Semantic Web

Semantic Web Architecture

Semantic Web Architecture

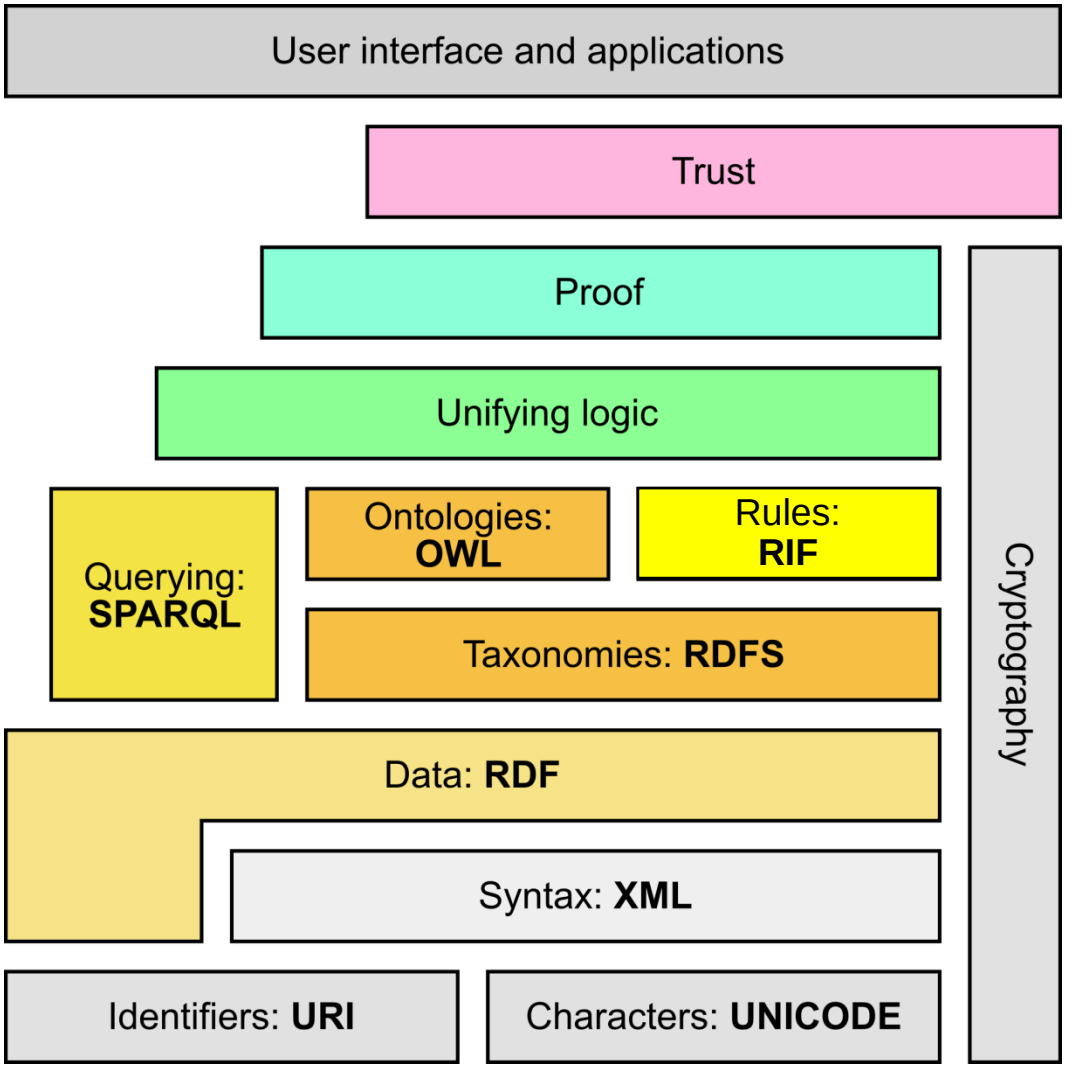
- Formalized components and their relationships
  - What technologies make up Semantic Web
  - What are the dependencies between components
- Roadmap for steps of developing the Semantic Web

