Is the Semantic Web hype?

Dr Mark H. Butler
Digital Media Systems Department
HP Labs Bristol
mark-h.butler@hp.com
http://www-uk.hpl.hp.com/people/marbut/

22 April 2004

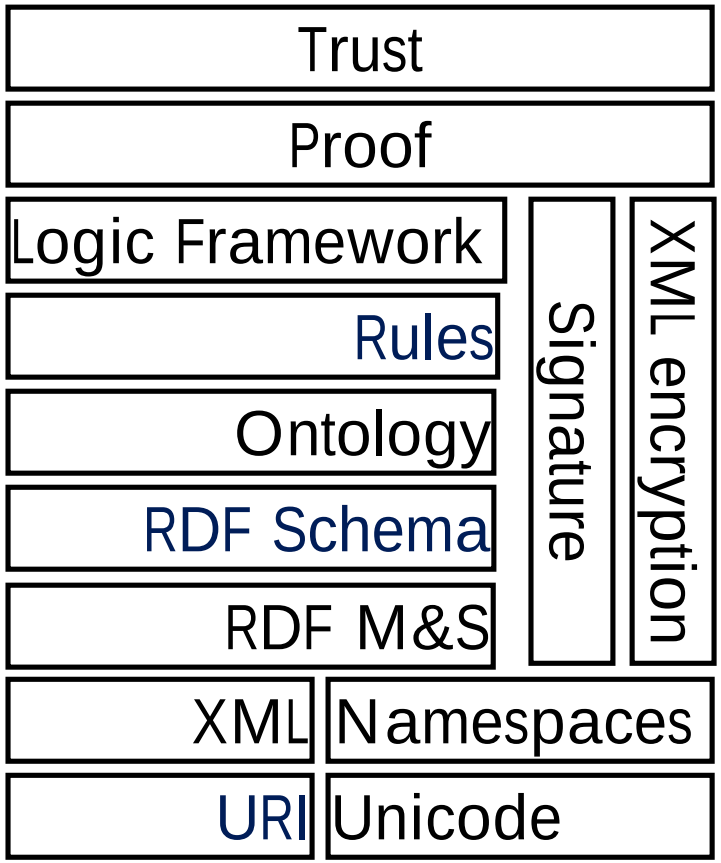
What is the Semantic Web?

"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. The mix of content on the web has been shifting from exclusively human-oriented content to more and more data content.

The Semantic Web brings to the web the idea of having data defined and linked in a way that it can be used for more effective discovery, automation, integration, and reuse across various applications. For the web to reach its full potential, it must evolve into a Semantic Web, providing a universally accessible platform that allows data to be shared and processed by automated tools as well as by people."

W3C Semantic Web Activity Statement
http://www.w3.org/2001/sw/Activity

The Semantic Web stack



Layer 1 – URIs and Unicode

- URIs stand for Uniform Resource Identifiers
- There are different subclasses of URIs e.g.
  - Universal Resource Names (URNs) allow things to be uniquely identified
  - Universal Resource Locators (URLs) allow resources to be retrieved e.g.

    http://www.hp.com/
- Unicode is a replacement for ASCII that can cope with multiple languages

Layer 2 – XML

- XML is a standard format for serializing data using tags
- It is derived from SGML and similar to HTML e.g.

```
<WV-CSP-Message xmlns="http://www.wireless-village.org/CSP1.0">
   <TransactionContent xmlns="http://www.wireless-village.org/TRC1.0">
    <CapabilityList>
     <ClientType>MOBILE_PHONE</ClientType>
     <InitialDeliveryMethod>P</InitialDeliveryMethod>
     <AcceptedContentLength>32767</AcceptedContentLength>
    </CapabilityList>
  </TransactionContent>
 </WV-CSP-Message>
```

