Theory and Practice of Agent Communication in the Semantic Web Era

Tim Finin

University of Maryland, Baltimore County

Yannis Labrou

Fujitsu Laboratories of America

UMBC
an Honors University in Maryland

http://agents.umbc.edu/tutorials/aamas03/t7/

Course Objectives

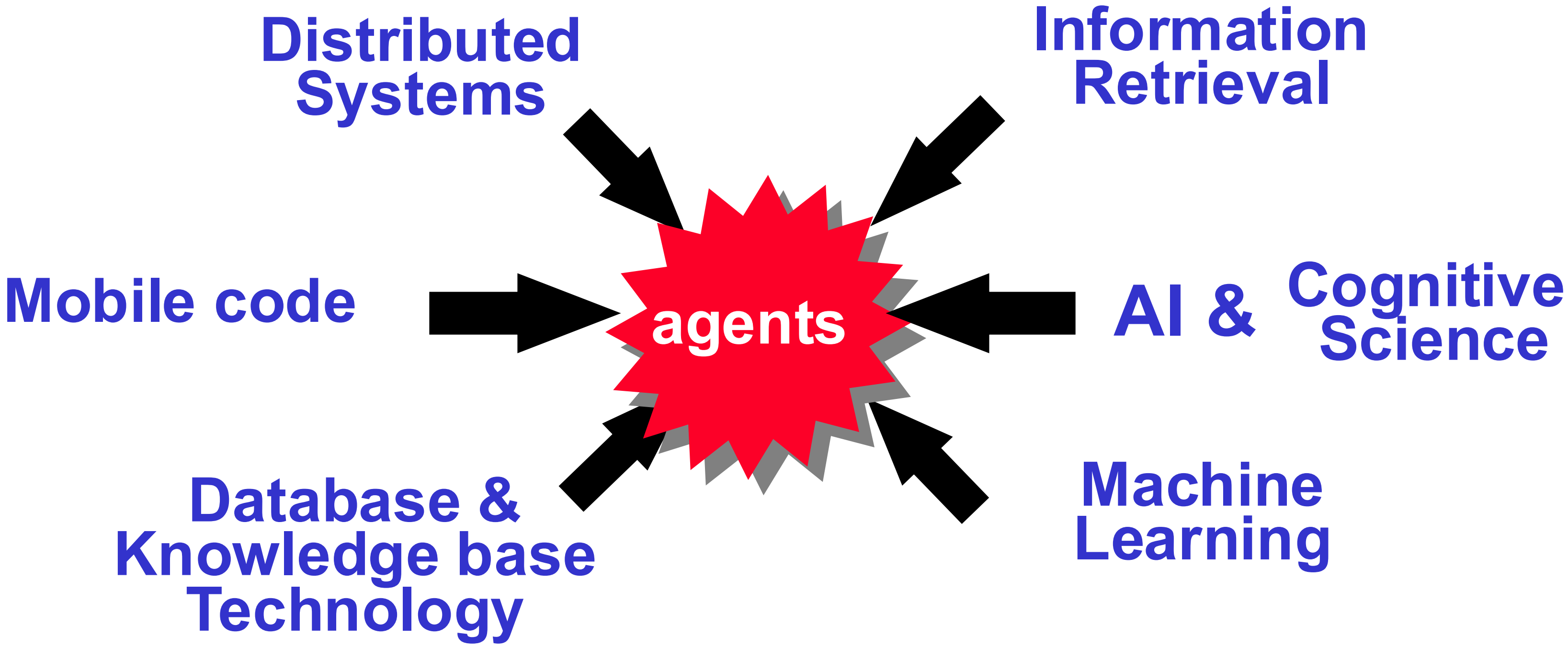
- Present the general requirements of Agent Communication Languages (ACLs) and the Semantic Web (SW)
- Sketch the conceptual and theoretical underpinnings of ACLs and SW
- Describe current ACLs and SW languages and their realizations in software implementations
- FIPA standardization efforts
- Review agent-based projects which are using some of the ACL components discussed
- Review applications and software tools for the Semantic Web
- Discuss directions in the practice of integrating agent and Semantic Web technologies

Outline

1. Introduction: Agents, Agent Communication and the Semantic Web
2. Conceptual & Theoretical Foundations
3. Origins – Knowledge Sharing Effort
4. Standardization - FIPA
5. ACL Semantics
6. ACLs and the Semantic Web
7. Semantic Web
8. Applications
9. Software Tools and Research Issues
10. Conclusions
11. Q & A

