large scale
- Goal is to generate RDF data automatically when possible and “fill in” by hand only when necessary
- Various data formats should be considered
  - databases (relational or otherwise)
  - data in XML, HTML, in pictures, videos, etc
- Details of the process is still subject of very active R&D!

Bridge to relational databases

- Huge amount of data are stored in (relational) databases
  - "RDFying" them is impossible
- "Bridges" are being defined:
  - a layer between RDF and the relational data
    - RDB tables are "mapped" to RDF graphs, possibly on the fly
  - a number of systems can be used as database as well as triple stores (eg, Oracle, OpenLink, …)
- Work for a survey on mapping techniques has just started at W3C
- SPARQL is becoming the tool of choice to query the data (via SPARQL "endpoints")

Linking Open Data Project

- Goal: “expose” open datasets in RDF
- *Set RDF links among the data items* from different datasets
- Set up SPARQL endpoints
- Billions triples, millions of “links”



Example data source: DBpedia

- DBpedia is a community effort to
  - extract structured (“infobox”) information from Wikipedia
  - provide a SPARQL endpoint to the dataset
  - interlink the DBpedia dataset with other datasets on the Web

Extracting structured data from Wikipedia

Amsterdam

The Keizersgracht at dusk

Location of Amsterdam

Coordinates: 52°22'23"N 4°53'32"E

Country	Netherlands
Province	North Holland
Government	
- Type	Municipality
- Mayor	Job Cohen[1] (PvdA)
- Aldermen	Lodewijk Asscher Carolien Gehrels Tjeerd Herrema Maarten van Poelgeest Marijke Vos
- Secretary	Erik Gerritsen
Area [2][3]	
- City	219 km² (84.6 sq mi)
- Land	166 km² (64.1 sq mi)
- Water	53 km² (20.5 sq mi)
- Urban	1,003 km² (387.3 sq mi)
- Metro	1,815 km² (700.8 sq mi)
Elevation [4]	2 m (7 ft)
Population (1 October 2008)[5][6]	
- City	755,269
- Density	4,459/km² (11,548.8/sq mi)
- Urban	1,364,422
- Metro	2,158,372
- Demonym	Amsterdammer
Time zone	CET (UTC+1)
- Summer (DST)	CEST (UTC+2)
Postcodes	1011 – 1109
Area code(s)	020

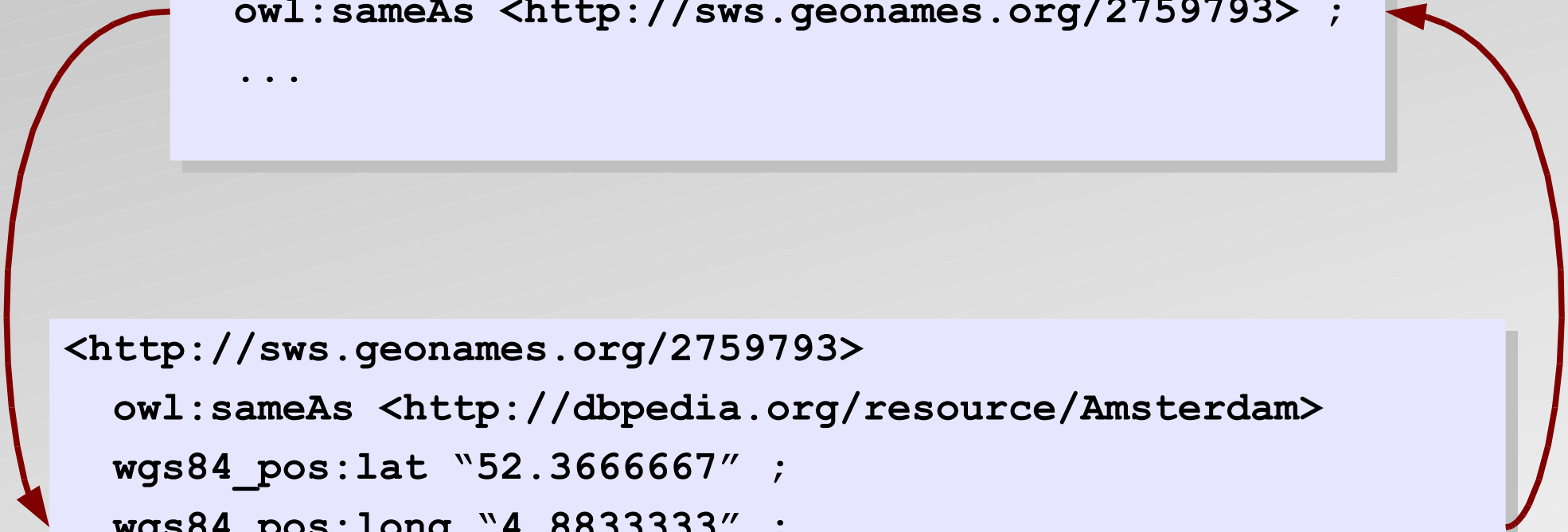
Website: www.amsterdam.nl

```
@prefix dbpedia <http://dbpedia.org/resource/>.
@prefix dbterm  <http://dbpedia.org/property/>.

dbpedia:Amsterdam
   dbterm:officialName "Amsterdam" ;
   dbterm:longd "4" ;
   dbterm:longm "53" ;
   dbterm:longs "32" ;
   ...
   dbterm:leaderTitle "Mayor" ;
   dbterm:leaderName dbpedia:Job_Cohen ;
   ...
   dbterm:areaTotalKm "219" ;
   ...
dbpedia:ABN_AMRO
   dbterm:location dbpedia:Amsterdam ;
   ...
```

Automatic links among open datasets

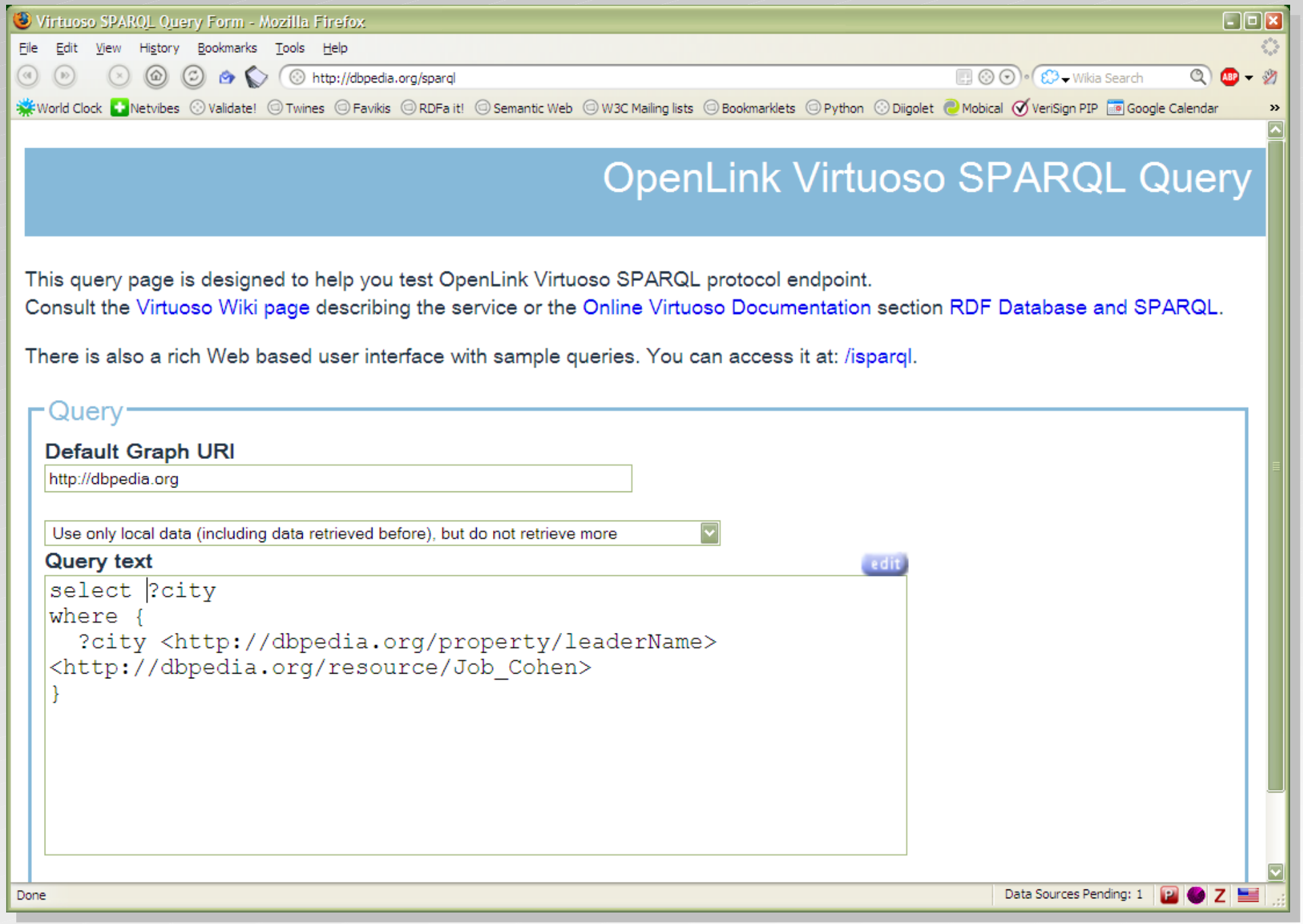
```
<http://dbpedia.org/resource/Amsterdam>
  owl:sameAs <http://rdf.freebase.com/ns/...> ;
  owl:sameAs <http://sws.geonames.org/2759793> ;
  ...
```



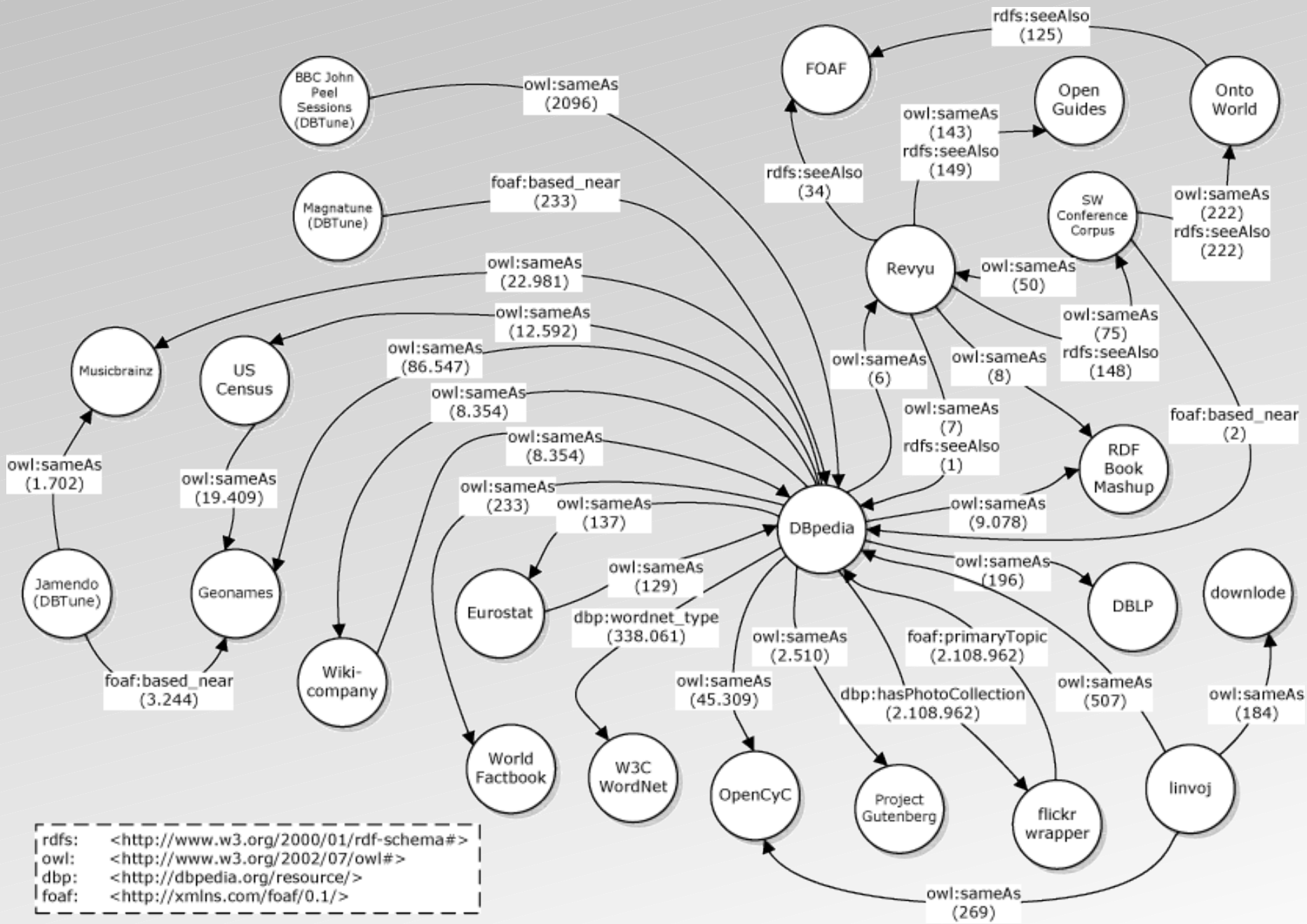
```
<http://sws.geonames.org/2759793>
  owl:sameAs <http://dbpedia.org/resource/Amsterdam>
  wgs84_pos:lat "52.3666667" ;
  wgs84_pos:long "4.8833333" ;
  geo:inCountry <http://www.geonames.org/countries/#NL> ;
 ...
```