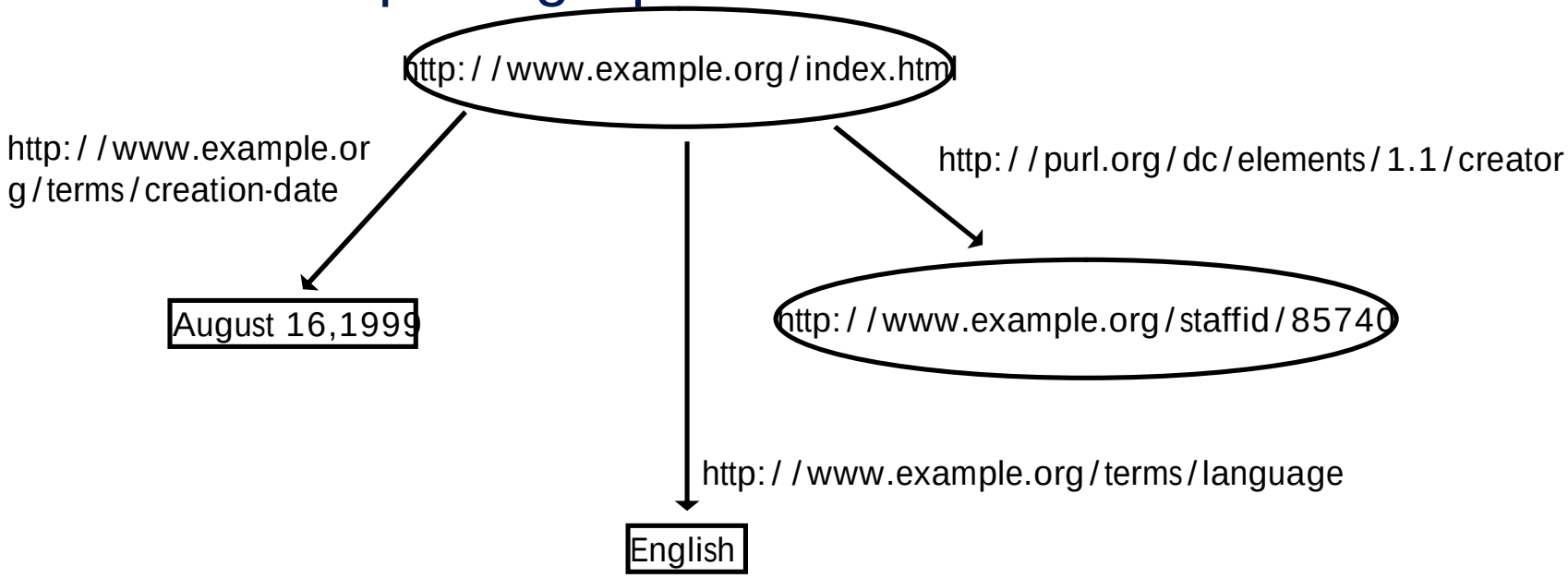
- There are many tools available for XML e.g.
  - XSLT for transformation
  - DOM and SAX parsers
  - schema languages like XML Schema for validation
  - XQuery for querying data

Layer 2 – Namespaces

- XML Namespaces are extensions to XML
- In a namespace, a URI is modelled as a qName e.g.
  `http://www.wapforum.org/profiles/UAPROF/ccppschema-20010330#BitsPerPixel`
- The qName consists of
  - a qualifier which indicates the vocabulary
  - a fragment which indicates the element in the vocabulary
- We group common elements into a vocabulary i.e. they share the same qualifier
- The Semantic Web assumes there will be many different and perhaps overlapping vocabularies. Namespaces provide a means of uniquely identifying every item in every vocabulary