1
Introduction to Agents, Agent Communication & the Semantic Web

Agents: A system-building paradigm



agents

Distributed Systems

Information Retrieval

Mobile code

AI & Cognitive Science

Database & Knowledge base Technology

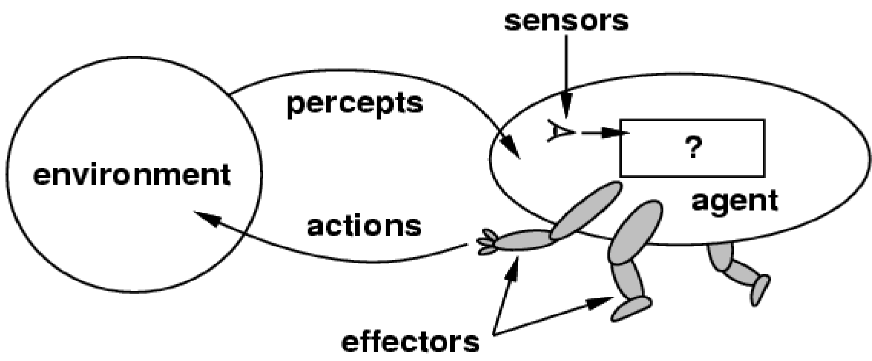
Machine Learning

$$\frac{\text{agents}}{2003} = \frac{\text{objects}}{1982} = \frac{\text{structured programming}}{1974}$$

Software agents

Several key properties are important to the agent paradigm. Agents are:

- **Autonomous,** taking the initiative as appropriate
- **Goal-directed**, maintaining an agenda of goals which it pursues until accomplished or believed impossible.
- **Taskable:** one agent can delegate rights/actions to another.
- **Situated** in an environment (computational and/or physical) which it is aware of and reacts to.
- **Cooperative** with other agents (software or human) to accomplish its tasks.
- **Communicative** with other or software)
- **Adaptive**, modifying beliefs behavior based on experience



Communication

- Communication almost always means “communication in a common language”
- Understanding a “common language” involves:
  - **Semantics:** understanding of its vocabulary, i.e., understanding of the meaning of its tokens
  - **Syntax:** knowing the rules for combining the tokens into well formed expressions
  - **Pragmatics:** knowing how to effectively use the vocabulary to perform tasks, achieve goals, effect one’s environment, etc.
- For software agents, an Agent Communication Language (ACL) is concerned with all three.

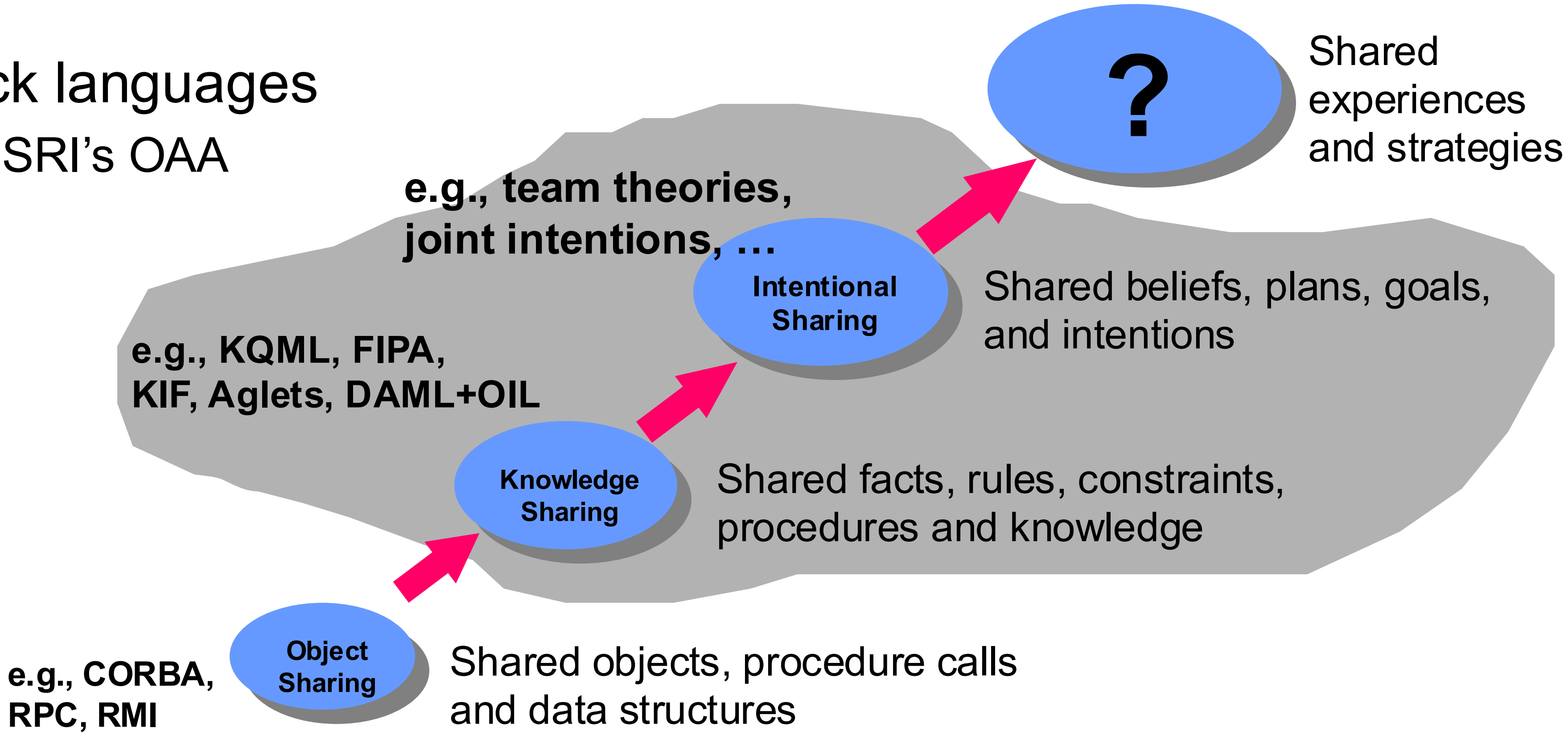
Agent Communication

- Agent-to-agent communication is key to realizing the potential of the agent paradigm, just as the development of human language was key to the development of human intelligence and societies.
- Agents use an ***Agent Communication Language*** (ACL) to communication information and knowledge.
- Genesereth (CACM 1994) defined a software agent as any system which uses an ACL to exchange information.