Processors can switch automatically from one to the other…

SPARQL-ing DBpedia



The links…



Courtesy of Michael Hausenblas, Wolfgang Halb, Yves Raimond, and Tom Heath

Linking Open Data Project (cont)

- This is a *major* community project
  - anybody can participate; to subscribe to the list:
    - http://lists.w3.org/Archives/public/public-lod/
  - or look at the project site:
    - http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData
  - if you know of open data sets: contact the project to incorporate it with the rest!
- Applications using this set of data in real-life setting should come to the fore soon

Data may be around already…

- Part of the (meta)data 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 web data would be easy!
- "SW-aware" tools are around (even if you do not know it…), though more would be good:
  - Photoshop CS stores metadata in RDF in, say, jpg files (using XMP)
  - …

Generate (meta)data from unstructured data

- An emerging approach:
  - use Natural Language Processing (NLP) to analyse text
  - generate “meta” data (eg, tags) automatically
  - some systems do it behind the scenes (eg, Twine)
  - some offer public web services (eg, Reuter’s Open Calais and Tagaroo, MOAT, Zemanta)

Generate (meta)data from unstructured data

- Not all these systems are geared directly at Semantic Web
- But they can be used to define some sort of controlled vocabulary (ontology, taxonomy, etc) without being heavy on users

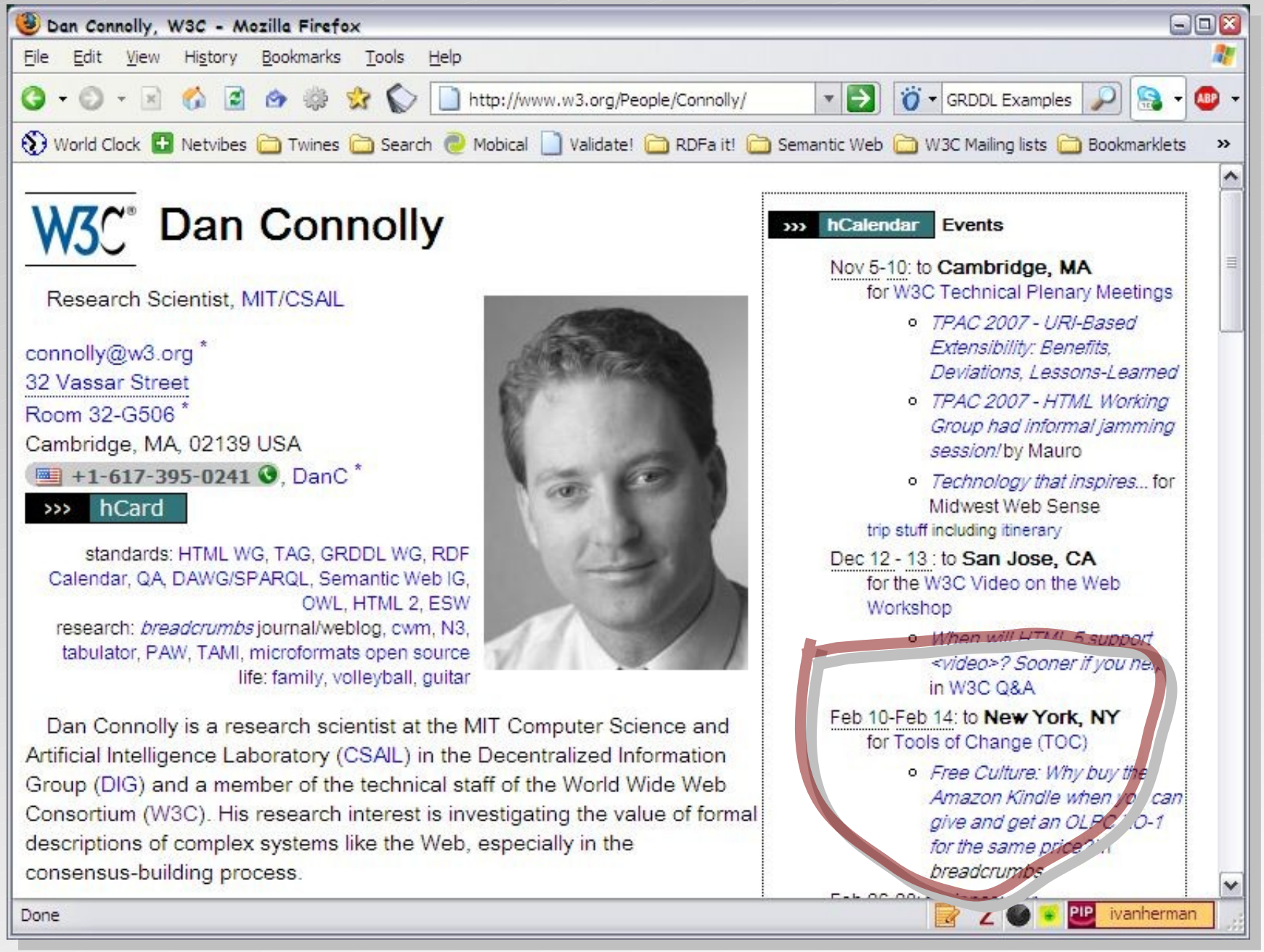
Data may be extracted (a.k.a. “scraped”)

- Different tools, services, etc, come around:
  - get RDF data associated with images, for example:
    - service to get RDF from flickr images
    - service to get RDF from XMP
  - scripts to convert spreadsheets to RDF
  - etc
- Many of these tools are still individual “hacks”, but show a general tendency
- Hopefully more tools will emerge
  - there is a separate wiki page collecting references to existing ones

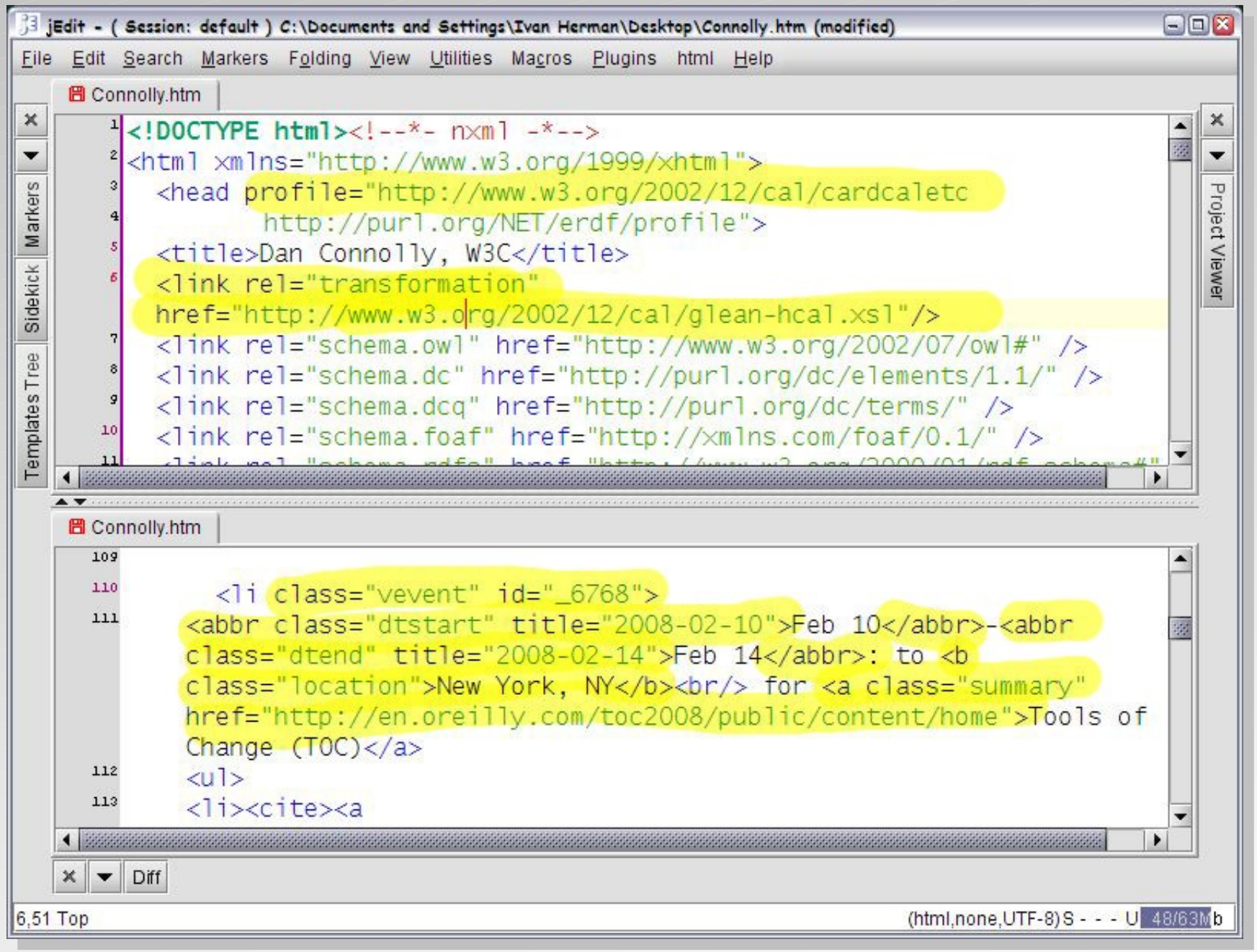
Getting structured data to RDF: GRDDL

- GRDDL is a way to access structured data in XML/ XHTML and turn it into RDF:
  - defines XML attributes to bind a suitable script to transform (part of) the data into RDF
    - script is usually XSLT but not necessarily
    - has a variant for XHTML
  - a “GRDDL Processor” runs the script and produces RDF on–the–fly
- A way to access existing structured data and “bring” it to RDF
  - eg, a possible link to microformats
  - exposing data from large XML use bases, like XBRL