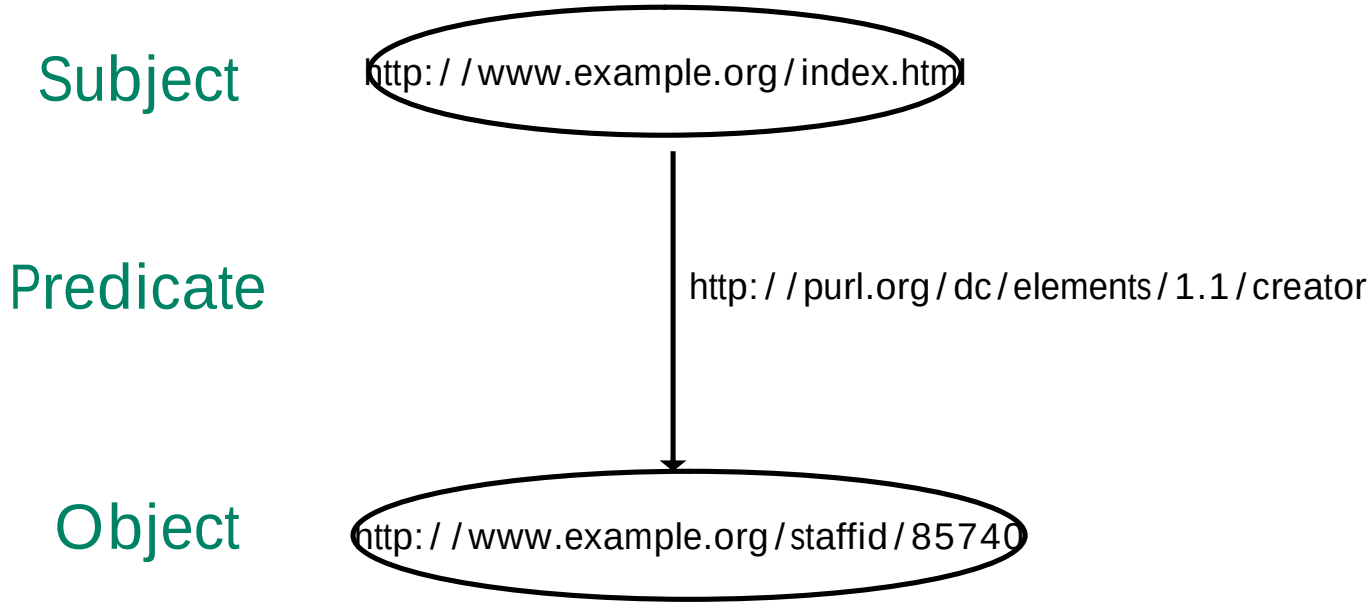
Layer 3 - RDF Model (1)

- RDF is a graph where the arcs are represented by qNames and nodes are represented by qNames, local names, blank nodes or literals
- Similar to semantic networks, frames or John Sowa's work on conceptual graphs



Layer 3 - RDF Model (2)

- The RDF graph consists of triples, where each triple represents a statement about a resource consisting of a subject, a predicate and an object



Layer 3 - Comparing RDF with databases (1)

- RDF, like object oriented databases and XML databases, can model semistructured data
- RDF is arguably more flexible but currently less efficient than other representations, typically requiring ten times as much memory
- This is because
  - in RDF everything is represented as a graph even if there is a more efficient representation
  - every node, and sometimes every arc, is indexed
  - all arcs and most nodes are URIs, namespace compression is not straightforward
- Persistant RDF frameworks generally serialize to an existing relational databases. Typically they are between 10 to 100 times slower than in memory representations

Layer 3 - Comparing RDF with databases (2)

- Merging heterogeneous databases has been a problem that the database community have been working on for a while
- SW advocates have stated it is easy to merge different databases in RDF because the nodes and arcs are represented as URIs, so you just denote which URIs are equivalent
- However in practice it is slightly more difficult:
  - if we have big datasets, determining equivalence by hand will take a lot of time. We need automated methods to identify equivalences, which is what the database research focused on
  - simply merging RDF graphs loses information, so we need mechanisms for resolution i.e. resolving conflicts or storing the context of the conflicting information

“The web is a powerful tool for sharing scientific data. Can semantic web technologies enhance that power, and will those technologies scale to support data-intensive applications? Here I explore the benefits of semantic web technologies RDF, RDF Schema over existing such as XML and XML Schema in the context of a scientific application. The ability of RDF to support graph-based query was identified as a potential benefit in this application, allowing chemists to search a repository of molecules for molecular sub-structures.

I report on tests carried on the performance of (…) query engine over RDF repositories containing molecular structures. Queries for simple molecular substructures (such as N-C-N) within modest repositories (100 molecules, 26118 statements) took a prohibitively large amount of time to complete (>24 hours). Unless efficiency of graph query engines is improved, RDF technologies remain inadequate for data-intensive applications.”

