Emerging Semantic Web Applications for the Life Sciences

Ken Baclawski
Northeastern University

Introduction

- The web is a versatile infrastructure for basic data availability.
- The main emphasis is on human-mediated interactions via web browsers.
- As the amount of data increases, there is a need for more automated integration of data and tools.
- In this talk we discuss the emerging products and applications that address this need in the life sciences.

Some Application Areas

- Interoperability and integration of legacy systems
- Semantic search
- Web services and composite applications
- Collaboration tools
- Medical records management
- Uncertain, incomplete and conflicting information
- Situation awareness and simulation

Analyzing an application domain

- Domain knowledge
  - Technical background
  - Community organization
- Identify urgent needs
- Understand the trends
  - Short-term evolution
  - Possible paradigm shifts
- Recognize an opportunity

Interoperability of legacy systems

- Legacy systems and databases are characterized by:
  - A large variety of formats
  - High degree of complexity
  - Many technologies of various ages
- Need to interoperate and integrate
- Trend is toward encoding more semantics in the data representation itself
- Opportunity to develop products and services for interoperability and integration.

Record Structures

- A flat file is a collection of records.
- A record consists of fields.
- Each record in a flat file has the same number and kinds of fields as any other record in the same file.
- The schema of a flat file describes the structure (i.e., the kinds of fields) of each record.
- A schema is an example of an *ontology*.

The eXtensible Markup Language

- XML is a format for representing data.
- XML goes beyond flat files by allowing elements to contain other elements, forming a hierarchy.

XML	Flat Files
Element	Record
Attribute	Field
DTD	Schema

XML Element Hierarchy

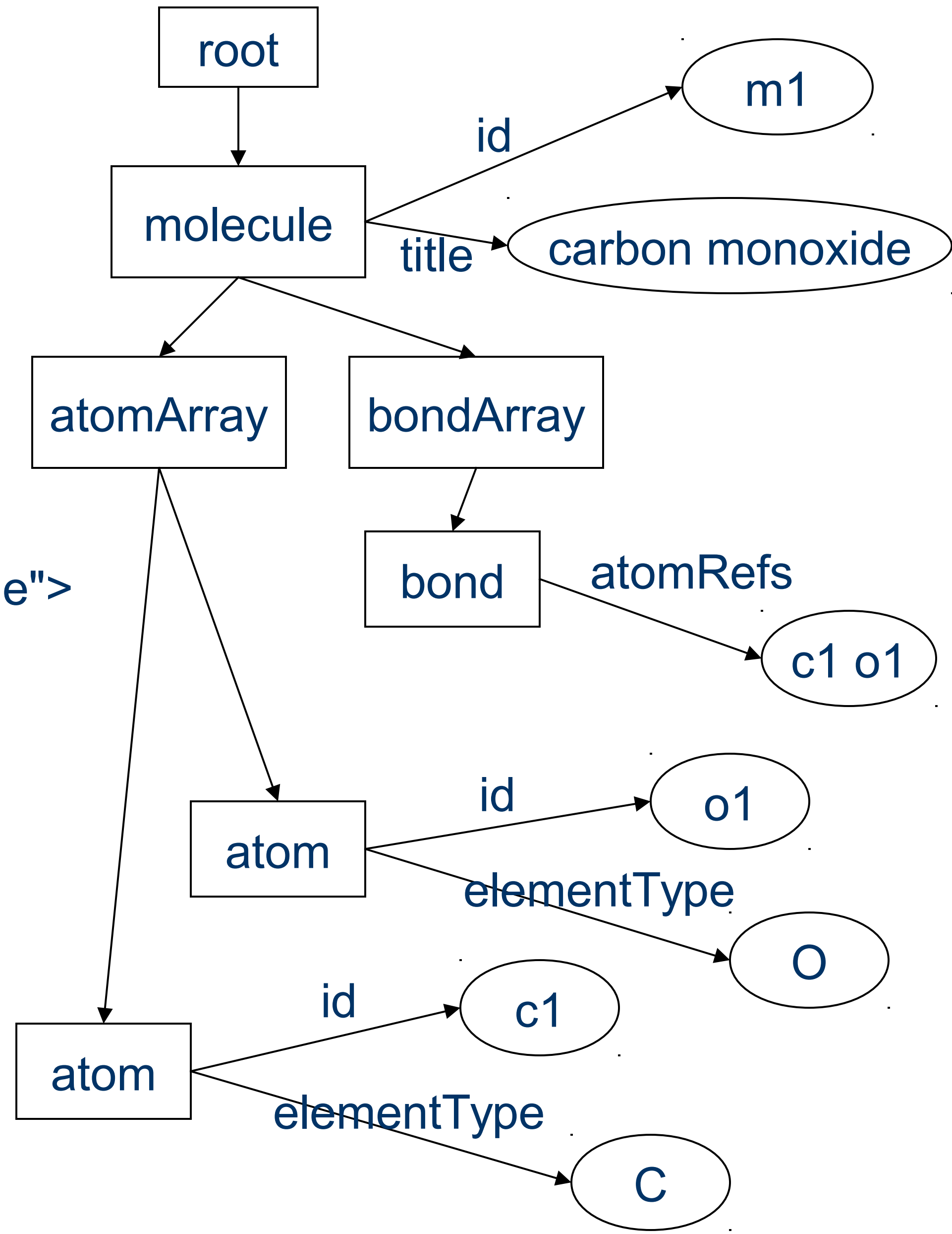
```
<bioml>
  <organism name="Homo sapiens (human)">
    <chromosome name="Chromosome 11" number="11">
      <locus name="HUMINS locus">
        <reference name="Sequence databases">
          <db_entry name="Genbank sequence" entry="v00565" format="GENBANK"/>
          <db_entry name="EMBL sequence" format="EMBL" entry="V00565"/>
        </reference>
        <gene name="Insulin gene">
          <dna name="Complete HUMINS sequence" start="1" end="4992">
          1 ctcgaggggc ctagacattg ccctccagag agagcaccca acaccctcca ggcttgaccg
          ...
          </dna>
          <ddomain name="flanking domain" start="1" end="2185"/>
          <ddomain name="polymorphic domain" start="1340" end="1823"/>
          <ddomain name="Signal peptide" start="2424" end="2495"/>
          <exon name="Exon 1" start="2186" end="2227"/>
          <intron name="Intron 1" start="2228" end="2406"/>
        </gene>
      </locus>
      <locus>
      ...
      </locus>
    </chromosome>
  </organism>
</bioml>
```

Formal Semantics

- Semantics is primarily concerned with *sameness*. It determines that two entities are the same in spite of appearing to be different.
- Number semantics: 5.1, 5.10 and 05.1 are all the same number.
- DNA sequence semantics: cctggacct is the same as CCTGGACCT.
- XML document semantics is defined by infosets.

XML infoset for carbon monoxide

```
<molecule id="m1" title="carbon monoxide">
  <atomArray>
    <atom id="c1" elementType="C"/>
    <atom id="o1" elementType="O"/>
  </atomArray>
  <bondArray>
    <bond atomRefs="c1 o1"/>
  </bondArray>
</molecule>
```



The Resource Description Framework

- RDF is a language for representing information about resources in the web.
- While RDF is expressed in XML, it has different semantics.
- Many tools exist for RDF, but it does not yet have the same level of support as XML.