GRDDL example: Dan's homepage…



… behind the scenes …



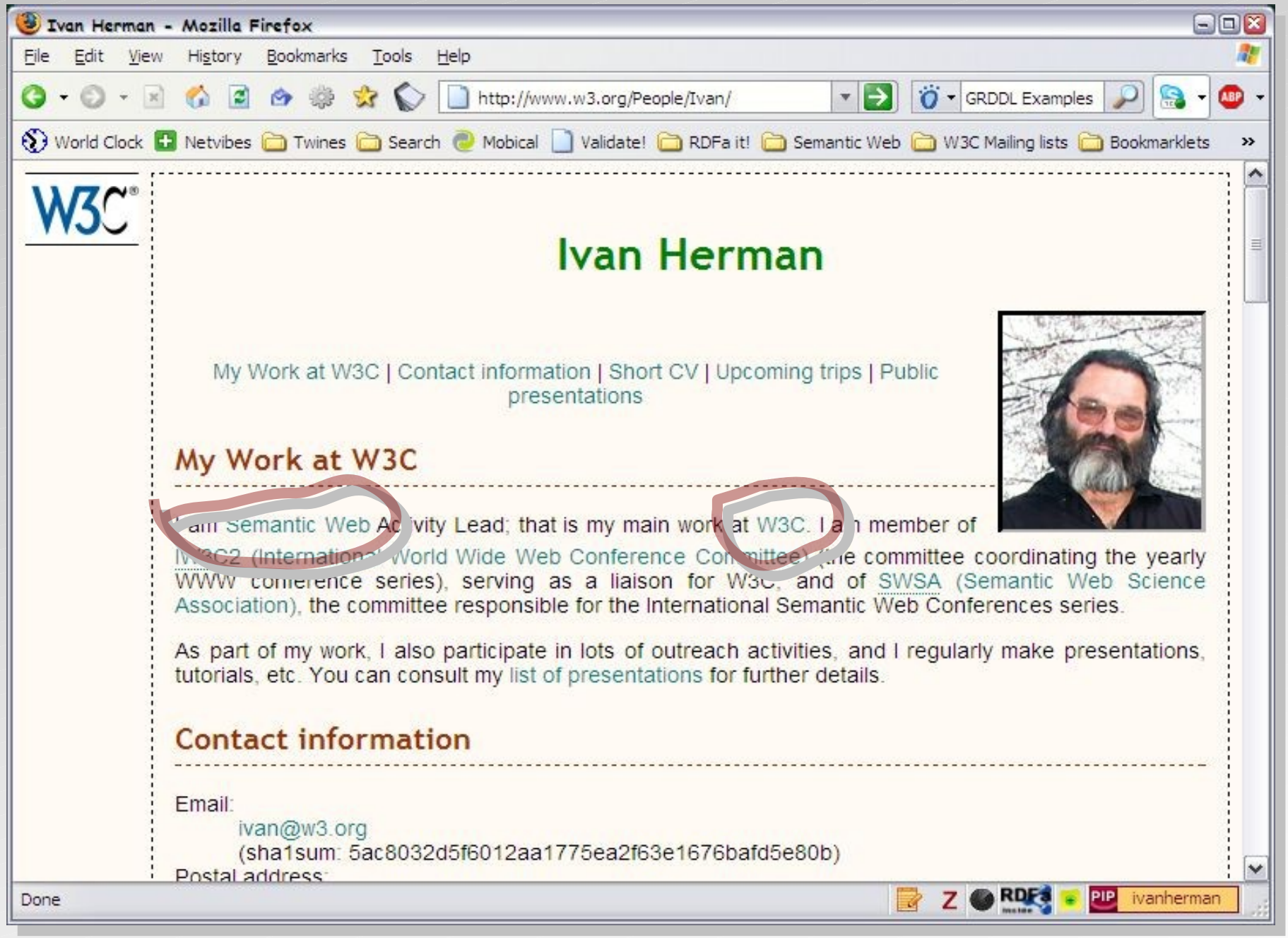
…yielding, through the GRDDL transform

```
<http://www.w3.org/People/Connolly/#_6768>
    a <http://www.w3.org/2002/12/cal/icaltzd#Vevent>;
    <http://www.w3.org/2002/12/cal/icaltzd#organizer>
          <http://www.w3.org/People/Connolly/#me>;
    <http://www.w3.org/2002/12/cal/icaltzd#summary>
          "Tools of Change (TOC)";
    <http://www.w3.org/2002/12/cal/icaltzd#dtstart>
          "2008-02-10"^^<http://www.w3.org/2001/XMLSchema#date>;
    <http://www.w3.org/2002/12/cal/icaltzd#dtend>
          "2008-02-14"^^<http://www.w3.org/2001/XMLSchema#date>;
    <http://www.w3.org/2002/12/cal/icaltzd#url>
          <http://www.w3.org/People/Connolly/#_6768>;
    <http://www.w3.org/2002/12/cal/icaltzd#location>
          "New York, NY" .
```

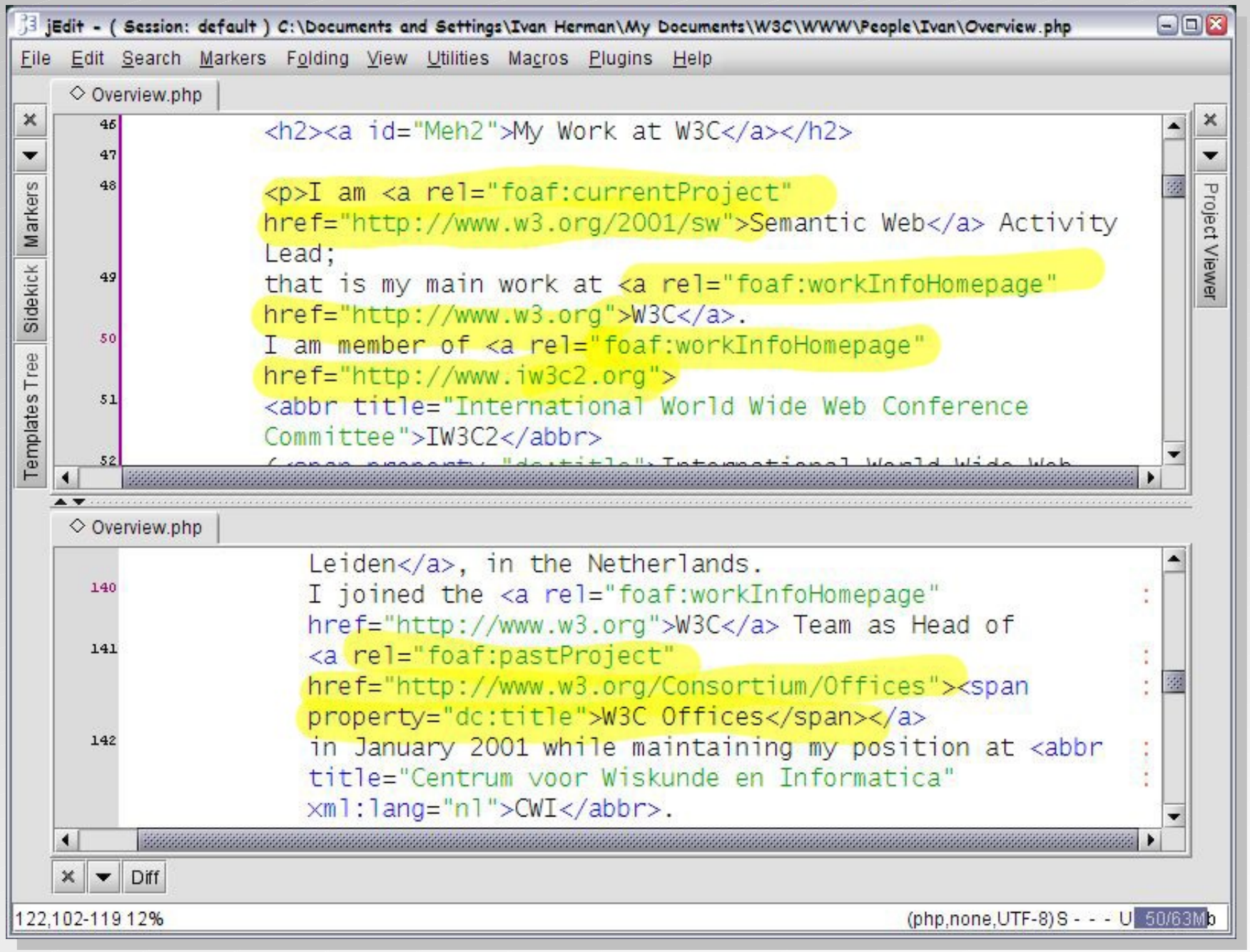
Getting structured data to RDF: RDFa

- RDFa extends XHTML with a set of attributes to include structured data into XHTML
- Makes it easy to “bring” existing RDF vocabularies into XHTML
  - uses namespaces for an easy mix of terminologies
- It can also be used with GRDDL
  - but: no need to implement a separate transformation per vocabulary

RDFa example: Ivan's homepage…



… behind the scenes …



…yielding, by extracting RDF from XHTML

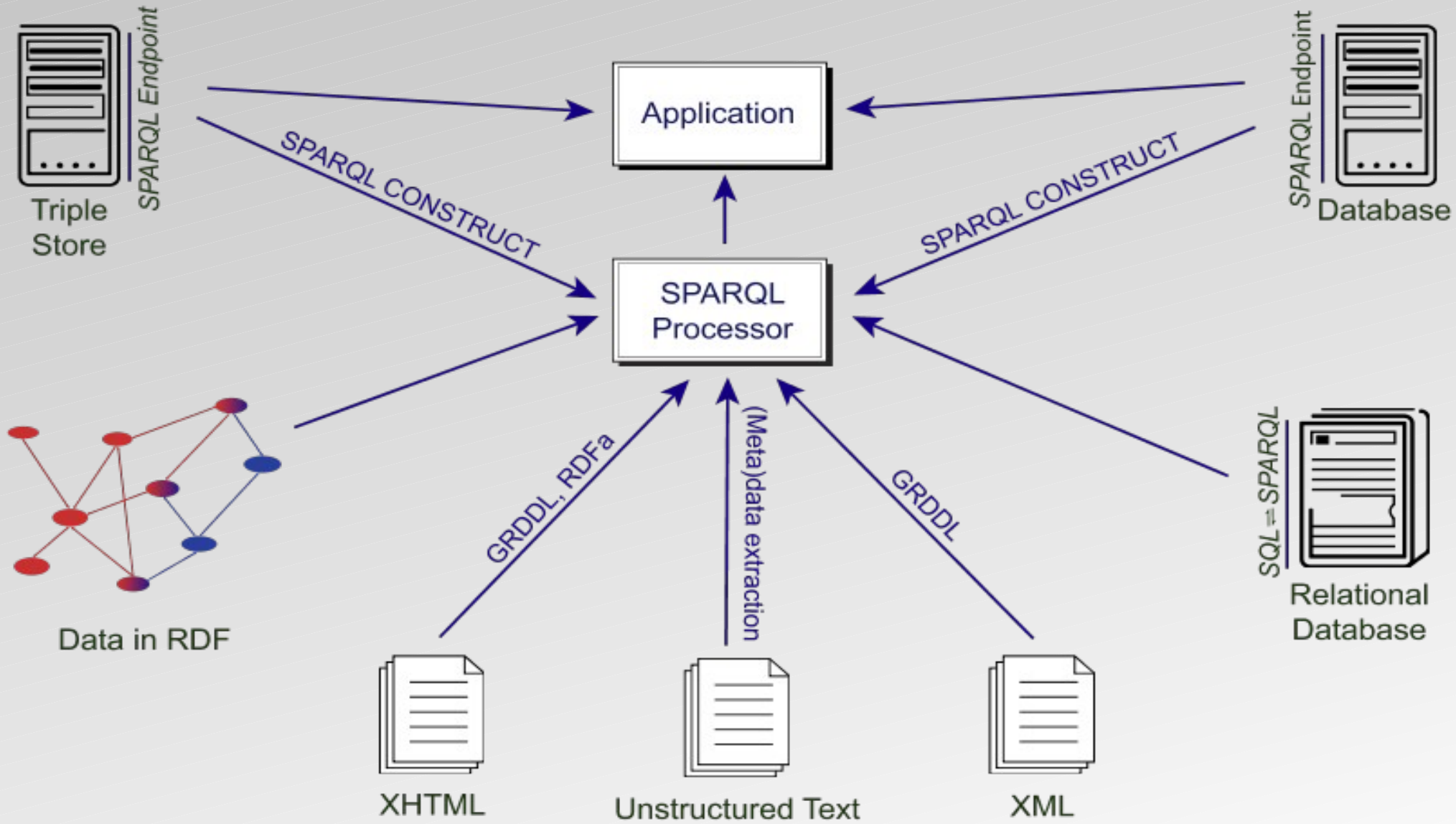
```
<Overview.html#me> a foaf:Person;
    foaf:currentProject <http://www.w3.org/2001/sw>;
    foaf:workInfoHomepage
         <http://www.iw3c2.org>,
         <http://www.w3.org>;
    foaf:pastProject
         <http://www.w3.org/Consortium/Offices>;
      ...

<http://www.w3.org/Consortium/Offices> dc:title "W3C Offices".
```

Such data can be SPARQL-ed

```
SELECT DISTINCT ?name ?home ?orgRole ?orgName ?orgHome
# Get RDFa from my home page:
FROM <http://www.w3.org/People/Ivan/>
# GRDDL-ing http://www.w3.org/Member/Mail:
FROM <http://www.w3.org/Member/Mail/>
WHERE {
?foafPerson  foaf:mbox ?mail;
             foaf:homepage ?home.
?individual  contact:mailbox ?mail;
             contact:fullName ?name.
?orgUnit ?orgRole ?individual;
          org:name ?orgName;
          contact:homePage ?orgHome.
}
```

SPARQL as a unifying point!