Genesereth, M. R., Ketchpel, S. P.: "Software Agents", Communications of the Association for Computing Machinery, July 1994, pp 48-53.

Some ACLs

- Knowledge sharing approach
  - KQML, KIF, Ontologies
- FIPA
- Ad hock languages
  - e.g., SRI's OAA



Shared experiences and strategies

e.g., team theories, joint intentions, …

Shared beliefs, plans, goals, and intentions

e.g., KQML, FIPA, KIF, Aglets, DAML+OIL

Shared facts, rules, constraints, procedures and knowledge

e.g., CORBA, RPC, RMI

Shared objects, procedure calls and data structures

To communicate is to manipulate a “common language”

- Effective agent communication involves two aspects:
  - possessing the understanding of a “common language”, as humans do for various domains and tasks
  - using the common language in order to achieve tasks and goals, and to effect an agent’s environment
- The understanding of the meaning of the tokens of a language is the substrate for any form of communication.
- Understanding the tokens alone, does not mean ability to communicate; the use of (any) language is driven by a purpose.

Agent Communication, at the technical level

- Messages are transported using some lower-level transport protocol (SMTP,TCP/IP, HTTP, IIOP, etc.)
- An Agent Communication Language (ACL) defines the types of messages (and their meaning) that agents may exchange.
- Over time, agents engage in “conversations.” Such interaction protocols (negotiation, auction, etc.), defines task-oriented, shared sequences of messages.
- Some higher-level conceptualization of an agent’s goals and strategies drives the agent’s communicative (and non-communicative) behavior.

What is the Semantic Web?

- **Focus on machine consumption:**

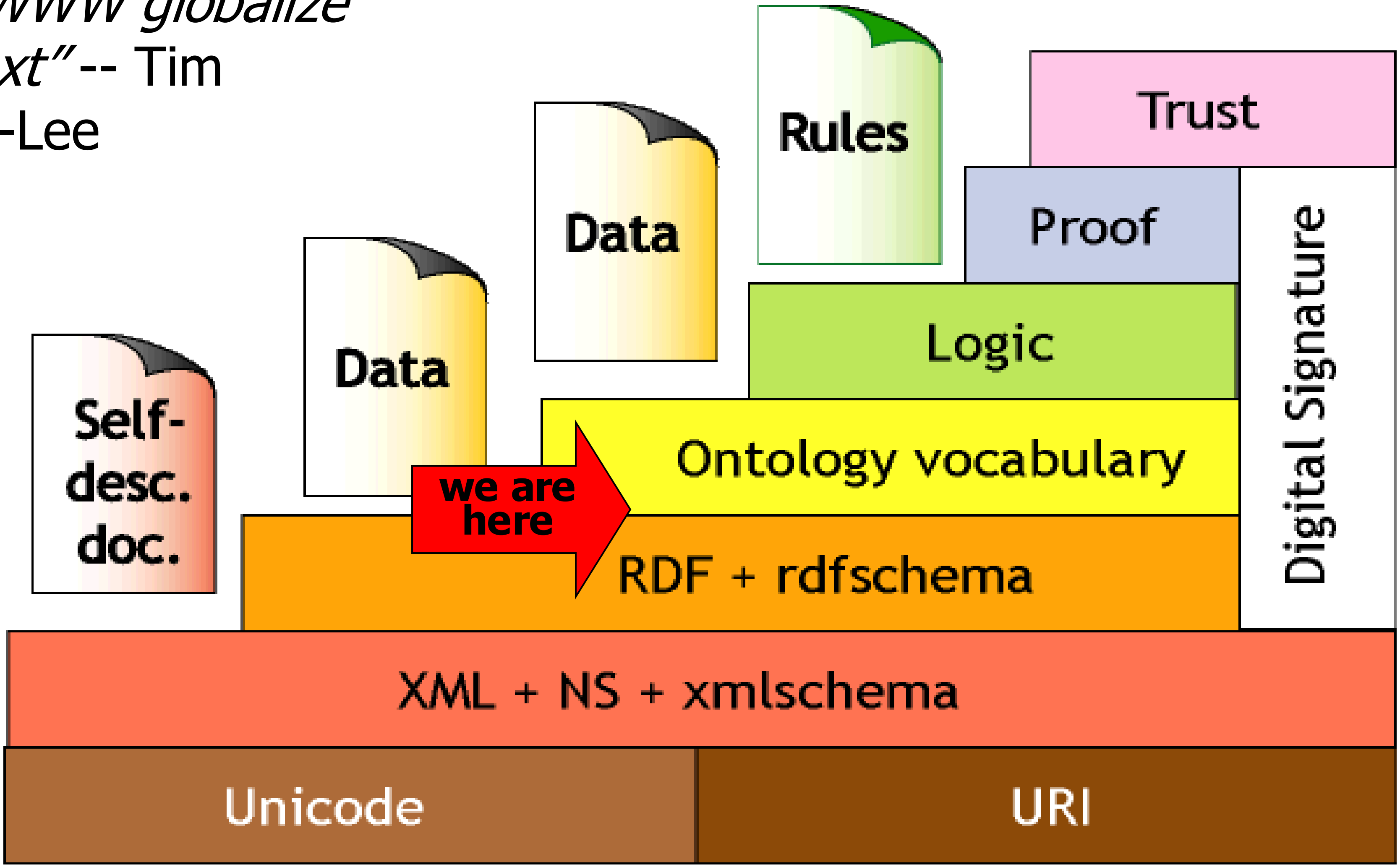
  "The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation."
  -- Berners-Lee, Hendler and Lassila, The Semantic Web, Scientific American, 2001