SemWeb Architecture: Requirements

- Extensibility
  - Each layer should extend the previous one(s)
- Support for data interchange
  - Using data from one source in other applications
- Support for ontology description with different complexity
  - Including rules
- Support for data query
- Support for data provenance and trust evaluation

see the Semantic Web Roadmap: http://www.w3.org/DesignIssues/Semantic.html

Semantic Web Stack



Adapted from http://en.wikipedia.org/wiki/Semantic_Web_Stack

UNICODE, URI and XML

- UNICODE is the standard international character set
  - E.g. used to encode the data in the repository
- Uniform Resource Identifiers (URIs) identify things and concepts
  - E.g. used to indentify resources on the Web and in the repository
- eXtensible Markup Language (XML) is a markup language used for data exchange
  - E.g. format that can be wrapped into RDF and imported into the repository

RDF, RDFS and OWL

- Resource Description Framework (RDF) is the HTML of the Semantic Web
  - Simple way to describe resources on the Web
  - Based on triples <subject, predicate, object>
  - Various serializations, including one based on XML
  - A simple ontology language (RDFS)
  - E.g. language used to store the data in the repository
  - More in Lecture 3

- Web Ontology Language (OWL) is a more complex ontology language than RDFS
  - Layered language based on DL
  - Overcomes some RDF(S) limitations
  - E.g. ontology language used to define the schemas used in the repository
  - More in Lecture 7

SPARQL and Rule Languages

- SPARQL
  - Query language for RDF triples
  - A protocol for querying RDF data over the Web
  - E.g. language used to query the repository from the user interface
  - More in lecture 6

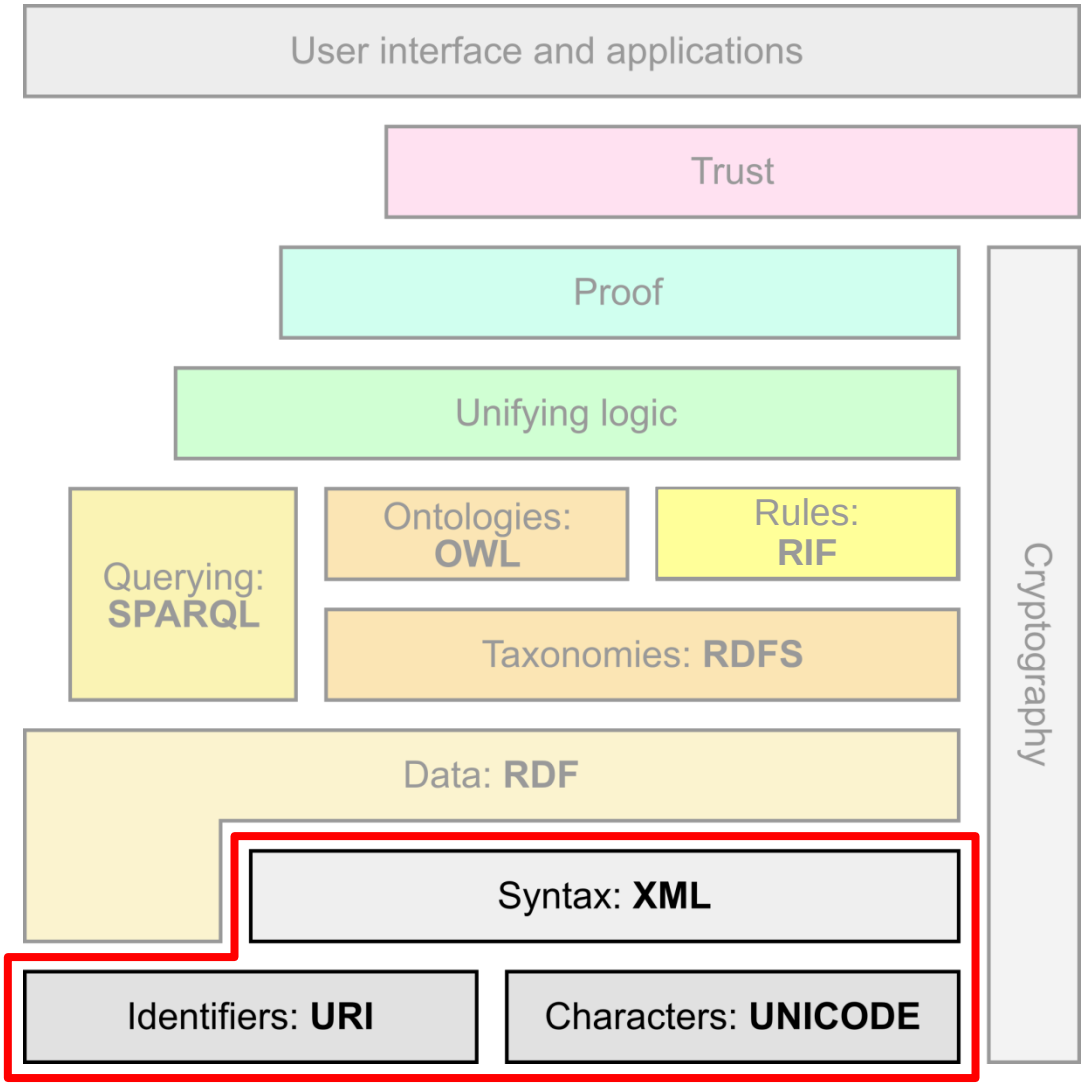
- Rule languages (esp. Rule Interchange Format RIF)
  - Extend ontology languages with proprietary axioms
  - Based on different types of logics
    - Description Logic
    - Logic Programming
  - E.g. used to enable reasoning over data to infer new knowledge
  - More in lecture 8