How to “assign” RDF data to resources?

- This is important when the RDF data is used as “metadata”
- Some examples:
  - copyright information for your photographs
  - is a Web page usable on a mobile phone and how?
  - bibliographical data for a publication
  - annotation of the data resulting from a scientific experiment
  - etc
- The issue: if I have the URI of the resource (photograph, publication, etc), how do I find the relevant RDF data?

The data might be embedded

- Some data formats allow the direct inclusion of (RDF) metadata:
  - SVG (Scalable Vector Graphics)
  - XHTML+RDFa
  - microformats+GRDDL
  - JPG files using the comment area and, eg, Adobe's XMP technology
- That can include all the information, or link to further data

Simple linkage

- Some formats have special link statements. Eg, in (X)HTML:

```
<html>
  <head>
    <link rel="meta" href="meta.rdf"/>
 ...
```

- Similar facilities might exist in other formats (eg, SMIL)

POWDER

- POWDER (Protocol for Web Description Resources) provides for more elaborate scenarios
- With POWDER:
  1. define a *set* of resources by constraints on the URIs; eg
     - URIs must begin with **`http://www.example.com/bla/`**
     - the port number in the URI-s should be **`XYZW`**
  2. define description resources, binding each resource in the set to additional information
     - this *must* be attributed and may be open to authentication
  3. get such description resources, eg, via a link statements, via HTTP, via SPARQL from a depository, …

A POWDER scenario: copyright for photos

http://www.ex.org/index

GET .../index

Return Descr. Res. URI

cc:license <http://cp.../>

for resources:

http://www.ex2.org/img/*

GET Descr. Res.

Client

GET http://.../img/imgXXX.jpg

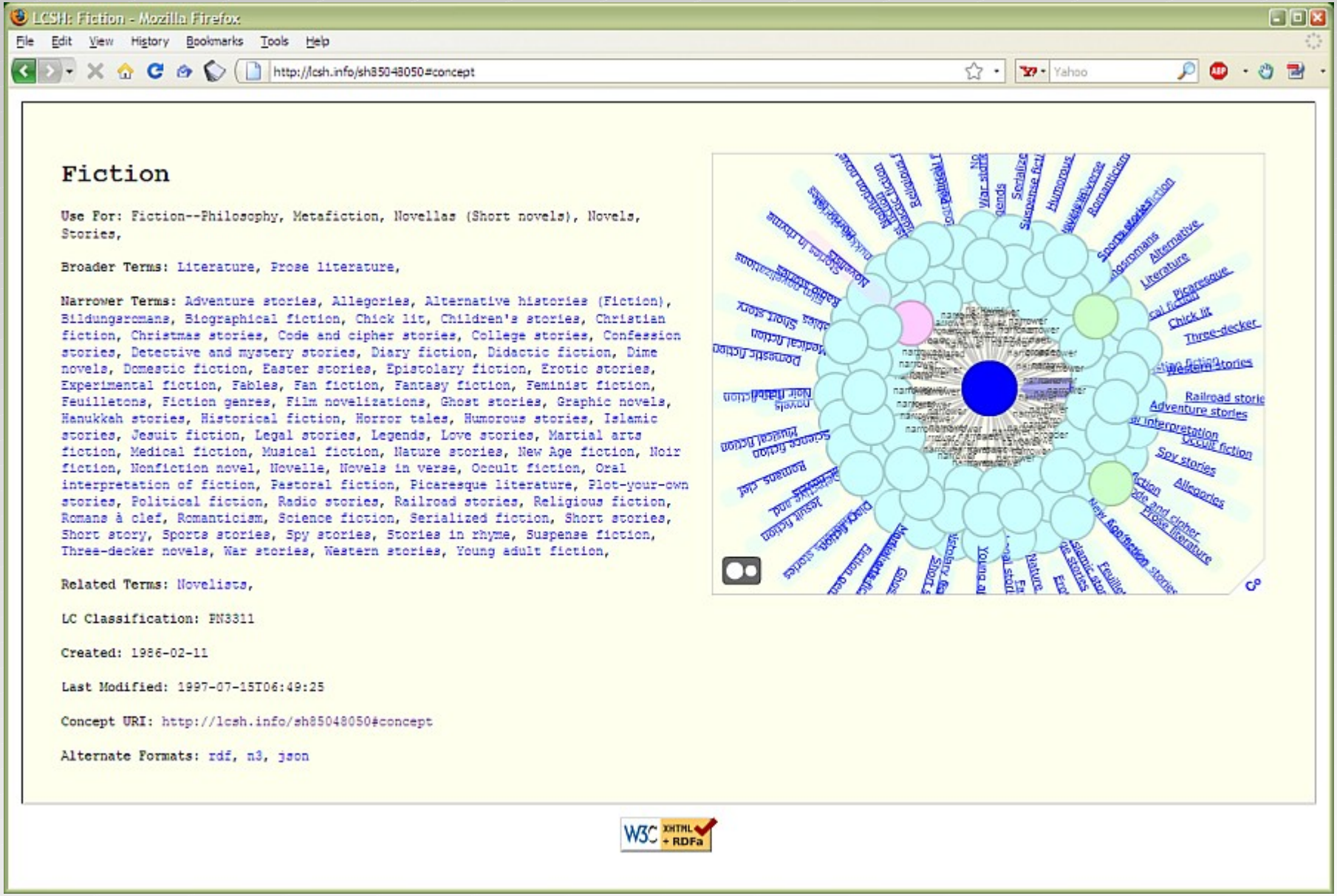
<http://.../img/imgXXX.jpg> cc:license <http://cp../>.

http://www.ex2.org/img/...

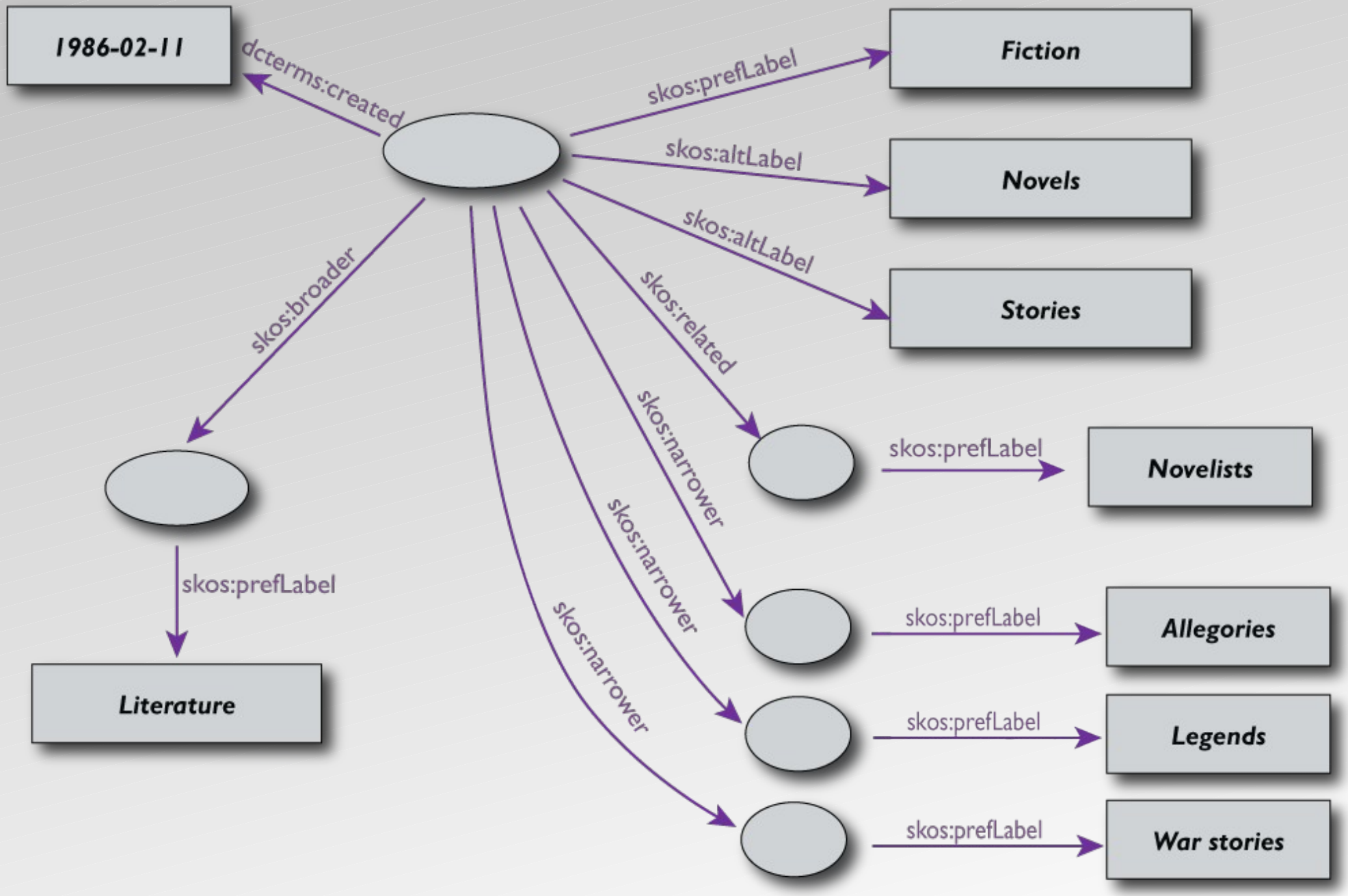
Simple Knowledge Organization System

- Goal: representing and sharing classifications, glossaries, thesauri, etc, as developed in the “Print World”. For example:
  - Dewey Decimal Classification, Art and Architecture Thesaurus, ACM classification of keywords and terms…
  - DMOZ categories (a.k.a. Open Directory Project)
- The system must be simple to allow for a quick port of traditional data (done by non-experts in, say, Semantic Web)
- This is where *SKOS* comes in: define classes, properties, where those structures can be added

Example: Library of Congress subject headings



The structure can be translated into RDF



SKOS Reference overview

- Classes and Properties:
  - Basic description (**`Concept`**, **`ConceptScheme`**, …)
  - Labeling (**`prefLabel`**, **`altLabel`**, …)
  - Documentation (**`definition`**, **`historyNote`**, …)
  - Semantic relations (**`broader`**, **`narrower`**, **`related`**, …)
  - Collections (**`Collection`**, **`OrderedCollection`**, …)
  - Concept mappings (**`broadMatch`**, **`narrowMatch`**, …)

SKOS and digital libraries

- SKOS plays an important role in “bridging” to digital libraries
- A huge community with its own traditions, style…
- … but huge amount of data to be “linked” to the Semantic Web!
- Major library metadata standards are being re-defined in terms of RDF (and SKOS),
  - eg, “Resource Description and Access” (RDA)
    - a major cataloging rule set for librarians
    - potentially, all major library catalogs around the globe could be translated into RDF and, eg, linked as an Open Linked Data…

Ontologies: OWL

- This is also a stable specification since 2004
- Separate layers have been defined, balancing expressibility vs. implementability (OWL-Lite, OWL-DL, OWL-Full)
- Looking at the tool list on W3C's wiki again:
  - a number programming environments include OWL reasoners
  - stand-alone reasoners (downloadable or on the Web)
  - ontology editors come to the fore
- OWL-DL and OWL-Lite relies on Description Logic, ie, uses a large body of accumulated knowledge