- **The current Web *stores* things whereas the semantic Web *does* things.**

=> *Good match for agents!*

TBL's semantic web vision

"The Semantic Web will globalize KR, just as the WWW globalize hypertext" -- Tim Berners-Lee



UMBC
an Honors University in Maryland

Semantic Web

- We’ll argue that the semantic web (SW) provides a good approach, languages and tools to support the development of intelligent agents.
- This isn’t obvious, since the SW seems grounded in the “traditional” wired web.
  - But, the principles which drive it are the right ones for multi-agent systems.
  - And, by grounding agents in web technology, they may make it out of the lab.
- SW languages are being used in MAS as (1) content languages, (2) message encodings, and (3) service description languages.

2
Conceptual and Theoretical Foundations

Historical Note: Knowledge Sharing Effort

- Initiated by DARPA circa 1990

  The DARPA Knowledge Sharing Effort: Progress Report, Patil et. al., KR'92, http://citeseer.nj.nec.com/patil93darpa.html

- Sponsored by DARPA, NSF, AFOSR, etc.
- Participation by dozens of researchers in academia and industry.
- Developing techniques, methodologies and software tools for *knowledge sharing* and *knowledge reuse*.
- Sharing and reuse can occur at *design*, *implementation* or *execution* time.

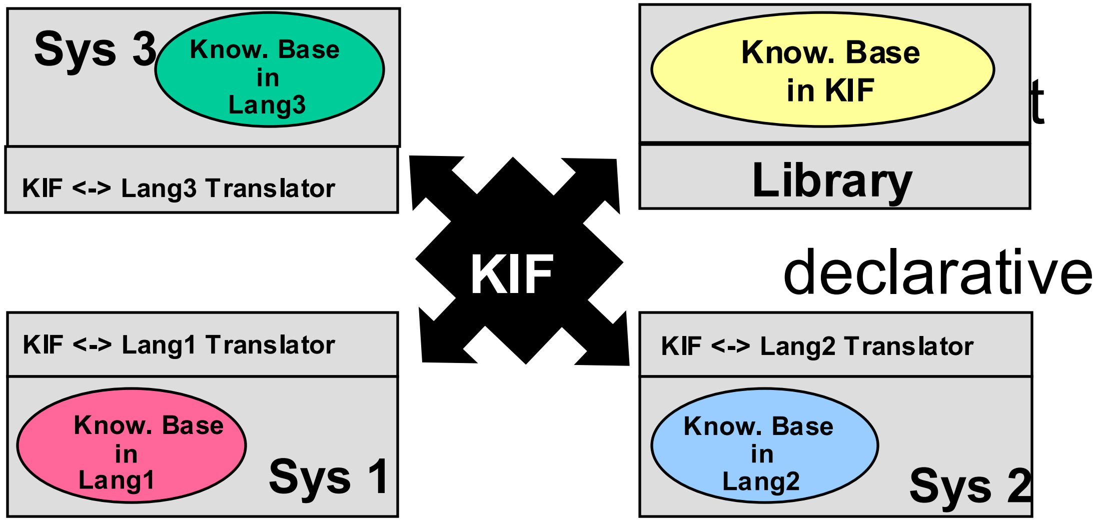
Knowledge Sharing Effort

- Knowledge sharing requires a communication
- … which requires a common language
- We can divide a language into syntax, semantics, and pragmatics
- Some existing components that can be used independently or together:
  - **KIF** - knowledge Interchange Format *(syntax)*
  - **Ontolingua** - a language for defining sharable ontologies *(semantics)*
  - **KQML** - a high-level interaction language *(pragmatics)*

Propositional

Propositional attitudes

Knowledge Interchange Format



- KIF ~ First order logic theory
- An **interlingua** for encoded knowledge
  - Takes translation among n from $O(n^2)$ to $O(n)$
- Common language for reusable knowledge
  - Implementation independent semantics
  - Highly expressive - can represent knowledge in typical application KBs.
  - Translatable - into and out of typical application languages
  - Human readable - good for publishing reference models and ontologies.
- Current specification at http://logic.stanford.edu/kif/ now being superceded by an effort to specify and standardize “Common Logic” (http://cl.tamu.edu/)

Common Semantics
Shared Ontologies and Ontolingua

- ***Ontology***: A common vocabulary and agreed upon meanings to describe a subject domain.
- Ontolingua is a language for building, publishing, and sharing ontologies.
  - A web-based interface to a browser/editor server.
  - Ontologies can be automatically translated into other content languages, including KIF, LOOM, Prolog, etc.
  - The language includes primitives for combining ontologies.