XSD vs. RDF

- XML semantics based on infosets
- Easy to convert from DTD to XSD
- Support for data structures and types
- Element order is part of the semantics

- Different semantics based on RDF graphs
- Cannot easily convert from DTD to RDF
- Uses only XSD basic data types
- Ordering must be explicitly specified using a collection construct

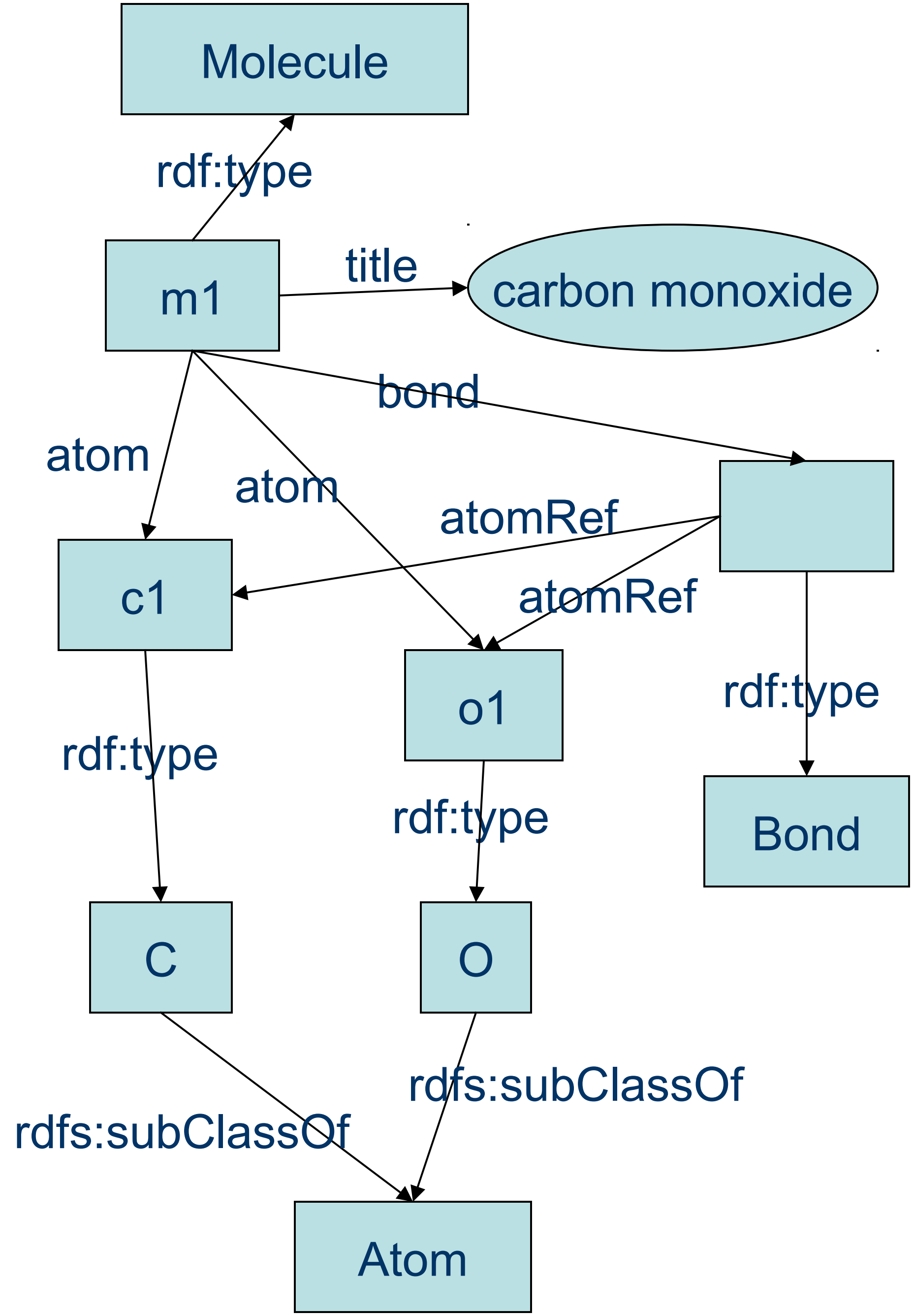
RDF Semantics

- All relationships are explicit and labeled with a property resource.
- The distinction in XML between attribute and containment is dropped, but the containment relationship must be labeled on a separate level. This is called *striping.*

```
...
    <locus name="HUMINS locus">
      <contains>
        <gene name="Insulin gene">
          <isStoredIn>
            <db_entry name="Genbank sequence" entry="v00565"
                      format="GENBANK"/>
            <db_entry name="EMBL sequence" format="EMBL"
                      entry="V00565"/>
          </isStoredIn>
          <isCitedBy>
            <db_entry name="Insulin gene sequence" format="MEDLINE"
                      entry="80120725"/>
            <db_entry name="Insulin mRNA sequence" format="MEDLINE"
                      entry="80236313"/>
            <db_entry name="Localization to Chromosome 11" format="MEDLINE"
                      entry="93364428"/>
          </isCitedBy>
          <hasSequence>
            <dna name="Complete HUMINS sequence" start="1" end="4992">
              1 ctcgaggggc ctagacattg ccctccagag agagcaccca acaccctcca ggcttgaccg
              ...
            </dna>
          </hasSequence>
        </gene>
      </contains>
    </locus>
...
```

RDF graph for carbon monoxide

```
<Molecule rdf:id=“m1”
      title=“carbon monoxide”>
 <atom>
  <C rdf:id=“c1"/>
  <O rdf:id=“o1“/>
 </atom>
 <bond>
  <Bond>
   <atomRef rdf:resource=“c1”/>
   <atomRef rdf:resource=“o1”/>
  </Bond
 </bond>
</Molecule>
```



RDF Triples

- RDF graphs consist of edges called *triples* because they have three components: subject, predicate and object.
- The semantics of RDF is determined by the set of triples that are explicitly asserted or inferred.
- In the chemical example, some of the triples are:
  - (m1, rdf:type, cml:Molecule)
  - (m1, cml:title, “carbon monoxide”)
  - (m1, cml:atom, c1)
  - (m1, cml:atom, o1)
- Notice that properties are many-to-many relationships.

Notes on RDF Semantics

- There is no easy way to convert from XML to RDF because RDF makes explicit many relationships that are implicit in XML.
- In the chemical example, the element types are classes in RDF but have no special meaning to XML.
- The fact that n1 is an atom can be inferred from the fact that N is a subclass of Atom.
- The ordering of atoms in a molecule is significant in XML but not in RDF. RDF is therefore closer to the correct semantics.

The Web Ontology Language

- OWL is based on RDF and has three increasingly general levels: OWL Lite, OWL-DL, and OWL Full.
- OWL adds many new features to RDF:
  - Functional properties
  - Inverse functional properties (database keys)
  - Local domain and range constraints
  - General cardinality constraints
  - Inverse properties
  - Symmetric and transitive properties

Class Constructors

- OWL classes can be constructed from other classes in a variety of ways:
  - Intersection (Boolean AND)
  - Union (Boolean OR)
  - Complement (Boolean NOT)
  - Restriction
- Class construction is the basis for *description logic.*

Description Logic Example

- Concepts are generally defined in terms of other concepts. For example:

  The iridocorneal endothelial syndrome (ICE) is a disease characterized by corneal endothelium proliferation and migration, iris atrophy, corneal oedema and/or pigmentary iris nevi.