Ontologies

- Large ontologies are being developed (converted from other formats or defined in OWL). For example:
  - eClassOwl: eBusiness ontology for products and services, 75,000 classes and 5,500 properties
  - National Cancer Institute's ontology: about 58,000 classes
  - Open Biomedical Ontologies Foundry: a collection of ontologies, including the Gene Ontology, to describe gene and gene product attributes; or UniProt for protein sequence and annotation terminology and data
  - BioPAX: for biological pathway data
  - ISO 15926: "Integration of life-cycle data for process plants including oil and gas production facilities"

OWL in applications

- An increasing number of applications rely on OWL (Pfizer, Nasa, Eli Lilly, Elsevier, FAO, …)
  - it is worth looking at the yearly OWLED workshop proceedings
- Not all use complex reasoning; in many cases a small fraction of OWL is used

New OWL Working Group

- A new Working Group started on the revision of OWL
- The goal of the group:
  1. add a few extensions to current OWL that are useful, and is known to be implementable
     - many things happened in research since 2004
     - features should (if possible) be valid both in the DL and OWL Full world
  2. define “profiles” of OWL that are:
     - smaller, easier to implement and deploy
     - cover important application areas and are easily understandable to non-expert users

“OWL 2”: new proposed features

- “Qualified cardinality restrictions” (eg, “class instance must have two black cats”)
- Disjoint, reflexive, irreflexive properties; disjoint union of classes
- Property chains (eg, the uncle example: “if z is brother of y and y is father of x, then z is uncle of x”)
- Own datatype restriction constructs instead of complex XML Schema datatypes
- “Easy Keys”, i.e., identifying instances through common values of some properties

“OWL 2”: new proposed features (cont)

- Metamodeling (a.k.a. “punning”): the same symbol may be used both as, e.g., a Class and an Instance
  - removes a significant restriction from current OWL DL
  - not and issue for OWL Full, but not all “natural” inferences can be drawn in OWL 2 DL. Eg,

```
q rdf:type A.
A owl:sameAs B.
```

  - does *not* yield

```
q rdf:type B.
```

“OWL 2”: smaller profiles

- For a number of applications RDFS is not enough, but even OWL Lite is too much
  - also: OWL Lite *is a DL subset*, ie, includes some restrictions v.a.v. OWL Full, which may be a problem in practice
- There is a need for “light” versions of OWL: just a few extra possibilities added to RDFS

New profiles in OWL 2 (DL)

- OWL 2 retains the duality of OWL 2 DL and OWL 2 Full
- Classification and instance queries in polynomial time: *OWL-EL*
- Smaller vocabularies over a possibly very large set of RDF data: *OWL-QL* and *OWL-RL*
  - OWL-QL implementable via query rewriting on RDB systems
  - OWL-RL can be implemented with rule engines, too

Vocabularies

- There are also a number “core vocabularies” (not necessarily OWL based)
  - Dublin Core: about information resources, digital libraries, with extensions for rights, permissions, digital right management
  - FOAF: about people and their organizations
  - DOAP: on the descriptions of software projects
  - SIOC: Semantically-Interlinked Online Communities
  - vCard in RDF
  - …
- One should never forget: ontologies/vocabularies must be shared and reused!

Ontologies, vocabularies

- Ontology and vocabulary development is still a complex task
  - but the *really* difficult task is to *understand* a specific area of knowledge
  - OWL is, in some ways, only syntax…
- A W3C Working Group has developed some documents:
  - “Best Practice Recipes for Publishing RDF Vocabularies”
  - “Defining N-ary relations”
  - “Representing Classes As Property Values”
  - “Representing "value partitions" and "value sets"”
  - “XML Schema Datatypes in RDF and OWL”
- More work is needed…

Rules

- There is a long history of rule languages and rule-based systems
  - eg: logic programming (Prolog), production rules
- Lots of small and large rule systems (from mail filters to expert systems)
- Hundreds of niche markets

Why rules on the Semantic Web?

- There are conditions that ontologies (ie, OWL) cannot express (or only with difficulties)
  - a well known example is Horn rules: $(P1 \wedge P2 \wedge \ldots) \rightarrow C$
- There are conditions that are complicated in rules and ontologies are better (eg, complex classification of terms)
- Simple rule engines might be easier to implement (eg, on top of database engines)
- A different way of thinking — people may feel more familiar in one or the other

Things you may want to express

- An example:
  - “if two Persons have the same name and the same email, or the same name and the same home page, then they are identical”
- Something like (with an ad-hoc syntax):

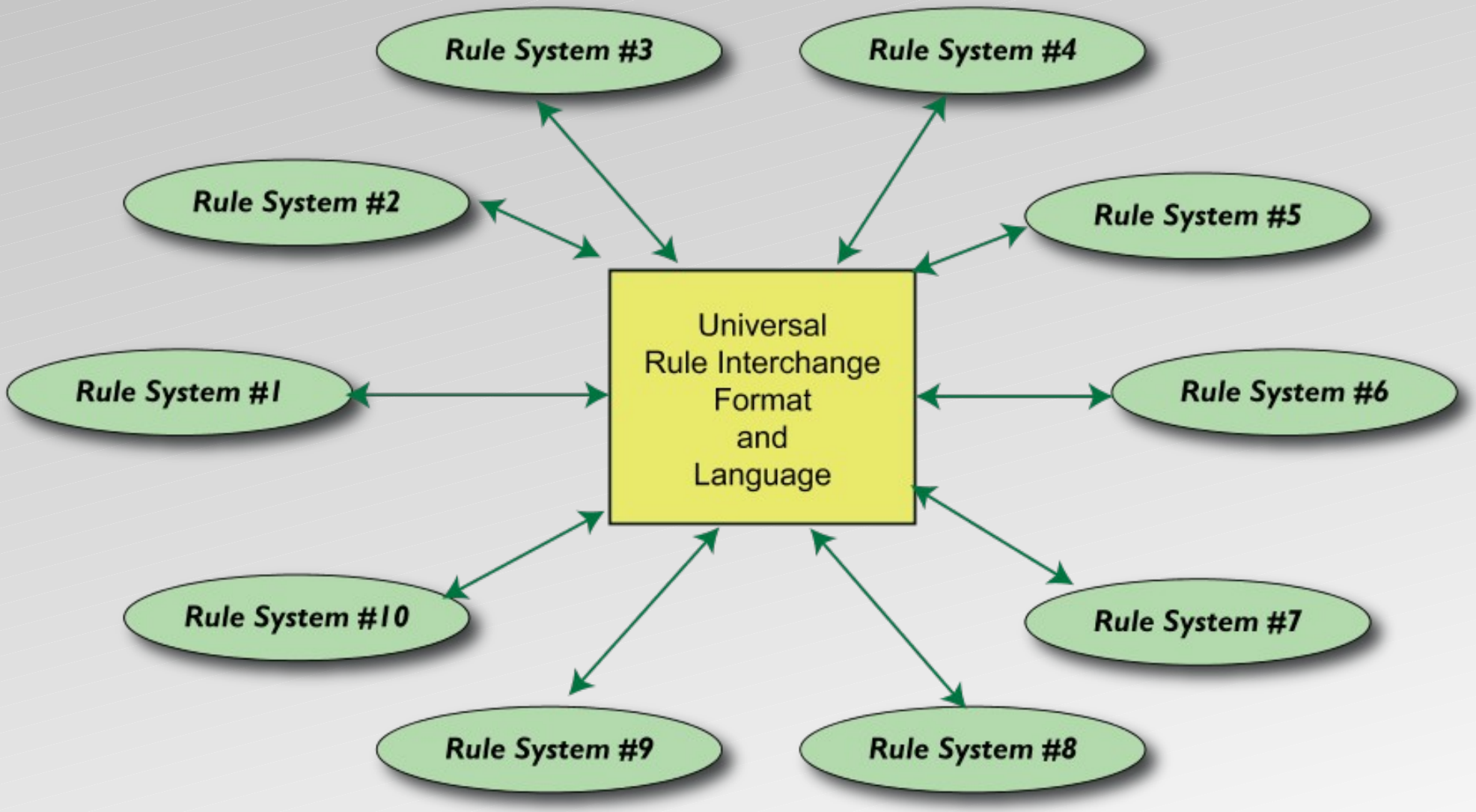
```
If { ?x rdf:type foaf:Person.
     ?y rdf:type foaf:Person.
     ?x foaf:name ?n.
     ?x foaf:homepage ?h.
     ?y foaf:name ?n.
     ?y foaf:homepage ?h. }
then { ?x owl:sameAs ?y }

If { ?x rdf:type foaf:Person.
     ?y rdf:type foaf:Person.
     ?x foaf:name ?n.
     ?x foaf:mailbox ?h.
     ?y foaf:name ?n.
     ?y foaf:mailbox ?m. }
then { ?x owl:sameAs ?y }
```

A new requirement: *exchange* of rules

- Applications may want to *exchange* their rules:
  - negotiate eBusiness contracts across platforms: supply vendor-neutral representation of your business rules so that others may find you
  - describe privacy requirements and policies, and let clients "merge" those (e.g., when paying with a credit card)
- Hence the name of the working group: *Rule Interchange Format*
  - a language that
    - expresses the rules a bit like a rule language with, eg, RDF
    - can be used to exchange rules among engines