Common Pragmatics
Knowledge Query and Manipulation Language

- KQML is a high-level, message-oriented, communication language and protocol for information exchange independent of content syntax and ontology.
- KQML is also independent of
  - transport mechanism, e.g., tcp/ip, email, corba, IIOP, http ...
  - High level protocols, e.g., Contract Net, Auctions, …
- Each KQML message represents a single *speech act* (e.g., ask, tell, achieve, …) with an associated semantics and protocol.
- KQML includes primitive message types of particular interest to building interesting agent architectures (e.g., for mediators, sharing intentions, etc.)

Common High-level Protocols

- There is also a need for communication agents to agree on the agent-level protocols they will use.
- The protocol is often conveyed via an extra parameter on a message
  - (ask :from Alice :to Bob … **:protocol auction42** …)
- Common protocols:
  - Contract net
  - Various auction protocols
  - Name registration
- These protocols are often defined in terms of constraints on possible conversations and can be expressed as
  - Grammars (e.g., DFAs, ATNs, DCGs…)
  - Petri networks
  - UML-like interaction (activity) diagrams
  - Conversation plans
  - Rules or axioms

Common Service Infrastructure

- Many agent systems assume a common set of services such as:
  - Agent Name Sever
  - Broker or Facilitator
  - Communication visualizer
  - Certificate server
- These are often tied rather closely to an ACL since a given service is implemented to speak a single ACL
- Moreover, some of the services (e.g., name registration) may be logically ACL-dependent
  - e.g., Some ACLs don't have a notion of an agent's name and others have elaborate systems of naming

Conceptual and Theoretical Foundations I

Speech Act Theory and BDI Theories

The intentional level, BDI theories, speech acts and ACLs: how do they all fit together?

- ACL have message types that are usually modeled after speech acts
- Speech acts may be understood in terms of an intentional-level description of an agent
- An intentional description makes references to beliefs, desires, intentions and other modalities
- BDI frameworks have the power to describe an agents’ behavior, including communicative behavior