Logics, Proof and Trust

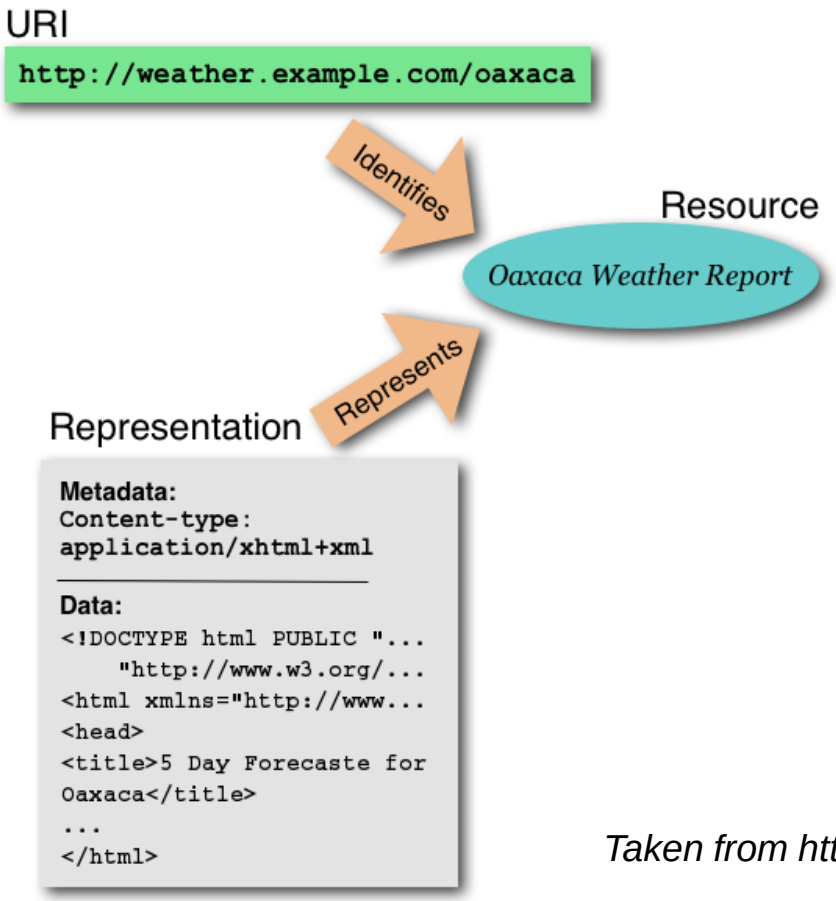
- Unifying logic
  - Bring together the various ontology and rule languages
  - Common inferences, meaning of data
- Proof
  - Explanation of inference results, data provenance
- Trust
  - Trust that the system performs correctly
  - Trust that the system can explain what it is doing
  - Network of trust for data sources and services
  - Technology and user interface