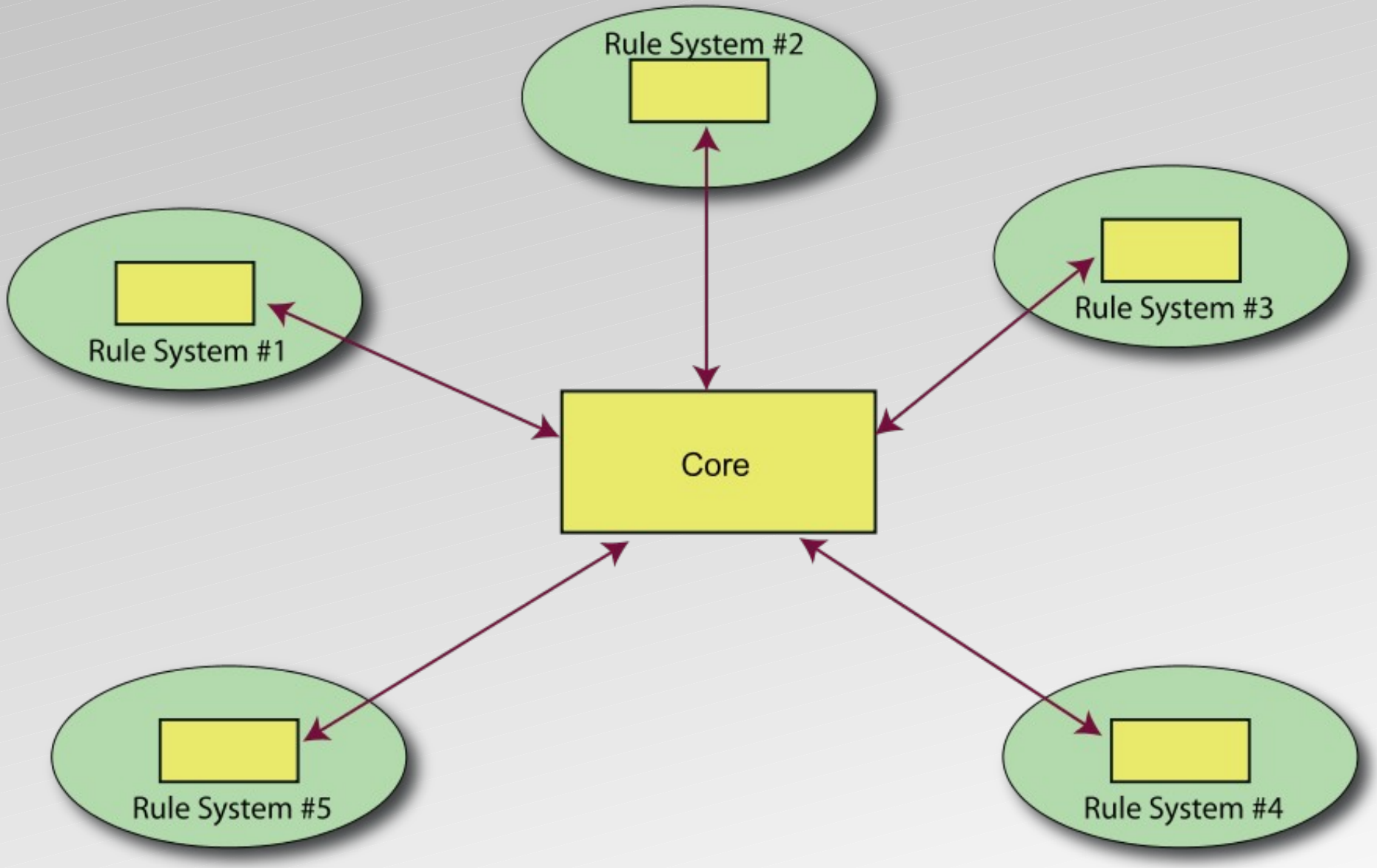
In an ideal World



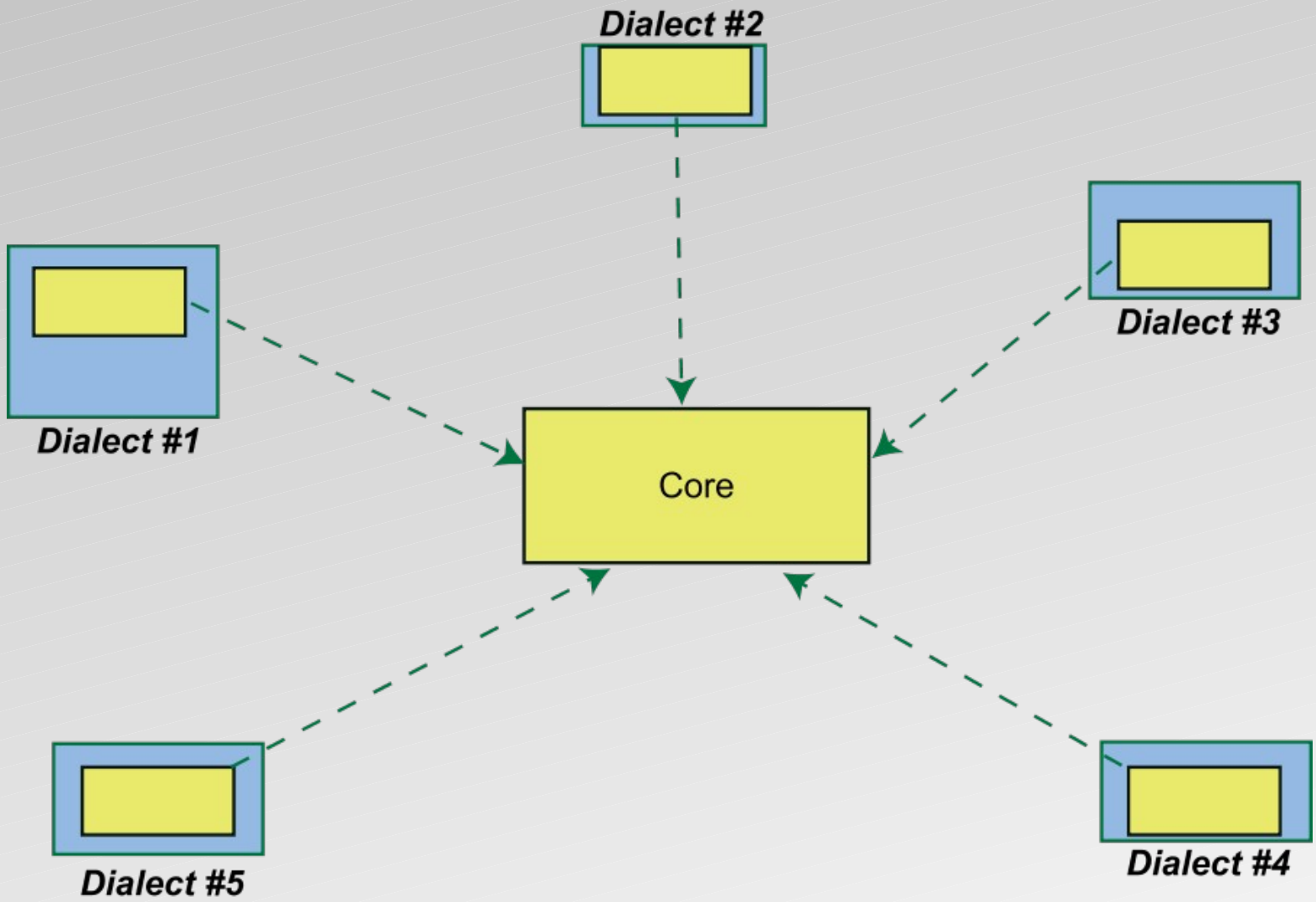
In the real World…

- Rule based systems can be very different
  - different rule semantics (based on various type of model theories, on proof systems, etc)
  - production rule systems, with procedural references, state transitions, etc
- Such universal exchange format is not feasible
- The idea is to define “cores” for a family of languages with “variants”

RIF “core”: only partial interchange

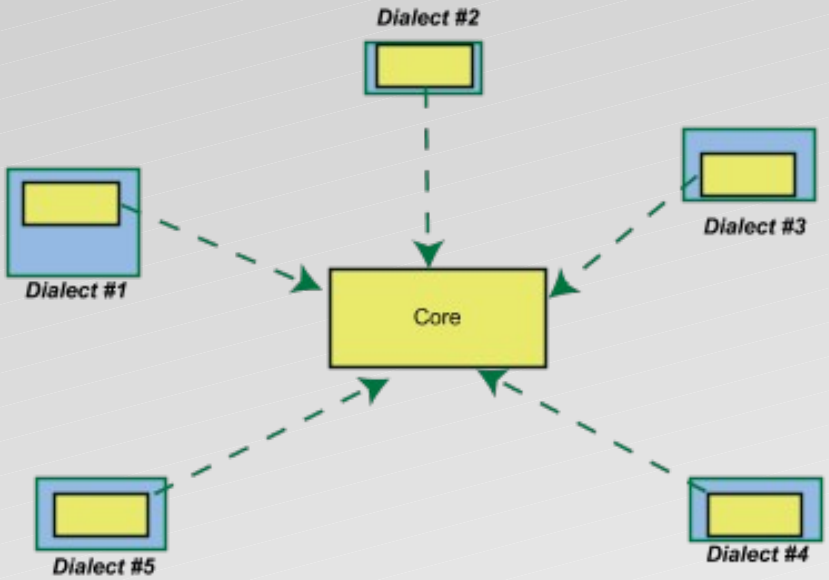


RIF “dialects”