Alistair Miles

http://www.w3c.rl.ac.uk/SWAD/papers/RDFMolecules_final.doc

Layer 3 - The multiple serialisations of RDF/XML (1)


```
<?xml version="1.0"?>
  <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
             xmlns:ex="http://example.org/schemas/vehicles">
    <ex:PassengerVehicle rdf:ID="johnSmithsCar">
         <ex:registeredTo rdf:resource="http://www.example.org/staffid/85740"/>
         <ex:rearSeatLegRoom>127</ex:rearSeatLegRoom>
         <ex:primaryDriver rdf:resource="http://www.example.org/staffid/85740"/>
    </ex:PassengerVehicle>
</rdf:RDF>

<?xml version="1.0"?>
  <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
             xmlns:ex="http://example.org/schemas/vehicles">
    <rdf:Description rdf:ID="johnSmithsCar">
         <rdf:type rdf:resource="http://example.org/schemas/vehicles#PassengerVehicle"/>
         <ex:registeredTo rdf:resource="http://www.example.org/staffid/85740"/>
         <ex:rearSeatLegRoom>127</ex:rearSeatLegRoom>
         <ex:primaryDriver rdf:resource="http://www.example.org/staffid/85740"/>
    </rdf:Description>
</rdf:RDF>

<?xml version="1.0"?>
  <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
             xmlns:ex="http://example.org/schemas/vehicles">
    <ex:PassengerVehicle rdf:ID="johnSmithsCar" >
         <ex:registeredTo rdf:resource="http://www.example.org/staffid/85740"/>
         <ex:primaryDriver rdf:resource="http://www.example.org/staffid/85740"/>
    </ex:PassengerVehicle>
    <rdf:Description rdf:ID="johnSmithsCar" ex:rearSeatLegRoom="127"/>
</rdf:RDF>
```

Layer 3 - The multiple serialisations of RDF/XML (2)

- RDF/XML has a striped syntax which is hard to understand
- As there are multiple serialisations it cannot be read using standard XML tools e.g.
  - cannot use schema validation languages so greater tendency for RDF/XML documents to contain mistakes
  - it cannot be manipulated and repurposed using XSLT
- Have to use a specialist RDF parser which are 5 – 20 times slower than XML parsers
- Possible solutions:
  - use a subset of RDF/XML - some groups (EARL, MPV, RSS etc) are doing this
  - use an alternative serialisation for RDF called N3, but this is not an official standard
  - use XML as canonical format, then use XSLT to convert XML to RDF/XML
- Easy to solve technological problem, real problem is political

"It does not seem to us that the XML serialization of RDF shows RDF to advantage. At the level of the underlying graph model, RDF information has a simple and regular structure, which appears in the XML serialization to be anything but simple and so irregular as to bring the words "capricious" and "arbitrary" to the lips of unprejudiced observers. Tastes in markup style differ, but we believe that the root of the problem is the high degree of variability with which the same underlying graph structures may be serialized, according to the rules given in this document.

Owing in part to the variability itself, and in part to the specific forms taken by that variability, it is not feasible to write an XML Schema schema, or a Relax NG schema, or an XML 1.0 DTD, which defines the set of correct serializations of correct RDF graphs. It is not convenient to run XSLT processes over arbitrary RDF serializations, nor to query or process arbitrary RDF data using XQuery. There is, as a result, something of a cleft between the RDF community and the set of RDF tools on the one hand, and the community of users and tools employing what some have called colloquial XML... The result is that not just arbitrary RDF data, but data encoded using vocabularies defined in RDF terms (for which current W3C work provides a number of examples), will be hard to process using off-the-shelf tools."

XML Schema Working Group

http://lists.w3.org/Archives/Public/www-rdf-comments/2003JanMar/0489.html

The RPV language proposal is motivated by a belief that RDF's problems are rooted at least in part in its syntax. Specifically:

1. The syntax of RDF/XML is sufficiently scrambled and arcane that it is neither human-writeable nor human-readable.
2. The RDF/XML syntax makes heavy use of qnames that is neither intuitive to humans nor conforms particularly well to Web Architecture, which requires that everything significant be identified by URI.
3. People who care about metadata have no trouble thinking in terms of resource/property/value triples.
4. Alternatives like N3 that make the RDF triples evident in syntax suffer in comparison to the RDF/XML syntax because they lack XML's widely-deployed base of software, i18n facilities, and APIs.
5. The notion that you RDF can be mixed into XML transparently enough to be unobtrusive has failed resoundingly in the marketplace.