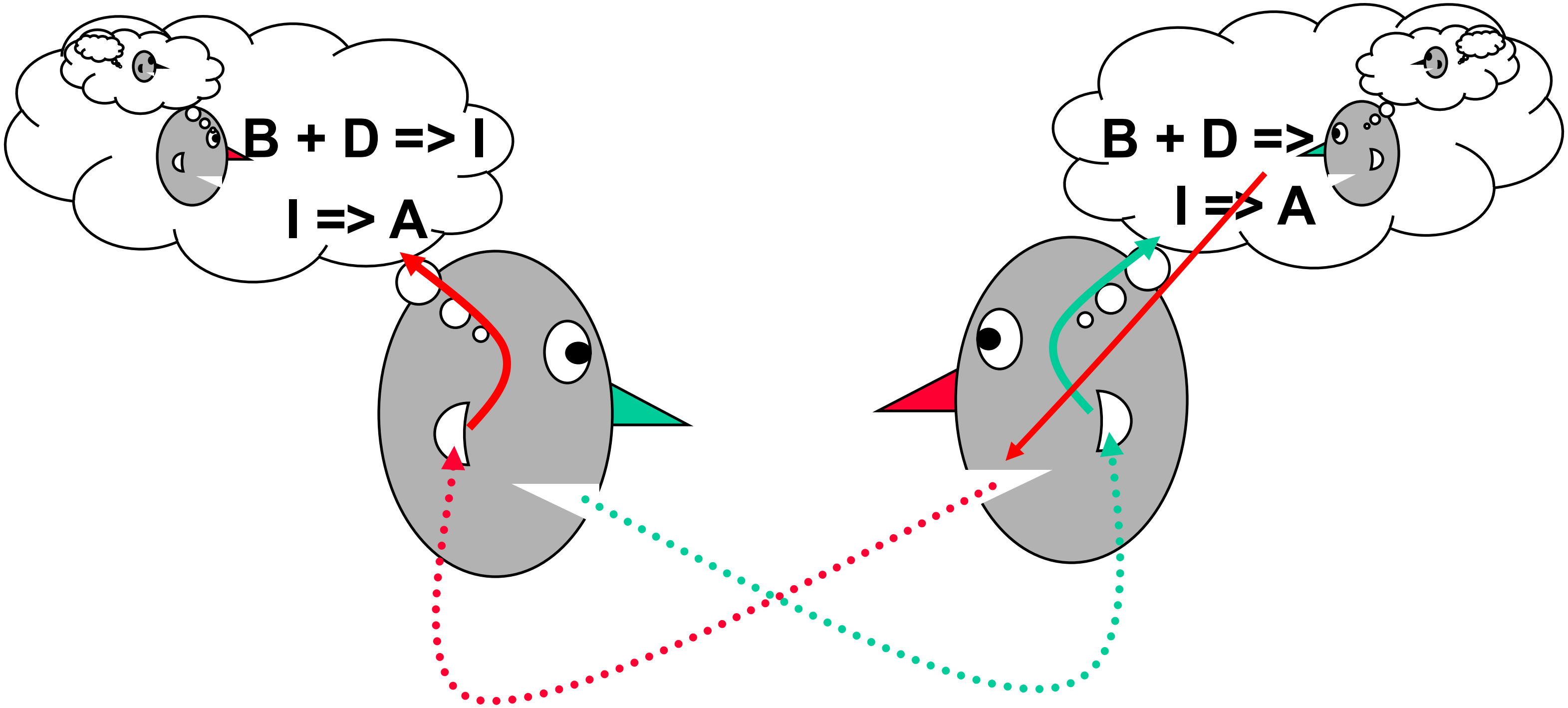
The intentional stance

- Agents have “propositional attitudes”
- Propositional attitudes are three-part relationship between
  - an agent,
  - a content-bearing proposition (e.g., “*it is raining*”), and
  - a finite set of propositional attitudes an agent might have with respect to the proposition (e.g., believing, asserting, fearing, wondering, hoping, etc.)
- <*a*, fear, raining(t_{now}) >

BDI Agents, Theories and Architectures

- BDI architectures describe the internal state of an agent by the mental states of beliefs, goals and intentions
- BDI theories provide a conceptual model of the knowledge, goals, and commitments of an agent
- BDI agents have some (implicit or explicit) representations of the corresponding attitudes

BDI Model and Communication



- Communication is a means to (1) reveal to others what our BDI state is and (2) attempt to effect the BDI state of others.
- Note the recursion: an agent has beliefs about the world, beliefs about other agents, beliefs about the beliefs of other agents, beliefs about the beliefs another agent has about it, ...

Speech Act Theory

High level framework to account for human communication

Language as Action (Austin)

- Speakers do not just utter true or false sentences
- Speakers perform speech acts: requests, suggestions, promises, threats, etc.
- Every utterance is a speech act

Conceptual and Theoretical Foundations II

Knowledge Representation and Ontologies

Representation and Reasoning

- Intelligent agents need to be able to represent and reason about many things, including:
  - models of other (human or artificial) agent's beliefs, desires, intentions, perceptions, plans, preferences, objective properties, etc.
  - task, task structures, plans, etc.
  - meta-data about documents and collections of documents
- In general, they will need to **communicate** the same range of knowledge.
- A variety of content languages have been used with ACLs, including KIF, SL, Loom, Prolog, SQL, RDF…
- There is a special interest in content languages that can serve as a neutral, but expressive, **interlingua** for a wide range of systems.
- We'll look at KIF in a bit more detail.

KR Language Components

- **A logical formalism**
  - Syntax for well formed formulae (wffs)
  - Vocabulary of logical symbols (e.g., and, or, not, =>, ...)
  - Interpretation semantics for the logical symbols, e.g.,

    (=> A B)" is true if and only if B is true or A is false.
- **An ontology**
  - **Vocabulary** of non-logical symbols (relations, functions, constants)
  - **Definitions** of non-primitive symbols, e.g.

    (<=> (Bachelor ?x) (AND (Man ?x) (Unmarried ?x)))
  - **Axioms** restricting the interpretations of primitive symbols, e.g.

    (=> (Person ?x) (Gender (Mother ?x) Female))
- **A proof theory**
  - Specification of the reasoning steps that are logically sound, e.g.

    From "(=> S1 S2)" and "S1", conclude "S2"

O-O Languages Too Restrictive

- Frames, object schema, description logics are popular KR languages used for ontologies
- They support definitional axioms of the form:
  - R(x) ⇒ … Λ P(x) Λ … {subclass}
  - R(x) ⇒ … Λ [S(x,y) ⇒ P(y)] Λ … {value class}
  - R(x) ⇒□… Λ E □y S(x,y) Λ … {value cardinality}

  …
- They don’t support –
  - N-ary relations and functions
  - Standard properties of relations and functions