- ICE-Syndrome class is the intersection of:
  - The set of all diseases
  - The set of things that have at least one of the four symptoms

```
<owl:Class rdf:ID="ICE-Syndrome">
  <owl:intersectionOf parseType="Collection">
    <owl:Class rdf:about="#Disease"/>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#has-symptom"/>
      <owl:someValuesFrom>
        <owl:Class rdf:ID="ICE-Symptoms">
          <owl:oneOf parseType="Collection">
            <Symptom name="corneal endothelium proliferation and migration"/>
            <Symptom name="iris atrophy"/>
            <Symptom name="corneal oedema"/>
            <Symptom name="pigmentary iris nevi"/>
          </owl:oneOf>
        </owl:Class>
      </owl:someValuesFrom>
    </owl:Restriction>
  </owl:intersectionOf>
</owl:Class>
```

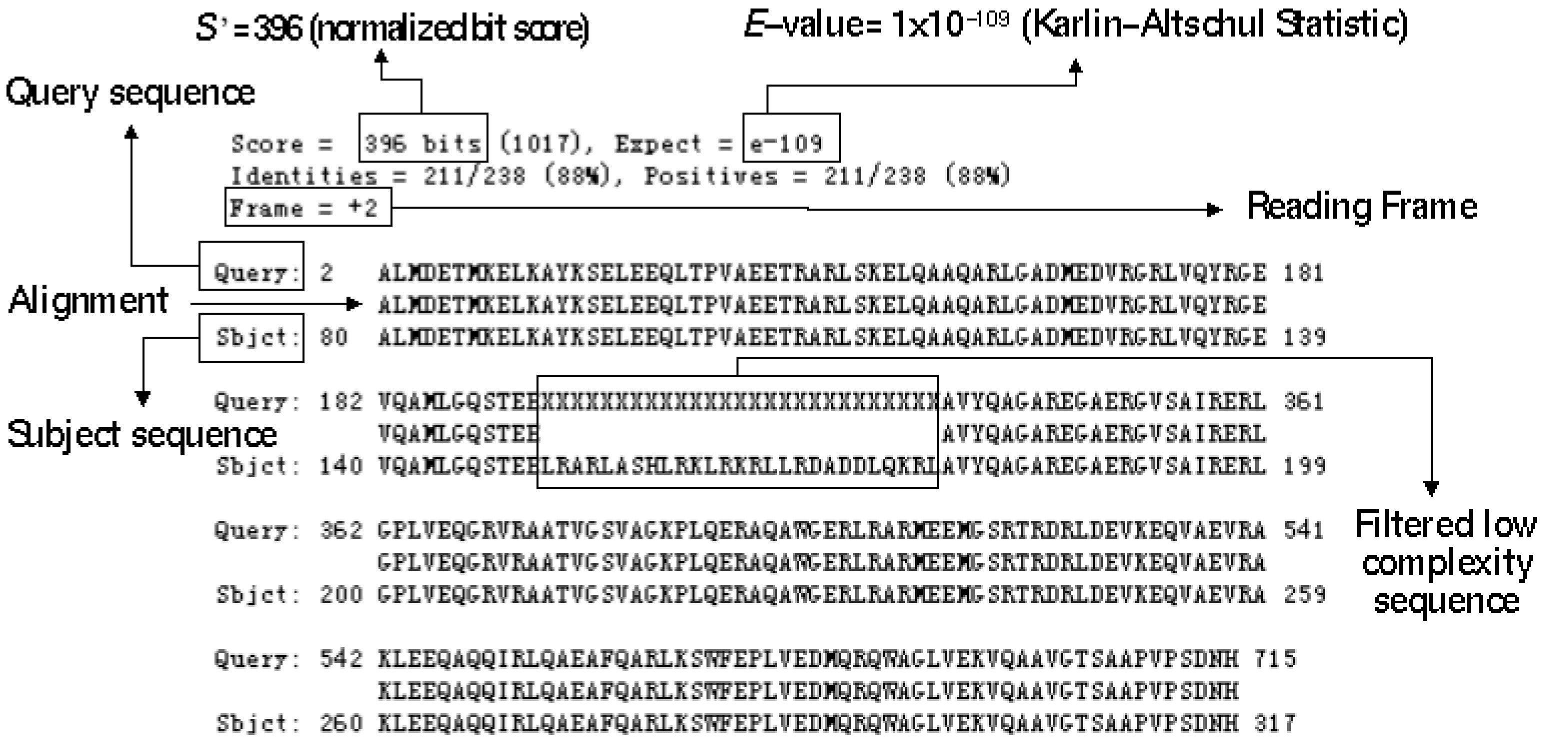
Example of Description Logic

OWL Semantics

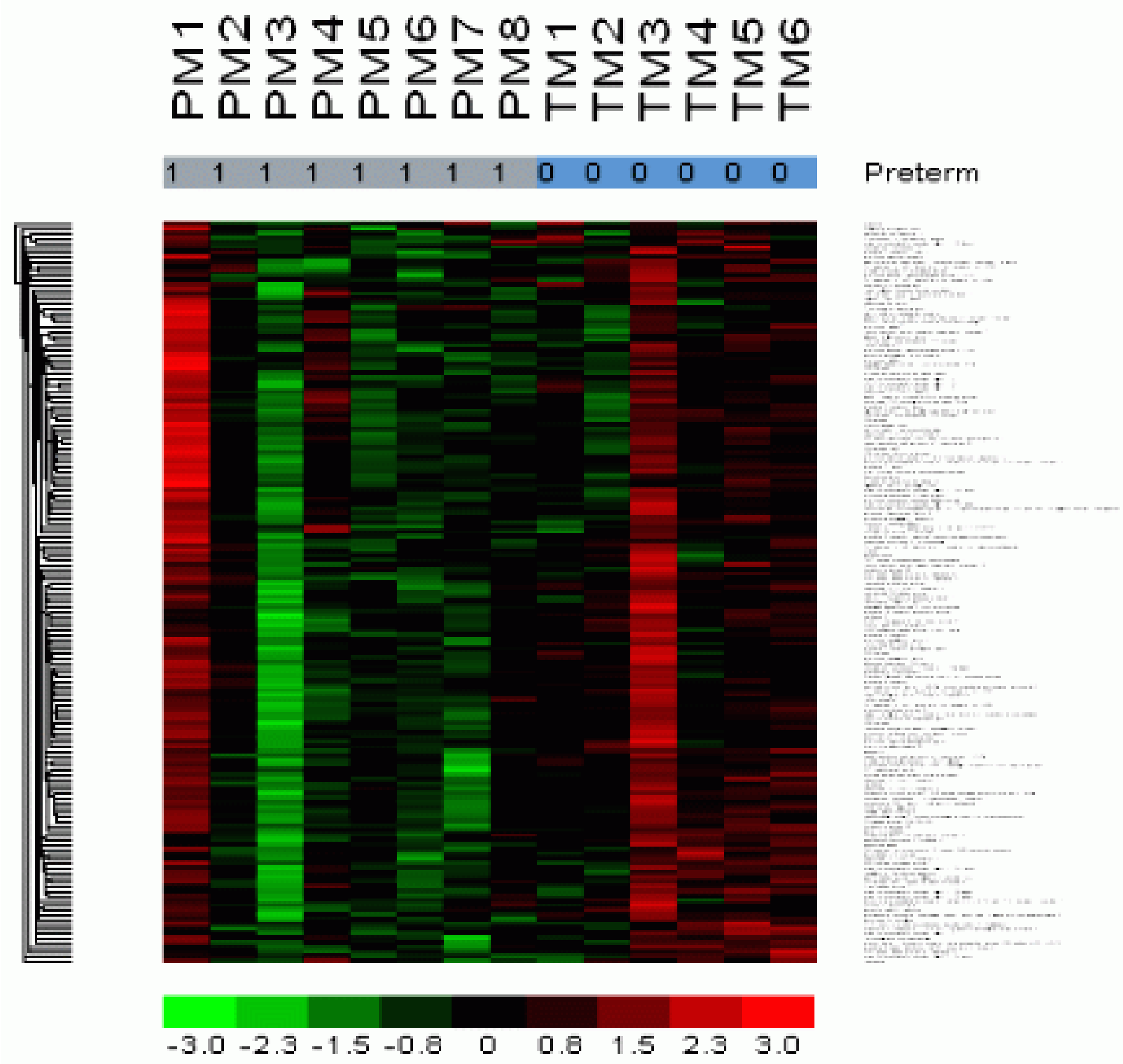
- An OWL ontology defines a theory of the world. States of the world that are consistent with the theory are called *interpretations* of the theory.
- A fact that is true in every model is said to be *entailed* by the theory. Logical inference in OWL is defined by entailment.
- Entailment can be counter-intuitive, especially when it entails that two resources are the same.