Tim Bray,
Co-Inventor of XML
http://www.textuality.com/xml/RPV.html

“The following was heard at a W3C/WAP Forum Workshop:

We (a working group of 7 technicians from the WAP FORUM Telematics Expert Group) tried it (RDF). We tried like hell for over a week's time and we never got it. Sure we could put some things together with nodes and arcs, but after that we had no idea where to go. We downloaded every thing we could find, only to become more confused. XML is a cinch - but with RDF you have to make yourself a choice; Either RDF is stupid - or you are!

I thought this was a pretty brave thing to say, since nobody else in the room had dared to say that they had had trouble understanding RDF. But then assenters starting making themselves known through out the room. Despite who or what is stupid, I guess I am not as brave as the kid who called the king naked, in saying that the syntax and model specifications are not the documents they should be if we are going to win converts to the RDF cause. Perhaps they should be tightened up to the terseness of XML 1.0. Or someone can find a good pedagogue to take care of the verbosity stuff. That this group of engineers made a sincere effort to implement RDF and failed, is saddening.”

Greg Fitzpatrick, MediaNet

http://lists.xml.org/archives/xml-dev/200002/msg00432.html

The overlap between RDF and XML

- XML can be used to represent documents or to model data, so both RDF and XML can be applied to problems that involve modelling data
  - http://www.rpbourret.com/xml/XMLAndDatabases.htm
- However as already noted, standard tools have difficulty operating on RDF/XML
- This means there is a danger there will be a large amount of "reinventing the wheel" e.g. XForms vs RDFForms, XSLT vs RDF Rules, XQuery vs RDQL
- It also means developers using RDF cannot take advantage of the maturing toolset available for XML and this may impede the adoption of RDF
- RDF and XML actually do very similar things, so the W3C needs to coordinate work so these two standards work well together

On RDF Query:

"The W3C has just managed to get XQuery energized, yet we are looking to redo that work in yet another recommendation or method? Why? Rather than specify that a re-implementation of the semantics of XQuery be done, why not study the requirements of Xquery that capture the additional semantics and uses needed for OWL & OWL-RULES and make a cogent argument to the XQuery working group to formally extend their recommendation to encompass additional capabilities? ... We have hybrid reasoning working here with a Logic Program that calls out to an XQuery to hit a compiled OWL knowledge base, and it works fine. ... The W3C membership is already asking integrators and developers to learn XQuery. Saying to them that they need to learn and implement yet another query-oriented or operation-oriented methodology in order to get to the semantic web seems to be yet another barrier in an already bumpy road. We should be striving for less recommendations, but ones that hang together."

Jack Berkowitz, Network Inference

http://lists.w3.org/Archives/Public/www-rdf-rules/2003Nov/0080.html

Layer 3 – RDF Model Theory

- The RDF Model Theory is a logical theory that defines how to logically interpret an RDF model
- It provides formal semantics i.e. how to logically interpret the model rather than semantics i.e. the real world meaning of the model
- It is optimized for efficiency: conventionally when adding new statements to knowledge bases we have to consider if there is a contradiction
- The RDF MT avoids this problem by allowing potentially contradictory statements to be asserted concurrently, with the exception of data type clashes
- It does not provide any additional semantics for containers, collections and constraints
- This creates limitations. For example <alt> (short for alternative) is not considered to represent a disjunction

"Taken as a weak KR language (which is its purpose) RDF appears to be higher order. Since it is possible to state, in RDF:

likes = hates

Pat Hayes has developed a semantics for RDF that skirts this problem but hey: we have a weak, untyped language that needs some sophisticated logic to get a reasonable semantics. That sounds promising!"