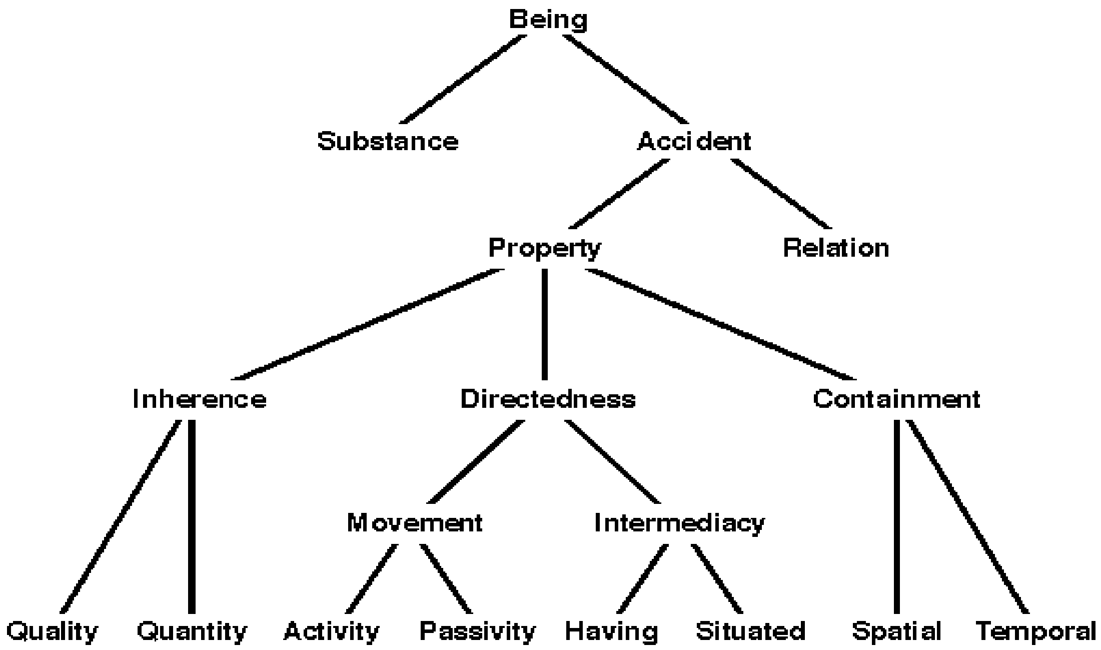
    e.g., transitive, symmetric, inverse
  - Partial sufficient conditions e.g., x>0 ⇒ R(x)
  - Disjunction of classes
  - General negation
  - …

Other alternatives

- OKBC (see ontologies)
- Java objects (see AgentBuilder)
- SL (see FIPA)
- Common Logic (son of KIF)
- Constraints
- Database tuples
- RDF, DAML, DAML+OIL, OWL
- *..your favorite representation language here..*

What is an ontology

- The subject of ontology is the study of the categories of things that exist or may exist in some domain.
- The word ontology is from the Greek *ontos* for being and *logos* for word.



- Aristotle offered an ontology which included 10 categories, shown as the leaves in this tree (from Sowa, after Brentano)

Common Semantics
Shared Ontologies and Ontolingua

Ontology : A common vocabulary and agreed upon meanings to describe a subject domain.

On*tol"o*gy (?), n. [Gr. the things which exist (pl.neut. of , , being, p.pr. of to be) + -logy: cf.F. ontologie.]

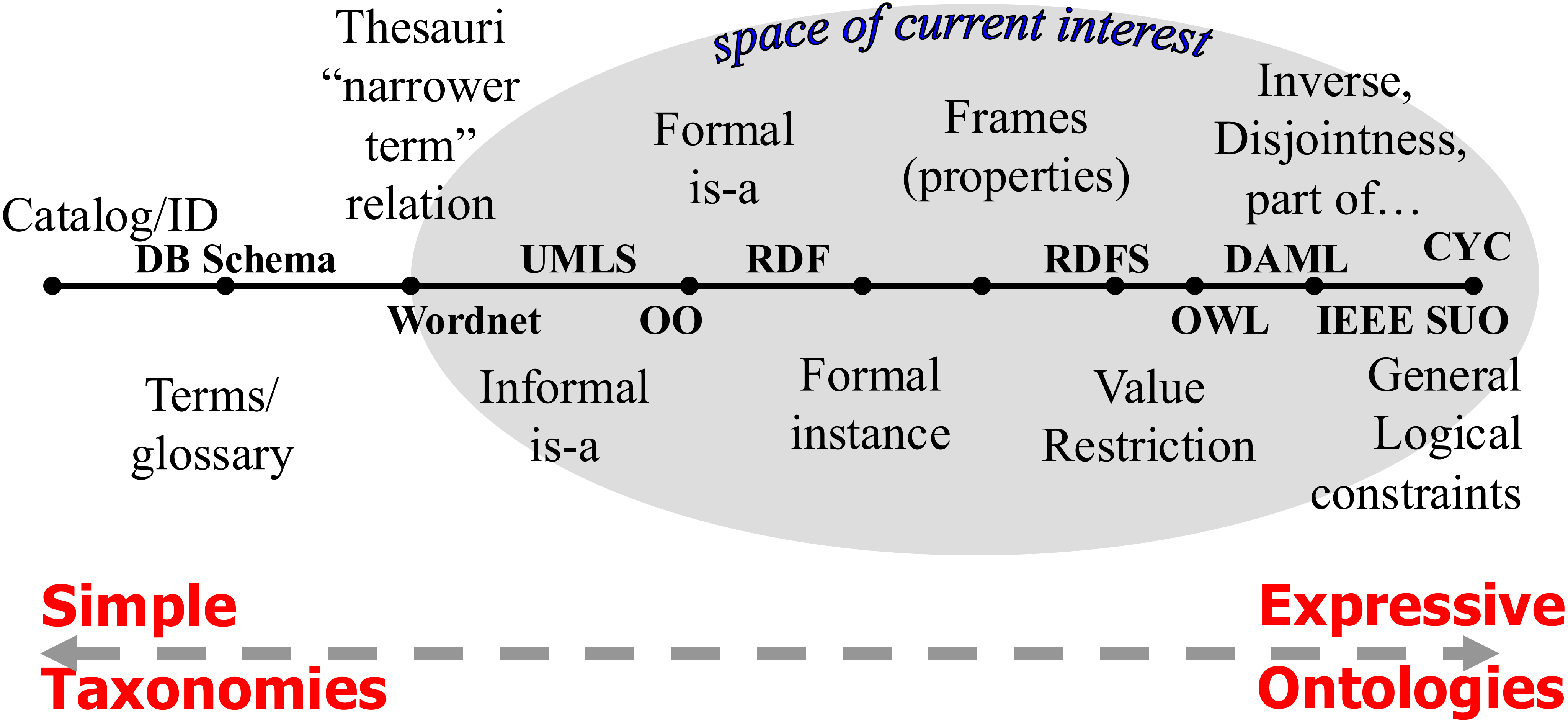
That department of the science of metaphysics which investigates and explains the nature and essential properties and relations of all beings, as such, or the principles and causes of being.