Search and retrieval

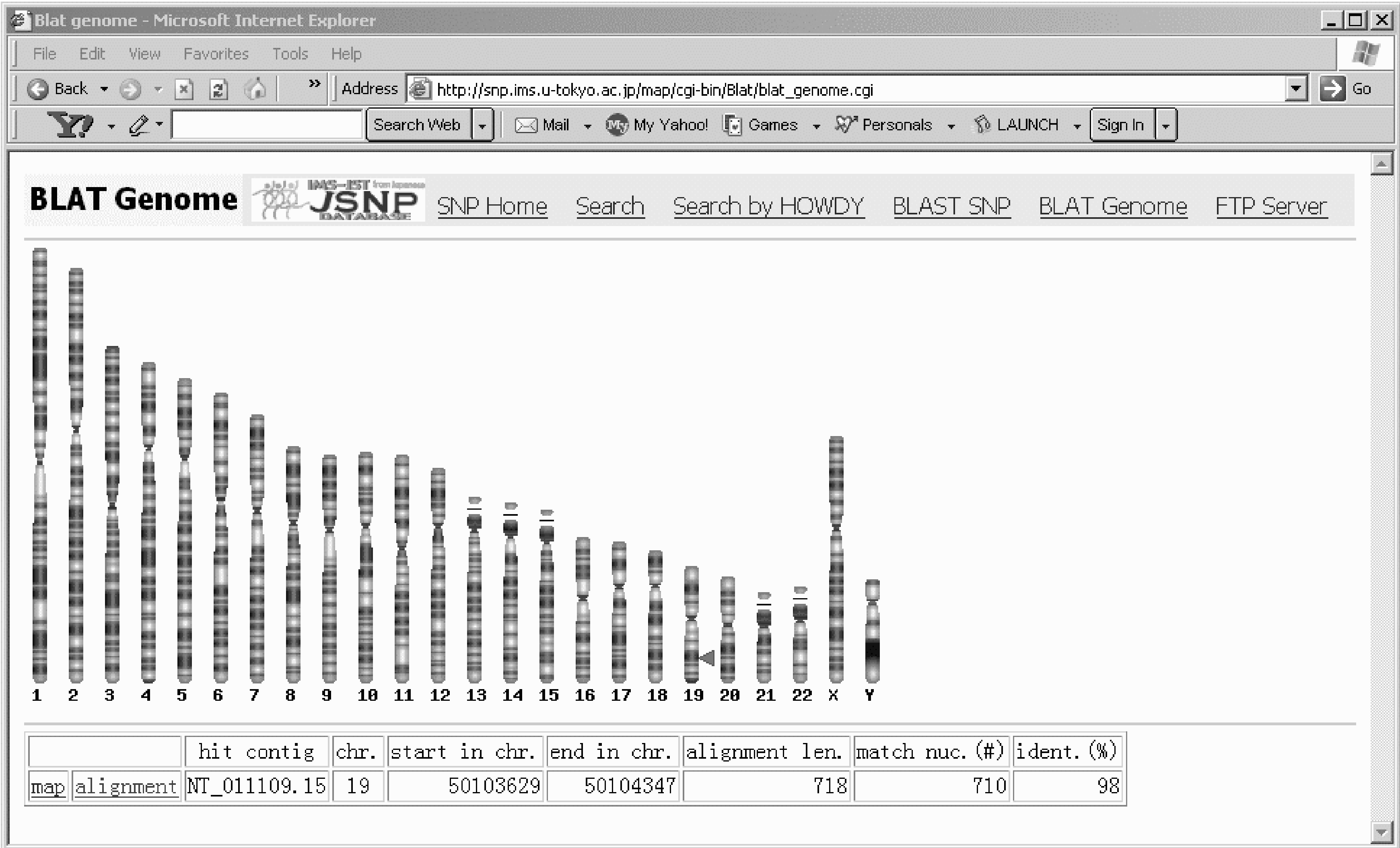
- Data is typically stored in either record/data structures or natural language.
- Need is to search and retrieve both kinds of data for a single query.
- There are several trends.
  - More semantics
  - Integration with other services
- Opportunities are mostly determined by the other services.



Example of a complex data format



Example of a complex data format



Output window for a bioinformatics web service

Ontologies for Information Retrieval

- Source of terminology
- RDF graph matching
- Queries based on formal logic

Using Ontologies for Formulating Queries

- Ontologies are an important source of terminology that can be used to formulate queries.
- Biological and medical ontologies can be so large and complex that specialized browsing and retrieval tools are necessary.
- Several browsers are now available for the UMLS: MeSH, Know-ME, Apelon DTS, SKIP, etc.
- One can use ontologies as a means of query modification when a query does not return satisfactory results.