Francis McCabe Fujitsu

http://lists.w3.org/Archives/Public/www-ws-arch/2002Aug/0162.html

Layer 4 – RDF Schema

- RDF Schema is a language for describing RDF vocabularies
- It can describe class hierarchies

```
rdfs:Resource
  example:LivingThing
    example:Dog
    example:Human
  example:EmailAddress
```

- And property hierarchies

```
rdf:Property
  example:relative
    example:parent
      example:father
  example:hasEmailAddress
```

- It allows the domain and range of properties to be constrained

```
example:hasEmailAddress
Domain=example:Human, Range=example:EmailAddress
```

- Recent revisions to RDFS support data typing

Layer 4 – Comparing RDF and RDF Schema with object technologies

- The symbolic AI community used approaches such as frames and semantic networks to model information
- RDF is closer to frames than objects as it is possible to add new slots to a class instance in RDF
- Object approaches can also describe class hierarchies but they are more limited in their ability to deal with property hierarchies. They can refine and generalize properties, but only via subclassing and polymorphism
- Property hierarchies are more complex conceptually than class hierarchies:
  - for example is biologicalFather a subproperty of father?

Layer 5 – Ontology languages

- OWL and DAML+OIL are more complex languages for describing RDF vocabularies
- They can describe cardinality constraints on properties
  - e.g. that a Person has exactly one biological father
- transitivity
  - e.g. if A hasAncestor B, and B hasAncestor C, then A hasAncestor C
- that a given property is a unique identifier
- that two different classes represent the same concept
- that two different instances represent the same individual
- new classes in terms of combinations of other classes
  - e.g. unions and intersections
- that two classes are disjoint
  - i.e. that no resource is an instance of both classes

Hierarchical ontology languages have some well known limitations

- Taken from Jones & Paton “Some problems in the formal representation of hierarchical knowledge” http://citeseer.nj.nec.com/jones98some.html
- Atypical instances:
  - an instance is not a typical example of a class to which it belongs, leading to difficulties with identification and inheritance
- Context sensitive membership:
  - in some context(s), an instance is a member of class but in some other context(s), is not a member
- Excluded instances:
  - an instance of a class cannot be included as an instance of any of the immediate subclasses of that class
- Non-instance similarity:
  - the definition of a class or individual is similar across many dimensions to the definition of a class that it is not an instance of

"RDF is a very simple language, propositional in character, when viewed as a language for expressing knowledge. This puts a serious dent in its utility. DAML 'solves' this by imposing a somewhat artificial layering on top of RDF, to the point where DAML is both crippled by its foundations and in fact pretty distant from them."

Francis McCabe Fujitsu

http://lists.w3.org/Archives/Public/www-ws-arch/2002Aug/0162.html

"A number of people have become very worried about the layering of OWL on top of RDF and RDF Schema. The problems can be summarised as follows: RDF is not well suited as syntax carrier and RDF Schema has some unconventional features in its meta-model. Our conclusion is that these problems are much harder to solve than originally anticipated.

Therefore, we propose to take another route for specifying the syntax of OWL. The syntax of class and property definitions in OWL (the ontology) is specified in XML in such a way that RDF can be used to specify instances of the ontology so that significant parts of RDF Schema end up as a sublanguage of OWL.

The advantage of using XML are both technical and also political: XML is well suited for specifying syntax (in fact, that is its main goal in life). It comes with a host of additional technology and standards that can then be exploited for OWL (XLink, XPointer, XPath, XQuery, XSLT, etc). We can think of useful applications for all of these. It will make our work immediately relevant to all of the XML community. They share many of our goals, but there is a constant danger that they will use different (XML-based) technology, instead of RDF based technology."

Horrocks, Patel-Schneider, van Harmelen

http://lists.w3.org/Archives/Public/www-webont-wg/2002Jan/0005.html

"An XML-based language like RDF seems to be interesting because it allows sharing of ontologies on the web by using URI and namespaces but it is not expressive enough.