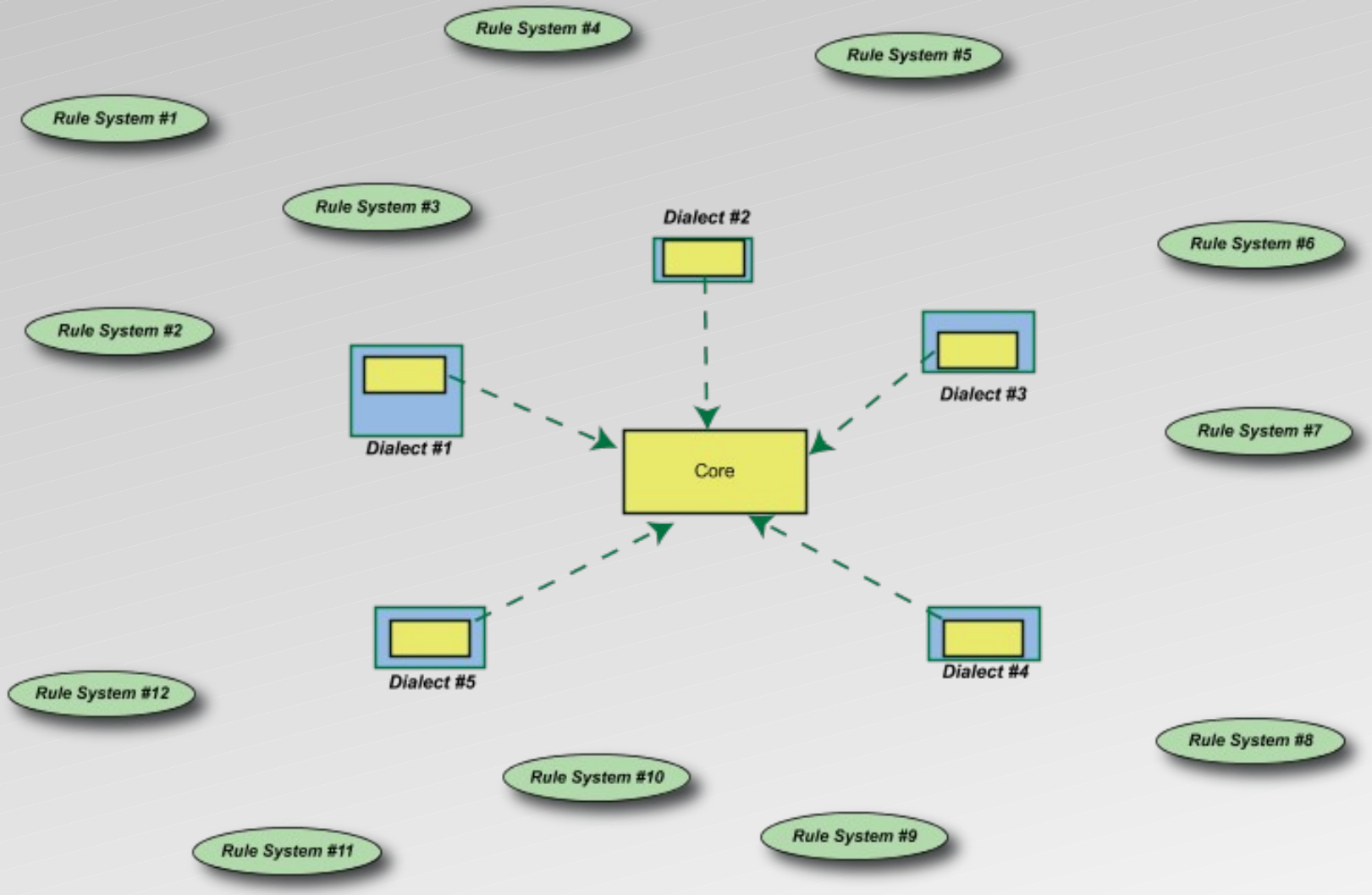


- Possible dialects: F-logic, production rules, fuzzy logic systems, …

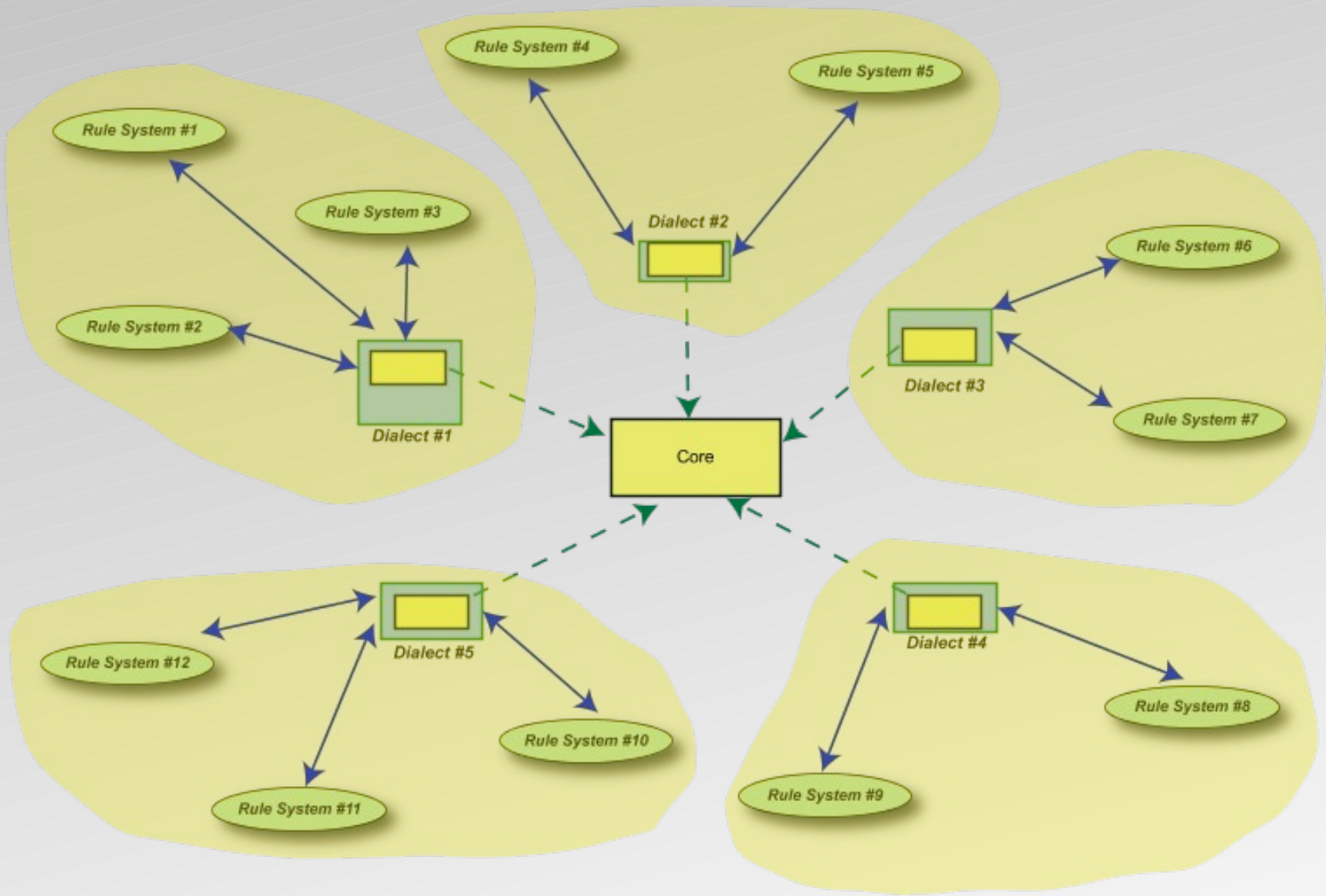
Role of dialects



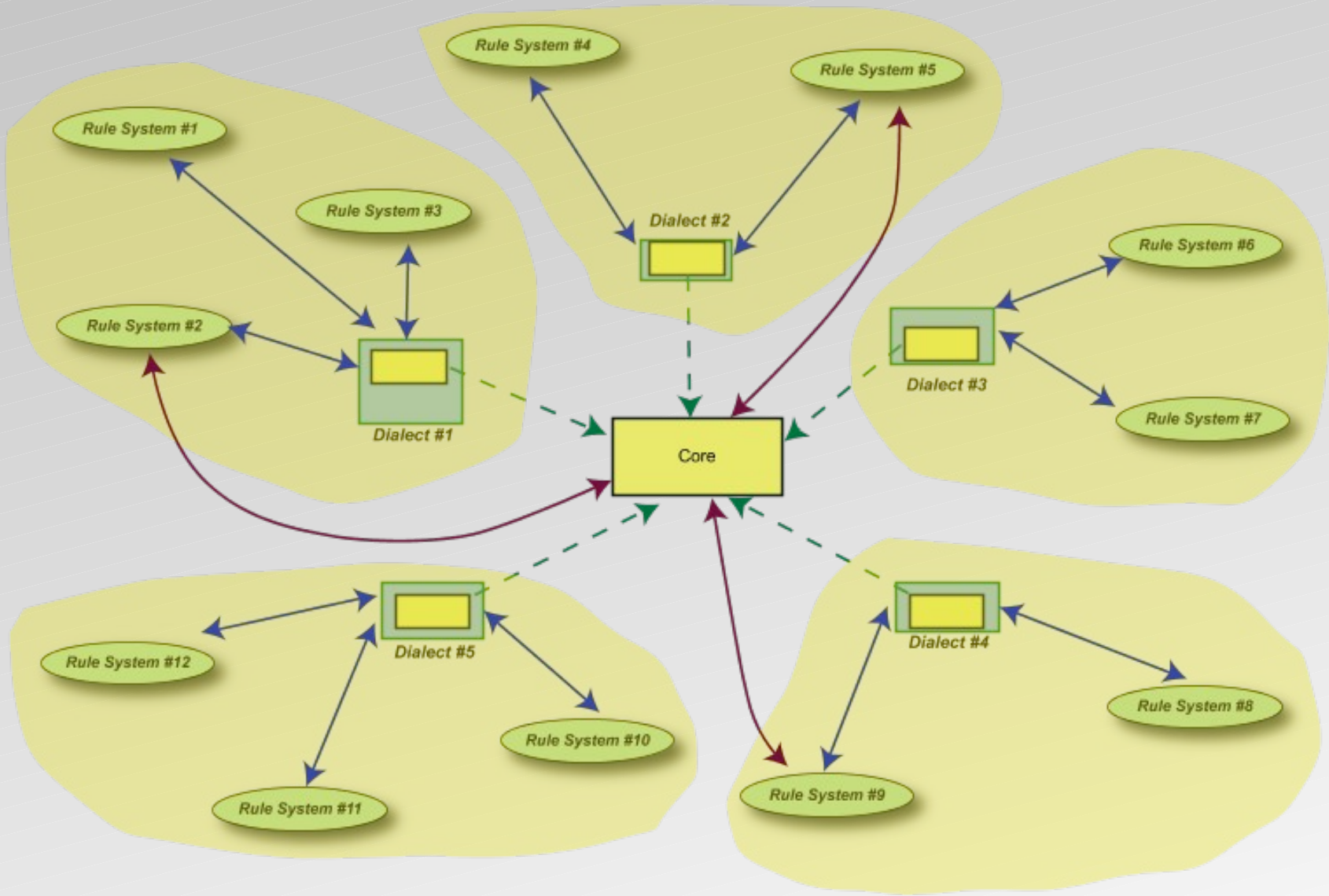
Role of dialect



Role of dialects



Role of dialects



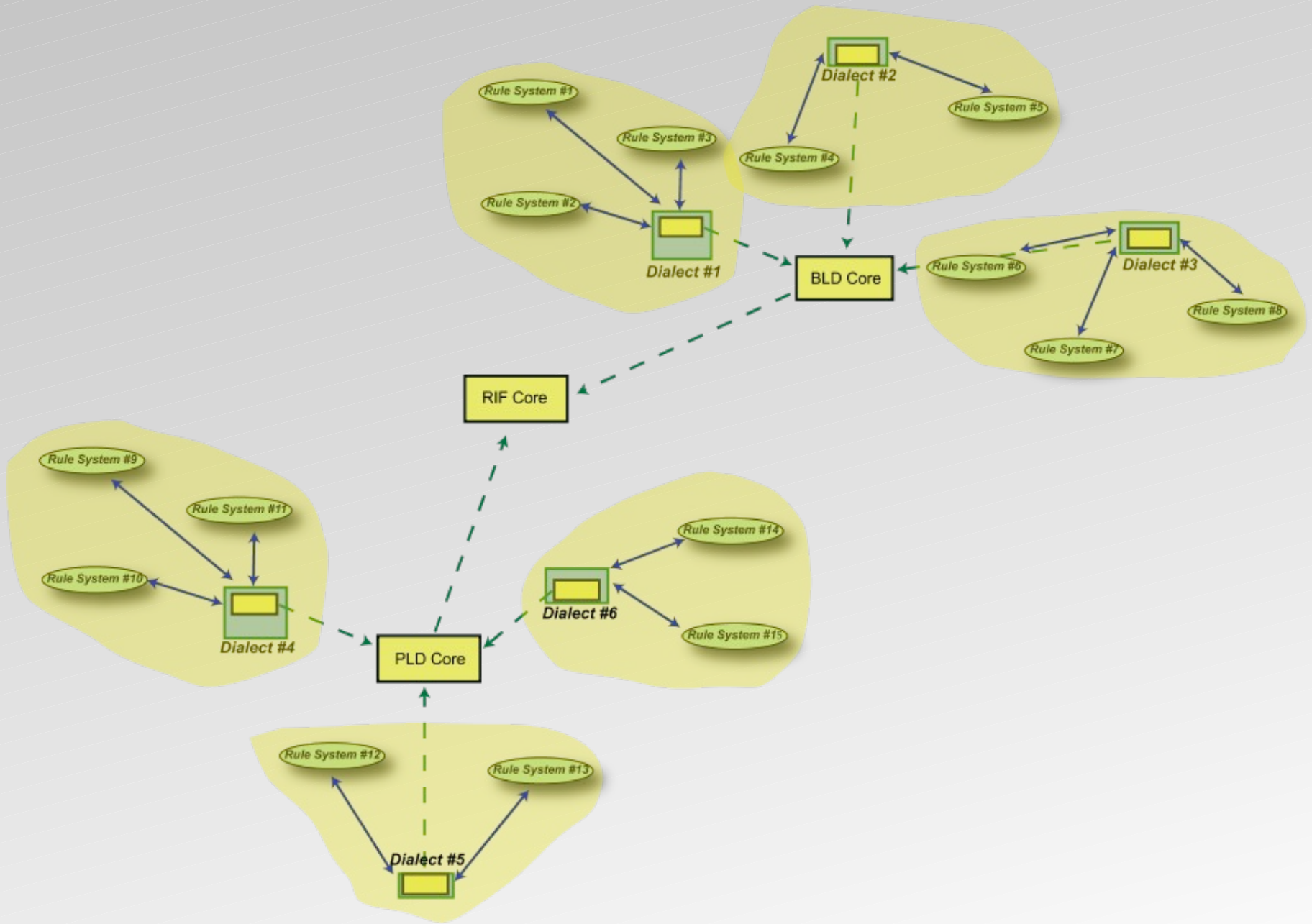
However…

- Even this model does not completely work
- The gap between production rules and logic systems seems to be deep
- A hierarchy of cores is necessary:
  - a *Basic Logic Dialect* and *Production Rule Dialect* as “cores” for families of languages
  - a common *RIF Core* binding these two

Schematically…

- The “Core”: shared subset of major languages
  - technically: positive Horn without function terms, with some simple datatypes
- The “BLD (Basic Logic Dialect)” is of the form:
  - “**`if condition true then this is true`**”
  - conditions may include functions, hierarchies
- The “PLD (Production Logic Dialect)” is of the form:
  - “**`if condition is true then do something`**”

Hierarchy of cores



Current status

- Drafts for BLD and PLD are available
  - they have XML syntax and an extension mechanism for dialects
- BLD can be used
  - with or without RDF data and/or OWL
  - as a rule *language* or a rule *interchange format*
- The Core will come as an abstraction from BLD and PLD
- Target date for finalization: spring 2009

Everything has not been solved…

- There are a number of issues, problems
  - missing functionalities: encryption/signatures, fuzzy reasoning, …
  - misconceptions, messaging problems
  - need for more applications, deployment, acceptance
  - etc

Other items…

- Security, trust, provenance
  - combining cryptographic techniques with the RDF model, sign a portion of the graph, etc
  - trust models
- Quality constraints on graphs
  - “may I be sure that certain patterns are present in a graph?”
- Ontology merging, alignment, term equivalences, versioning, development, …
- etc

Other items: uncertainty

- Fuzzy logic
  - look at alternatives of DL based on fuzzy logic
  - alternatively, extend RDF(S) with fuzzy notions
- Probabilistic statements
  - have an OWL class membership with a specific probability
  - combine reasoners with Bayesian networks
- A W3C Incubator Group issued a report on the current status, possibilities, directions, etc
  - report published in April 2008
- Possible RIF dialect for fuzzy logic, for example?