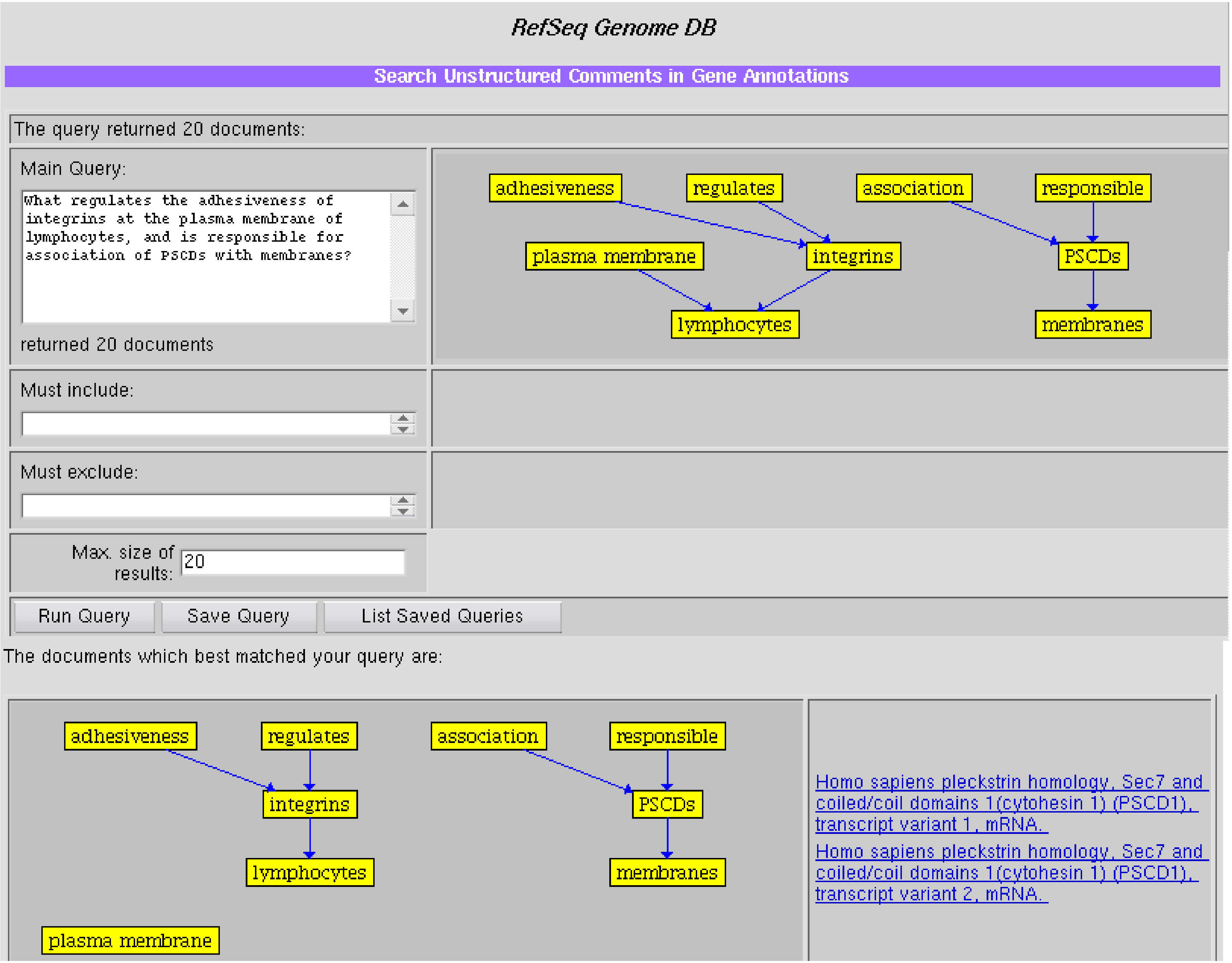
RDF Graph Matching

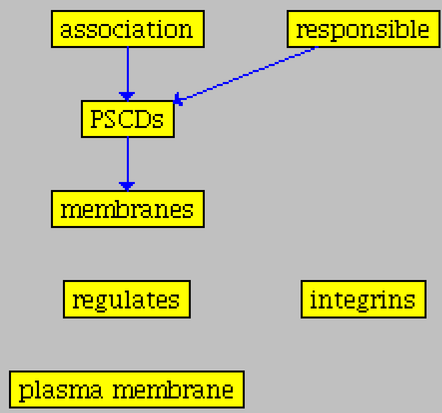
- Graph matching is analogous to sequence matching, such as in BLAST.
- Translating natural language text to an RDF graph that captures meaning remains an unsolved problem, but reasonably good tools are available.
- Systems that use RDF graph matching are available. Such a system allows one to query a corpus such as PubMed using natural language.

RefSeq Genome DB

Search Unstructured Comments in Gene Annotations

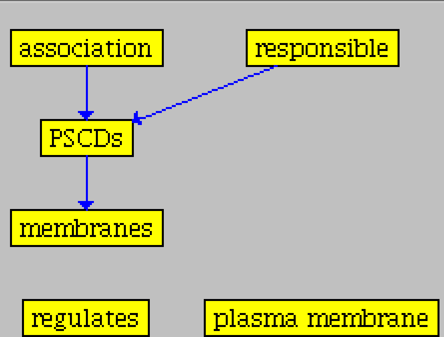
Enter your query here:	What regulates the adhesiveness of integrins at the plasma membrane of lymphocytes, and is responsible for association of PSCDs with membranes?
Must **include**:	
Must **exclude:**	
Max. size of results:	20
	Run Query / List Saved Queries


RefSeq Genome DB
Search Unstructured Comments in Gene Annotations
The query returned 20 documents:
Main Query:
What regulates the adhesiveness of integrins at the plasma membrane of lymphocytes, and is responsible for association of PSCDs with membranes?
returned 20 documents
adhesiveness
regulates
association
responsible
plasma membrane
integrins
PSCDs
lymphocytes
membranes
Must include:
Must exclude:
Max. size of results:
20
Run Query
Save Query
List Saved Queries
The documents which best matched your query are:
adhesiveness
regulates
association
responsible
integrins
PSCDs
lymphocytes
membranes
plasma membrane
Homo sapiens pleckstrin homology, Sec7 and coiled/coil domains 1(cytohesin 1) (PSCD1), transcript variant 1, mRNA.
Homo sapiens pleckstrin homology, Sec7 and coiled/coil domains 1(cytohesin 1) (PSCD1), transcript variant 2, mRNA.

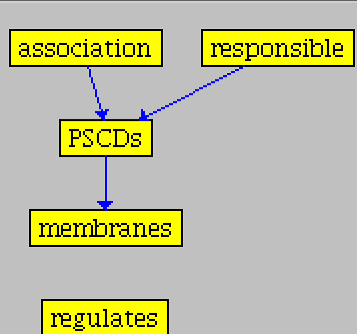


Homo sapiens pleckstrin homology, Sec7 and coiled/coil domains 2 (cytohesin-2) (PSCD2), transcript variant 2, mRNA.

Homo sapiens pleckstrin homology, Sec7 and coiled/coil domains 2 (cytohesin-2) (PSCD2), transcript variant 1, mRNA.



Homo sapiens pleckstrin homology, Sec7 and coiled/coil domains 3 (PSCD3), mRNA.



Homo sapiens pleckstrin homology, Sec7 and coiled/coil domains 4 (PSCD4), mRNA.

Web Query Languages

Ontology language	Query Language	Remarks
XML DTD and XSD	XQuery	Combines document navigation with an SQL-like query language
RDF and OWL	SparQL	Similar to SQL, specialized to the case of a 3-column table

Querying XML Using XQuery

- XQuery is the standard query language for processing XML documents.
- Every XPath expression is a valid query.
- A general query is made of four kinds of clause:
  - A **for** clause scans the result of an XPath expression, one node at a time.
  - A **where** clause selects which of the nodes scanned by the for clauses are to be used.
  - A **return** clause specifies the output of the query.
  - A **let** clause sets a variable to an intermediate result.

In the PubMed database, find all citations dealing with the therapeutic use of glutethimide. More precisely, find the citations that have"glutethimide" as a major topic descriptor, qualified by "therapeutic use."

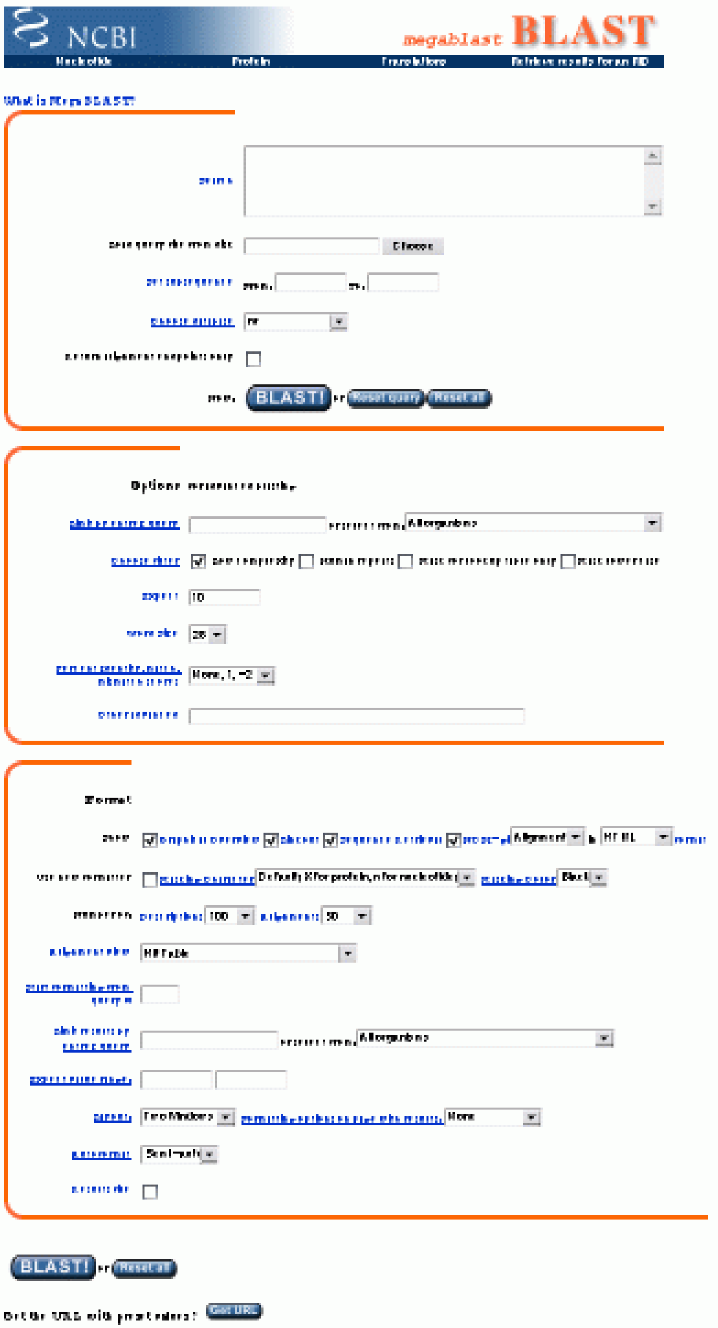
```
for $citation in document("pubmed.xml")//MedlineCitation
where exists
 (for $heading in $citation//MeshHeading
  where $heading/DescriptorName/@MajorTopicYN="Y"
  and $heading/DescriptorName="Glutethimide"
  and $heading/QualifierName="therapeutic use"
  return $heading)
return $citation
```