RDF-based languages like DAML+OIL are interesting because of the rich expressiveness to represent concepts and their relationships and also most common used axioms. The drawback of this language is just its readability." (DAML+OIL is the predecessor to OWL)

Myriam Ribière, Patricia Charlton

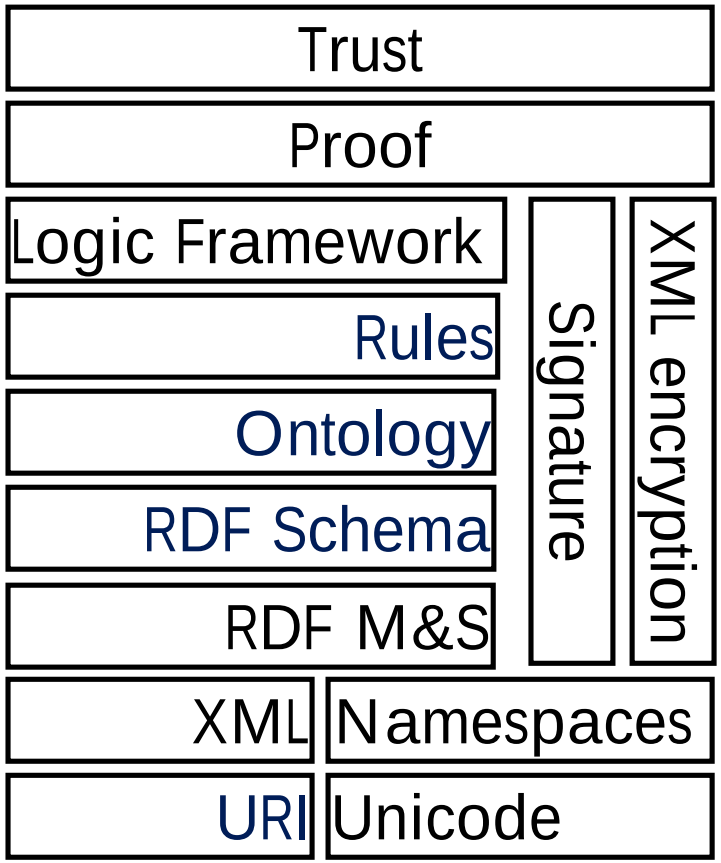
Motorola Labs Report on Ontology Languages

http://www.fipa.org/docs/input/f-in-00045/f-in-00045.pdf

Layer 6 – Rules

- People are experimenting with tools to process RDF using rules e.g. cwm but no standards activity yet
- Ontology languages like OWL can be implemented using rules or other approaches, so rules can be thought of as a more generalized version of OWL
- Prolog is an example of a language that works in a similar way
- Back in the 90s, the database community investigated "deductive databases" e.g. databases that used datalog languages that were similar to Prolog but were efficient over persistant stores. We face similiar problems when inferencing or using rules over persistant RDF stores

The Semantic Web layer cake



My position

- RDF, despite the complexity of the W3C specifications, is just a graph where arcs and optionally nodes are labelled with URIs
- Some applications could use RDF/XML or XML, but the latter currently has the advantage of a more mature set of tools.
- The RDF model theory provides “formal semantics” optimized for efficiency but with limited descriptive power
- RDF Schema also has limited descriptive power and cannot describe equivalence
- Languages such as OWL provide more descriptive power but mapping them on to RDF/XML adds unnecessary complexity
- There is a big overlap between the semantic web and areas like semi-structured and deductive databases, knowledge representation and artificial intelligence (although this is often denied!)
- The name “Semantic Web” is highly misleading: machines cannot “understand” data, “reason” or “interpret meaning”, they just process symbols!

"The fact that the programmer and the interpreter of the computer output use the symbols to stand for objects in the world is totally beyond the scope of the computer. The computer, to repeat, has a syntax but no semantics."

John Searle

Professor of Cognitive Science, University of California

http://members.aol.com/NeoNoetics/MindsBrainsPrograms.html

"There is a very basic, almost philosophical, point underlying this. In a very real sense, on the SW, there IS NO CONTENT. There is only formal language. The "content" is what the writers and readers of the languages intend, but there is no way to send an intention along a wire. When, as in the SW, some of the readers and writers are programs, they have no chance at all of guessing at all the subtleties that a human might have intended."

Pat Hayes
Editor, RDF Model Theory

http://lists.w3.org/Archives/Public/www-rdf-rules/2001Oct/0043.html

"Developing XML as a richer version of HTML was generally a good idea. But what botched the Semantic Web is that promoting a universal syntax does nothing to promote semantics.

To avoid further confusion, it would be a good idea to rename it the syntactic web."