Other items: naming

- The SW infrastructure relies on unique naming of “things” via URI-s
- Lots of discussions are happening that also touch upon general Web architecture:
  - HTTP URI-s or other URN-s?
    - using non-HTTP unnecessarily complicates the general infrastructure
  - URI-s for “informational resources” and “non informational resources”
  - how to ensure that URI-s used on the SW are dereferencable
  - what inferences can be drawn upon an HTTP session?
  - etc

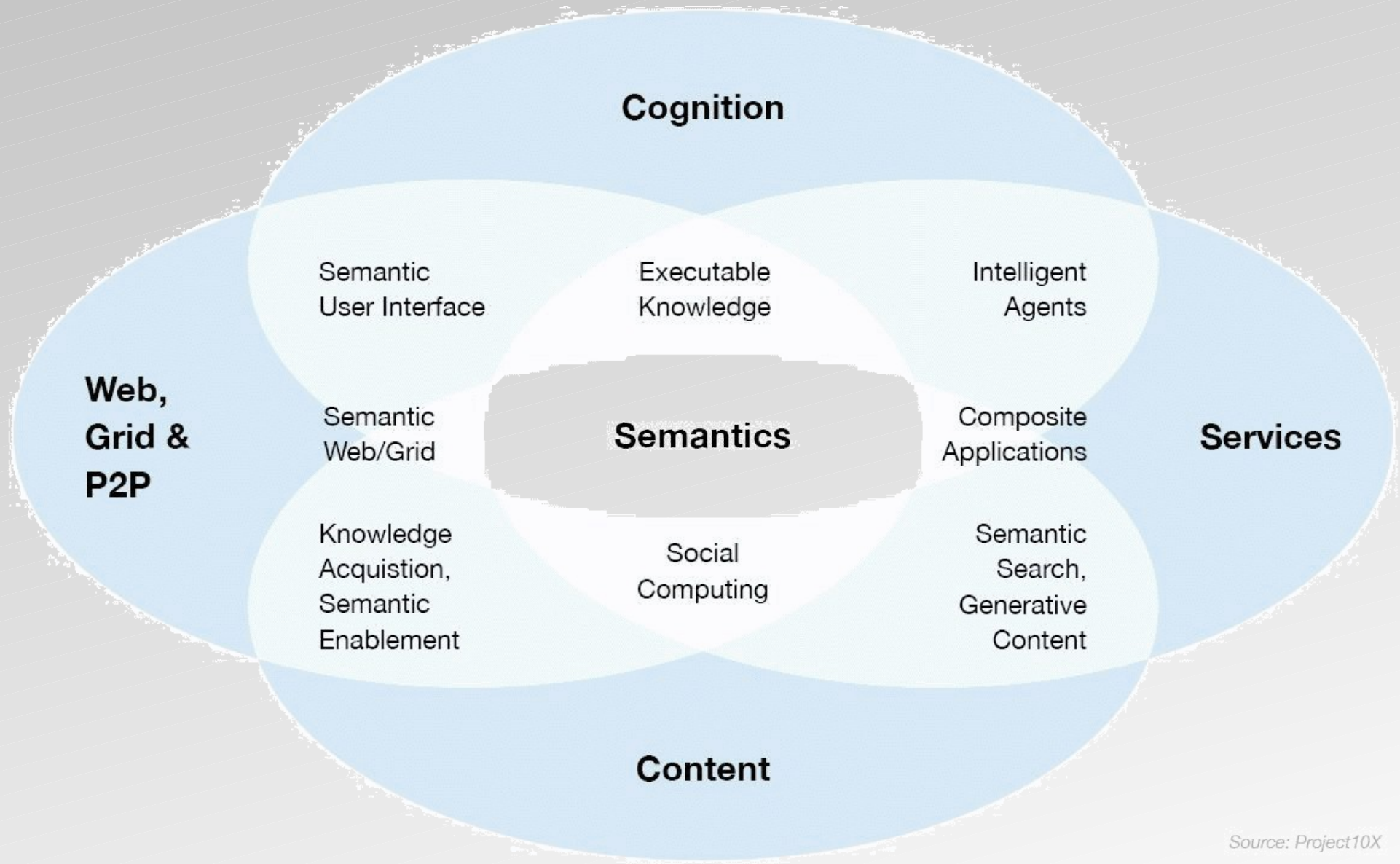
Other items: naming (cont)

- A different aspect of naming: *what* is the URI for a specific entity (regardless of the technical details)
  - what is the unique URI for, eg, Bach's Well-Tempered Clavier?
    - obviously important for, eg, music ontologies and data
    - who has the authority or the means to define and maintain such URI-s?
    - should we define characterizing properties for these and use **`owl:sameAs`** instead of a URI?
    - the traditional library community may be of a big help in this area
  - what is the URI of time-dependent entity (e.g., a specific point *within* a video)?

Revision of the RDF model?

- Some restrictions in RDF may be unnecessary (bNodes as predicates, literals as subject, …)
- Issue of “named graph”: possibility to give a URI to a set of triplets and make statements on those
- Syntax issues in RDF/XML (eg, QNames in properties)
- Add a time tag to statements?
- Internationalization issues with literals (how do I set “bidi” writing?)
- …

Another view of the R&D trends



Courtesy of Miles Davis, © Project10X

A major problem: messaging

- Some of the messaging on Semantic Web has gone terribly wrong over the years
- This has created lots of (unnecessary) controversies

Some of the frequent misconceptions…

- The Semantic Web is simply Artificial Intelligence on the Web
- It is just an ugly application of XML
- One has to add metadata to all Web pages, convert all databases and XML data to use the Semantic Web
  - which is obviously unrealistic…
- One has to learn formal logic, knowledge representation techniques, etc, to use it (ie, it has nothing to do with the Web)

… and some more …

- It relies on a centrally controlled super-ontology for "meaning"
  - and this is impossible, because people will never agree on all terms
  - as opposed to a democratic, bottom-up control of terms
- It is, essentially, an academic project, of no interest to industry and the real World

RDF ≠ RDF/XML!

- RDF is a model, and RDF/XML is only one possible serialization thereof
  - lots of people prefer, for example, Turtle
  - a good percentage of the tools have Turtle parsers, too!
- The model is, after all, simple: interchange format for Web resources. That is it!

RDF ≠ RDF/XML! (cont.)

- RDF/XML is indeed a very complex serialization format
- Certainly not the nicest possible XML application
  - good to know that it was created when XML was not yet final…
- Again: it is only syntactic sugar!
- One has to emphasize: RDF is *not* an XML application!

RDF is not that complex…

- Of course, the formal semantics of RDF is complex
- But the average user should not care, it is all "under the hood"
  - how many users of SQL have ever read its formal semantics?
  - it is not much simpler than RDF…
- *People should "think" in terms of graphs*, the rest is syntactic sugar!

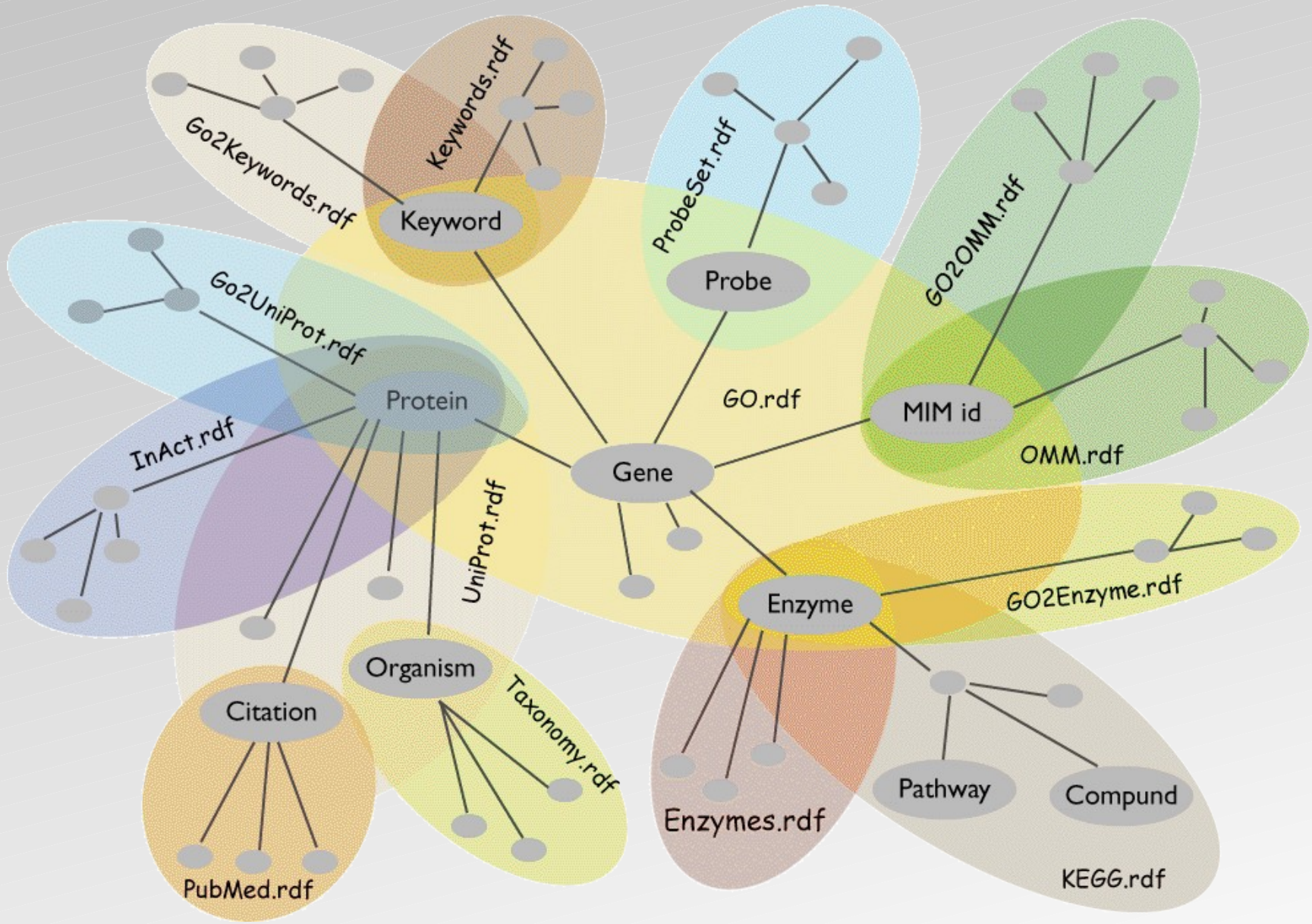
Semantic Web ≠ Ontologies on the Web!

- Large, formal ontologies (like OWL) are important
  - general applications patterns indeed rely on them
- But: use complex ontologies *only when necessary*
  - you can be a perfectly decent citizen of the Semantic Web if you do not use OWL, or not even RDFS…
- Some of the ontologies may not be that complex
  - this is the essence of the development of “profiles” in the current OWL Working Group
  - just a small set of suitable terms to discover new relationships

SW Ontologies ≠ a central, big ontology!

- The “ethos” of the Semantic Web is on sharing, ie, sharing ontologies (small or large)
- A huge, central ontology would be unmanageable
- The practice:
  - SW applications using ontologies always mix large number of ontologies and vocabularies (FOAF, DC, and others)
  - the real advantage comes from this mix: that is also how new relationships may be discovered

A good example…



Courtesy of Joanne Luciano, Mitre, and the W3C HCLS IG

Web 2.0 and SW are no enemies…

- Web 2.0 recognized the importance of data to be processed, mashed-up, mixed
  - this is at the heart of the Semantic Web
  - SW provides a set of consistent tools and definitions to achieve that

Web 2.0 and SW are no enemies… (cont.)

- Sometimes the simplicity of Web 2.0 (eg, in tagging, microformats) pays off…
- …sometimes more rigour is necessary in which case Semantic Web technologies come to the fore
  - GRDDL is a good example for a “bridge”
  - SPARQL can be used for more complex mash-ups
- Let us forget about a turf/ego war; it is unnecessary and counter-productive

It is not about SW vs. RDB vs. Web Services

- Each technology has its place and role
- Relational databases and SQL are here to stay forever…
- The goal is *integration* and not replacement

Semantic Web ≠ academic research only!

- SW has indeed a strong foundation in research results
- For a list of applications, see separate slide set

Thank you for your attention!

- These slides are publicly available on:

http://www.w3.org/People/Ivan/CorePresentations/State_of_SW/