- Many open problems, topics for future research

Foundations



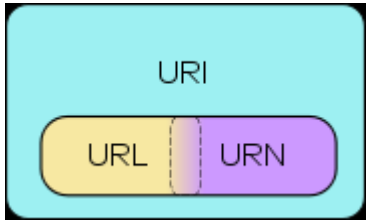
Identifier, Resource, Representation



Taken from http://www.w3.org/TR/webarch/

URI, URN, URL

- A Uniform Resource Identifier (URI) is a string of characters used to identify a name or a resource on the Internet



- A URI can be a URL or a URN
- A Uniform Resource Name (URN) defines an item's identity
  - the URN *urn:isbn:0-395-36341-1* is a URI that specifies the identifier system, i.e. International Standard Book Number (ISBN), as well as the unique reference within that system and allows one to talk about a book, but doesn't suggest where and how to obtain an actual copy of it
- A Uniform Resource Locator (URL) provides a method for finding it
  - the URL *http://www.sti-innsbruck.at/* identifies a resource (STI's home page) and implies that a representation of that resource (such as the home page's current HTML code, as encoded characters) is obtainable via HTTP from a network host named www.sti-innsbruck.at

URI Syntax

- Examples
  - http://www.ietf.org/rfc/rfc3986.txt
  - mailto:John.Doe@example.com
  - news:comp.infosystems.www.servers.unix
  - telnet://melvyl.ucop.edu/
- URI Syntax **scheme: [//authority] [/path] [?query] [#fragid]**
  - The scheme distinguishes different kinds of URIs
  - Authority normally identifies a server
  - Path normally identifies a directory and a file
  - Query adds extra parameters
  - Fragment ID identifies a *secondary resource*

URI Syntax cont'd

- Reserved characters (like /:?#@$&+*)
- Many allowed characters
- Rest percent-encoded from UTF-8
  - http://google.com/search?q=technikerstra%C3%9Fe

- IRI – Internationalized Resource Identifier
  - Allows whole UNICODE
  - Specifies transformation into URI – mostly UTF-8 encoding

URI Schemes

- Schemes partition the URI space into subspaces
- Schemes can add or clarify properties of resources
  - Ownership (how authorities are formed)
  - Persistence (how stable the URIs should be)
  - Protocol (default access protocol)

Scheme	Description	RFC
file	Host-specific file names	[1738]
ftp	File Transfer Protocol	[1738]
http	Hypertext Transfer Protocol	[2616]
https	Hypertext Transfer Protocol Secure	[2818]
im	Instant Messaging	[3860]
imap	internet message access protocol	[5092]
ipp	Internet Printing Protocol	[3510]
iris	Internet Registry Information Service	[3981]
ldap	Lightweight Directory Access Protocol	[4516]
mailto	Electronic mail address	[2368]
mid	message identifier	[2392]

From http://www.iana.org/assignments/uri-schemes.html

Questions?