John Sowa

Are there any semantics in the Semantic Web?

- People have semantics but machines do not
- However major advances in computer science have occurred because they make it easier for us to encode semantics so the way programs encode the real world is clearer
- Consider the changes between
  - assembly language programming
  - non-structured programming e.g. Basic
  - procedural programming e.g. Pascal
  - object oriented programming e.g. C++
- In the SW, someone has to provide a mapping to allow different vocabularies to interoperate
- The following statements are nonsense

  “RDF is more semantic than XML”

  “RDF allows us to reason concretely about the real world”

"25 years ago, Ed Feigenbaum described Terry Winograd's work (on Artificial Intelligence) as a "breakthrough in enthusiasm".

I worry that web services and the Semantic Web, in their reliance on effective computational semantics are vulnerable to the same criticism."

Henry S Thompson

http://www.ltg.ed.ac.uk/~ht/Web_Services_Glasgow.ppt

What are the standardisation and research issues?

- Need to bring the XML and RDF/XML standards closer together
- Demonstrate
  - how XML and RDF/XML can coexist and share tools
  - how OWL can be used to solve simple real world problems
  - benefits of using RDF for real world use cases e.g. interoperability
- Research
  - more efficient in memory and secondary storage representations of RDF
  - representations that support locking, transactions etc
  - simpler APIs for RDF
  - do we need quad models as they are better at representing context than triple models?
  - are ontology languages sufficient, or do we need more generalized rule languages instead?
  - devise a design methodology, similar to entity-relationship modelling, for modelling the real world using SW tools
  - compare and contrast with existing work on semistructured and deductive databases, artificial intelligence and object technologies

Other issues for the Semantic Web

- Realizing the Semantic Web vision is dependant on people and organizations making their data freely available by the Web in such a way it can be reused by others
  - This requires a major change in behaviour – "Open Source data"
  - Organizations, even non-profit ones, are often unwilling to do this as they see data as a critical part of their intellectual property. There is some variation between communities: geographic information system researchers make data available, but art historians do not
  - Some organizations give away information, but in a way that supports their revenue model e.g. via advertising. This is harder on the Semantic Web
  - People "reusing your data in unexpected ways" may have undesirable consequences e.g. spam, making it easier to commit identity fraud etc
- However as RSS demonstrates, if we can get people to make more data available, we can do some interesting things just by aggregating data from different sources, without using some of the more complex parts of the Semantic Web stack

Conclusions

- Some negatives
  - I think the title “Semantic Web” is unhelpful and confusing – the “symbolic web” would be much better
  - people have unrealistic expectations of what the Semantic Web will achieve e.g. Sci Am article
  - The current Semantic Web stack requires simplification
  - The W3C is running risk of unnecessary duplication of effort between XML and RDF
- Some positives
  - We need better ways to deal with heterogeneous, semi-structured data
  - Although symbolic AI did not live up to the hype, it was effective at solving problems in certain domains. Perhaps this work will encourage a re-evaluation of these techniques?
  - Techniques that make the mapping between programs, data and the real world itself more explicit are an important area of research for computer science

Further reading

- Which Semantic Web?
  http://www.csdl.tamu.edu/~marshall/ht03-sw-4.pdf
- The Semantic Web, Syllogism and WorldView
  http://www.shirky.com/writings/semantic_syllogism.html
- MetaCrap
  http://www.well.com/~doctorow/metacrap.htm
- XML and Databases
  http://rpbourret.com/xml/XMLAndDatabases.htm
- XML Data: From Research to Standards
  http://www.vldb.org/archive/vldb2000/tutorial_05.pdf
- An Introduction to RDF and Prolog
  http://www.xml.com/pub/a/2001/04/25/prologrdf/index.html
  http://www.xml.com/pub/a/2001/07/25/prologrdf.html
- Barriers to real world adoption of Semantic Web technologies
  http://www.hpl.hp.com/personal/marbut/barriersToRealWorldAdoptRDF.pdf


More machine processable than before
New, Improved
SEMANTIC
Web
Now with added meaning
May be incompatible with existing XML tools. Databases may take up to ten times as much memory and 24 hours to load.


hp
invent