Example of an PubMed query using XQuery

Web services and composite applications

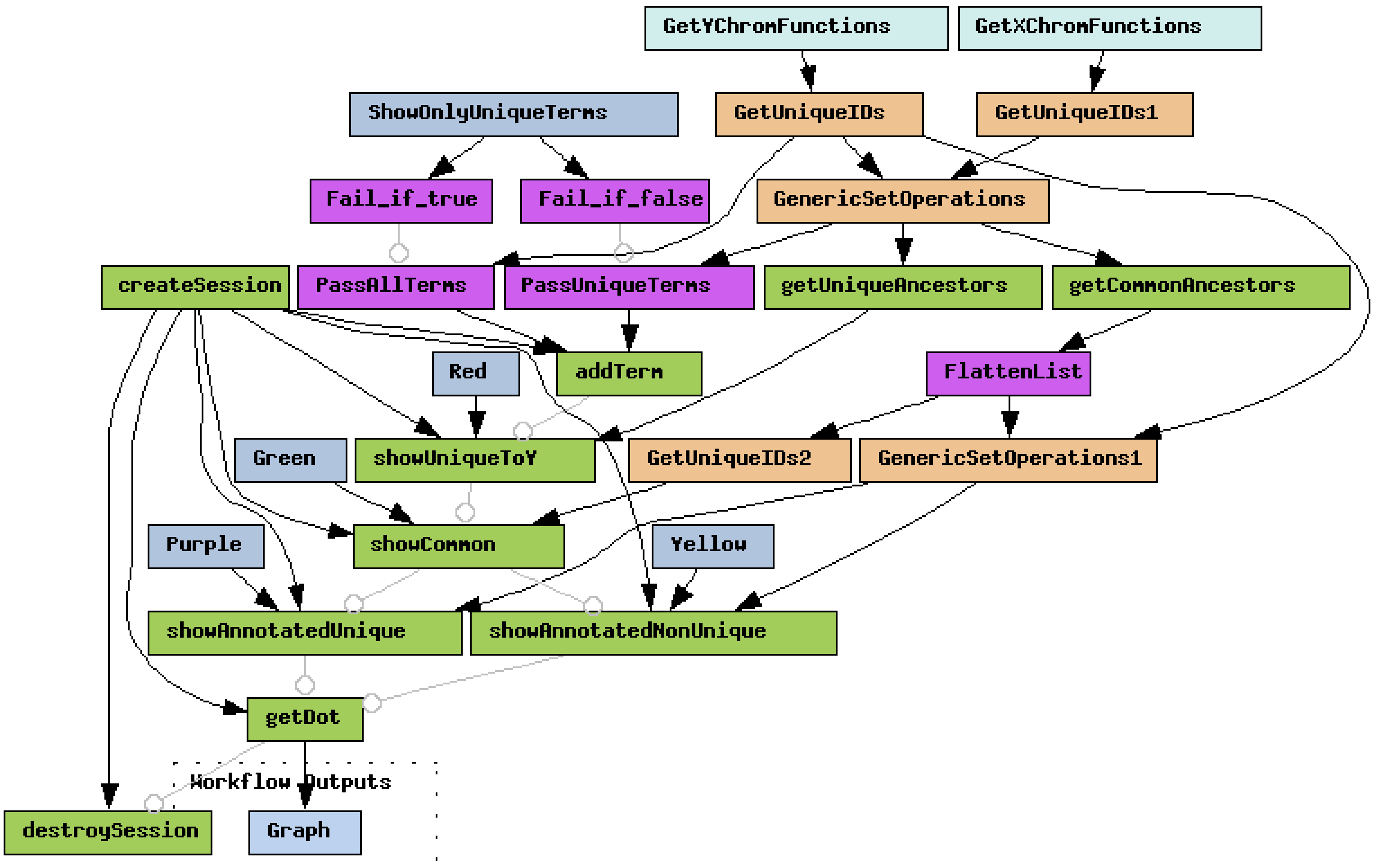
- The web is being used not only for retrieval of data but also for using tools and services.
- The need is to find the required services, and to get them to communicate with each other.
- The trend is to use semantic annotation to describe/advertise services, to express requests, and to represent the responses, but very unevenly.
- The opportunity is to built agile workflow management tools that can deal with the differing levels of semantic annotation.

Example of a typical interface for a bioinformatics web service



The myGrid Project

- Taverna workbench supports the scientific process for in silico experiments.
  - Management
  - Sharing and reusing results
  - Recording their provenance and the methods used to generate results
- Workflows link together third party and local resources using database queries and web service protocols.



MyGrid Workflow

Web Services

- In the traditional programming model, all processing is done locally.
- Web services allow one to use programs that run on other machines.
- Tools have been developed that greatly reduce the effort of developing and offering web services.

Traditional Programming

- Traditional approach to application development
  - Write the program in a language such as Perl.
  - Compile the program.
  - Place the program and auxiliary files in a location where they can be found and downloaded.
- Using the application requires these steps:
  - Find the URL of the program.
  - Download the compiled program and auxiliary files.
  - Run it on the command line (or by clicking on an icon), specifying options as needed.

Web Services Approach

- Web service development has some new steps:
  - Write the program in some language like Perl.
  - Compile the program.
  - Describe the program using the WSDL.
  - Provide the application as a web service using a SOAP tool.
- Using the application proceeds as follows:
  - Find the URL of the web service.
  - Run on the command line using a WSIF tool.

Automating Transformations

- Reconciling differing terminology has many names depending on the particular context where it is done, such as: ontology mediation, schema integration, data warehousing, virtual data integration, query discovery, and schema matching.
- Automated ontology mediation systems attempt to reduce manual effort, but they rarely provide a net gain.
- Most automated ontology mediation systems are still research prototypes.

Standards for Web Services

- Web Service Definition Language (WSDL)
  - Defines the web service
  - Written by the developer
- Simple Object Access Protocol (SOAP)
  - Format for running a service and receiving results
  - SOAP tools hide the underlying format and protocol from the developer and client.
- Universal Description, Discovery and Integration (UDDI)
  - Mechanism for advertising web services

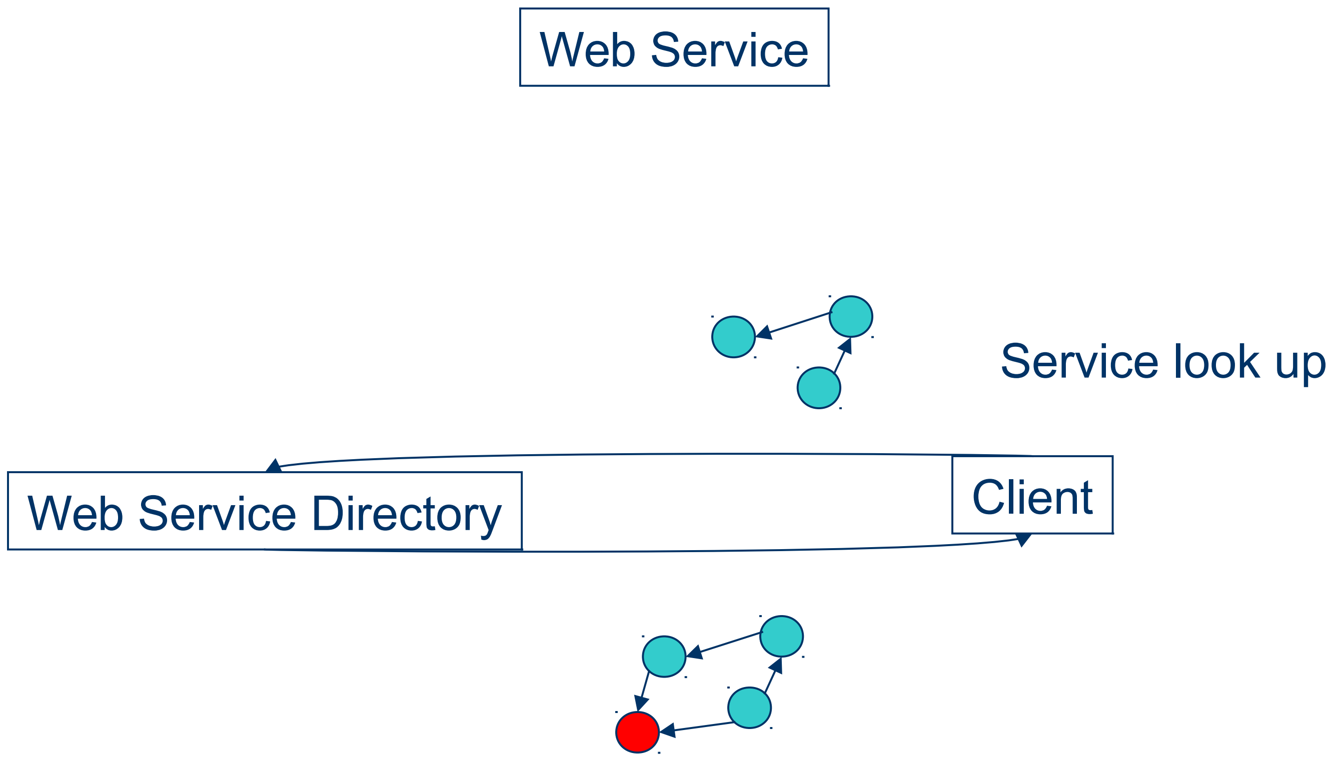
Semantic Web for Web Services

- Service Discovery
  - UDDI uses informal natural language descriptions.
  - The descriptions should be specified using ontologies.
- Service Definition
  - WSDL uses XSD for defining services, options and parameters.
  - RDF or OWL can be used to express the services using terminology in an ontology.

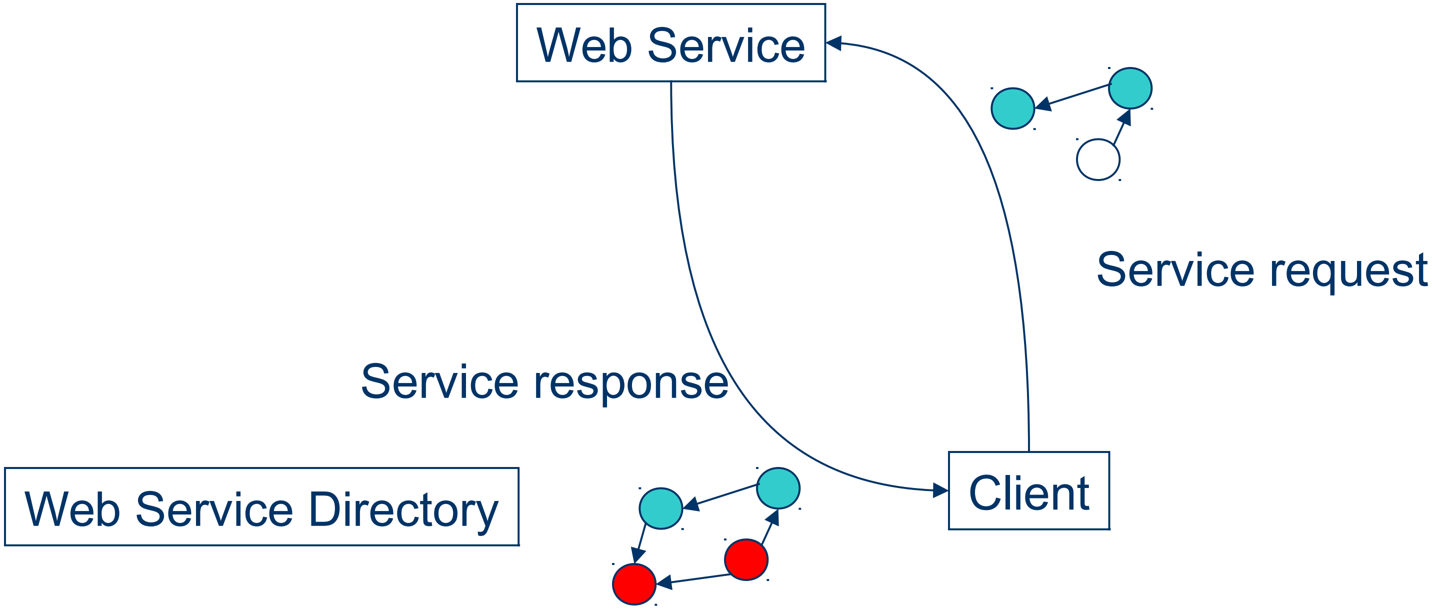
Semantic MOBY

- This is an infrastructure for semantic web services that evolved out of the popular BioMOBY system.
- The Semantic MOBY was developed for semantic interoperability and integration of plant genetics.
- Semantic MOBY uses the same OWL graphs for finding services, making requests and sending replies.

Architecture of Semantic MOBY I



Architecture of Semantic MOBY II



Collaboration tools

- People need to collaborate to solve problems.
- The need is to support rapid team formation and problem solving even when the people are geographically dispersed.
- The trend is to use wikis and blogs rather than face-to-face meetings.
- The opportunity is to develop tools that facilitate collaboration over the web without losing the advantages of face-to-face meetings that make them desirable.

Wikis

- Wikis are a popular tool for collaboration.
- They have been used for rapid team formation and collaboration.
- They have a number of disadvantages:
  - Mix of natural language and untyped links.
  - Focus is on simplicity and presentation, not structure and semantics.

Semantic Wikis

- A wiki with an underlying knowledge model (ontology) is a *semantic wiki.*
- Data in the wiki is annotated with meta-data in RDF or OWL.
- Links are typed and annotated, also in RDF or OWL.
- Machines can infer new facts from the explicitly asserted facts.
- Search and retrieval are facilitated by the semantics.
- Interoperability is greatly improved.

Web conferencing

- Online conferencing is not new, but is it becoming much easier to arrange.
- Another important feature of web conferencing is the integration with wikis.
- The trend is toward the use of semantic wikis.
- Visual and small group dynamics are still difficult to support.

Blogs

- While a blog is not usually regarded as a collaboration tool, communities of blogs have the effect of a distributed collaboration.
- The need is for support of distributed collaboration that is less focused and controlled than wikis but having more credibility and consistency than blogs.
- Ontologies could provide the consistency and focus that blogs usually lack.

Medical records management

- Significant efforts are new being undertaken to transform the US health care delivery system.
- Between 44,000 and 98,000 Americans die each year from medical errors.
- Early in 2004, President Bush called for widespread adoption of interoperable electronic health records within the next 10 years.
- Over 80% of health care providers in the US in 2005 did not have electronic health record systems. Of the systems that do exist few are interoperable.
- (Quoted from Health IT in Government: Transforming health care and empowering citizens. Health IT in Government)

Medical records issues

- Solving the electronic health record problem will add little to the existing paper-based records if the systems are not interoperable.
- Simply automating paper-based processes has relatively little impact on productivity.
- Gains in efficiency and improved patient care require a change in the overall process of medical care delivery.

Formulating the problem

- The Health IT problem is currently defined in terms of providing electronic health *records.*
- This effectively mandates a solution that is closely tied with existing workflows and processes.
- This is likely to have the unintended consequence of imposing inappropriate workflows and rigidly defined processes.

Medical Records Opportunity

- Develop medical event ontologies that:
  - Support interoperability
  - Are independent of workflows and processes
  - Are compatible with existing processes
- Develop products that:
  - Assist medical organizations to evolve toward electronic data management
  - Serve the interests of many stakeholders

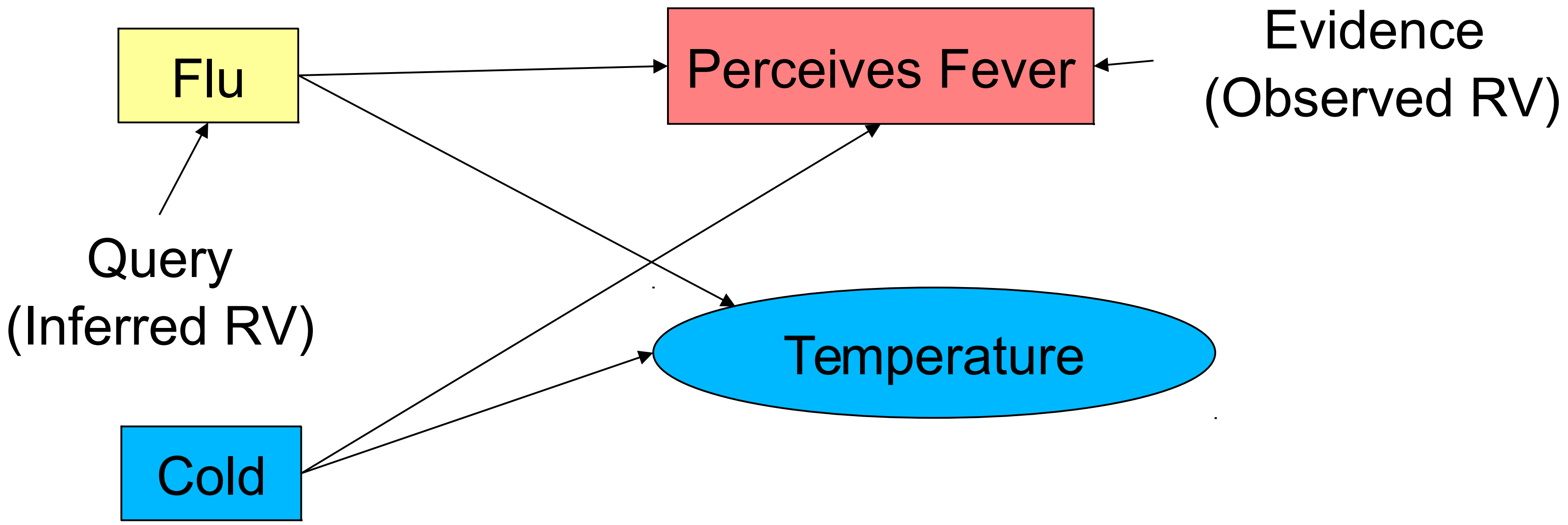
Reasoning with uncertainty

- The Semantic Web is an extension of the current web in which information is given well defined meaning… (Berners-Lee, Hendler & Lassila)
- The Semantic Web is based on formal logic for which one can only assert facts that are unambiguously certain.
- Unfortunately, there are many sources of uncertainty, such as measurements, unmodeled variables, and subjectivity.

The Bayesian Web

- The challenge is to develop a full-featured stochastic reasoning infrastructure, comparable to the logical reasoning infrastructure of the Semantic Web.
- The *Bayesian Web* is a proposal to add reasoning about uncertainty to the Semantic Web.

Bayesian Network Inference



Inference is performed by observing some RVs (evidence) and computing the distribution of the RVs of interest (query).
The evidence can be a value or a probability distribution.
The BN combines the evidence probability distributions even when there are probabilistic dependencies.

Bayesian Web facilities

- Common interchange format
- Ability to refer to common variables (diseases, drugs, ...)
- Context specification
- Authentication and trust
- Open hierarchy of probability distribution types
- Component based construction of BNs
- BN inference engines
- Meta-analysis services

Bayesian Web Capabilities

- Use a BN developed by another group as easily as navigating from one Web page to another.
- Perform stochastic inference using information from one source and a BN from another.
- Combine BNs from the same or different sources.
- Reconcile and validate BNs.

Situation awareness and simulation

- Sensor technology is making it possible for a single person to have access to large amounts of data about the local environment.
- One need is to organize this information and present it so that it enhances situation awareness and contributes to decision making.
- Another important need is to attempt to predict the future by simulating various scenarios, weighted by their likelihood.

Medical device evolution

- Biomedical devices produce larger amounts of data.
  - High-thruput screening
  - Microarrays
  - Radiological and medical imaging devices
- There is a need for such devices to interoperate, sometimes ad hoc.
- The trend is from hardware to software, and software is evolving to make the devices more self-aware.

Device evolution

- The trend for devices has been to move functionality from hardware to software as improvements in processor speed has made this feasible.
- Software has evolved from special purpose, hand-crafted programs to programs built from standard components.
- More recently, software is becoming *self-aware:*
  - They know their own structure.
  - They can query their own state and capabilities during operation.
  - They can dynamically reconfigure and reprogram themselves.
- The Semantic Web adds flexibility, inferencing and reasoning features that are not available with ad hoc data structures or database schemas.

Situation awareness

- Situation awareness (SAW) is “knowing what is going on around oneself.”
  - More precisely, SAW is the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future (Endsley & Garland)
- SAW is part of a larger process known as *data fusion.*

The various levels where information can be combined has been standardized by the Joint Defense Laboratories (JDL) model. The whole process is called *data fusion*.

Level	Name	Process	Estimation	Product	Medicine
0	Signal Assessment	Identify features	Detection	Signal State	Observation
1	Object Assessment	Identify entities	Attributive State	Entity State	Symptom
2	Situation Assessment	Relationships among entities	Relation	Situation State	Diagnosis
3	Impact Assessment	Evaluation	Game Theory	Situation Utility	Prognosis

Opportunities

- To develop semantic data fusion tools for biomedicine that support researchers and clinicians in the task of situation awareness (diagnosis) and impact assessment (prognosis).
- Some examples of applications of such a tool include:
  - Tracking epidemics
  - Monitoring the patient during surgery
  - Meta-analysis services for researchers
  - Assessing the health of populations by region or recognized group