Webster's Revised Unabridged Dictionary (G & C. Merriam Co., 1913, edited by Noah Porter)

This is not a profoundly new idea …

- Vocabulary specification
- Domain theory
- Conceptual schema (for a data base)
- Class-subclass taxonomy
- Object schema

Ontology languages vary in expressivity



space of current interest

Catalog/ID

Terms/
glossary

Thesauri
"narrower
term"
relation

Informal
is-a

Formal
is-a

Formal
instance

Frames
(properties)

Value
Restriction

Inverse,
Disjointness,
part of…

General
Logical
constraints

DB Schema

Wordnet

UMLS

OO

RDF

RDFS

OWL

DAML

IEEE SUO

CYC

Simple
Taxonomies

Expressive
Ontologies

After Deborah L. McGuinness (Stanford)

Conceptual Schemas

A conceptual schema specifies the intended meaning of concepts used in a data base

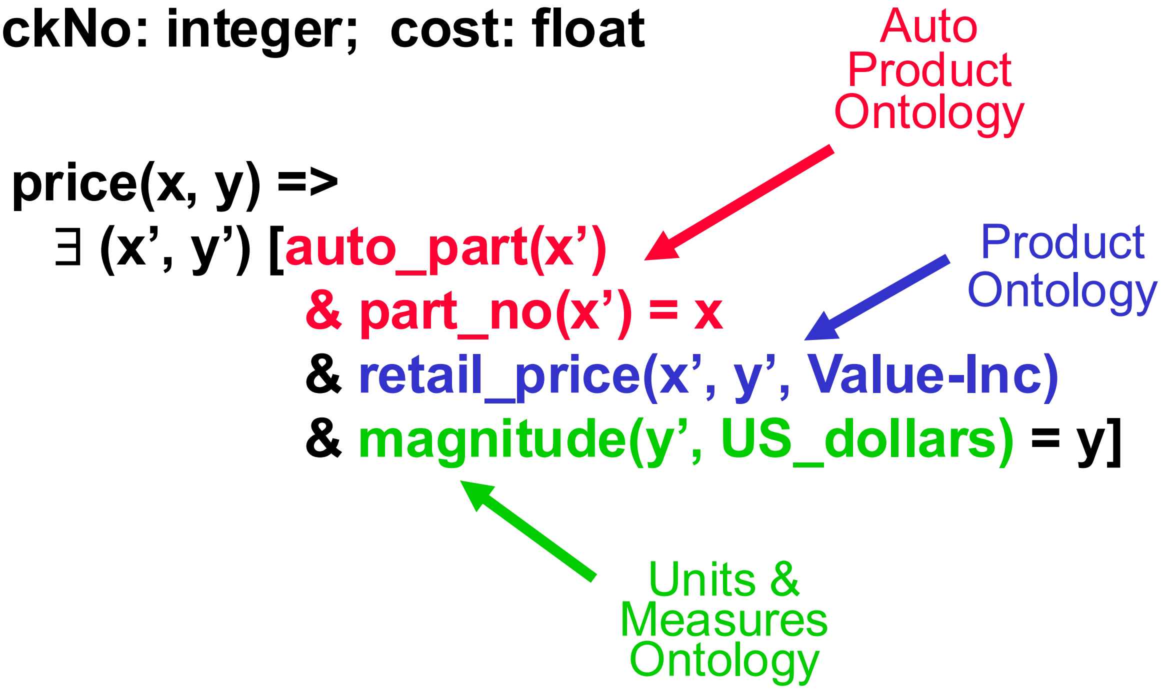
Data Base:

139	74.50
140	77.60
…	…

Data Base Schema:

Table: price
***stockNo: integer; cost: float**

Conceptual Schema:



price(x, y) =>
∃ (x’, y’) [auto_part(x’)
& part_no(x’) = x
& retail_price(x’, y’, Value-Inc)
& magnitude(y’, US_dollars) = y]

Auto Product Ontology

Product Ontology

Units & Measures Ontology

Implicit vs. Explicit Ontologies

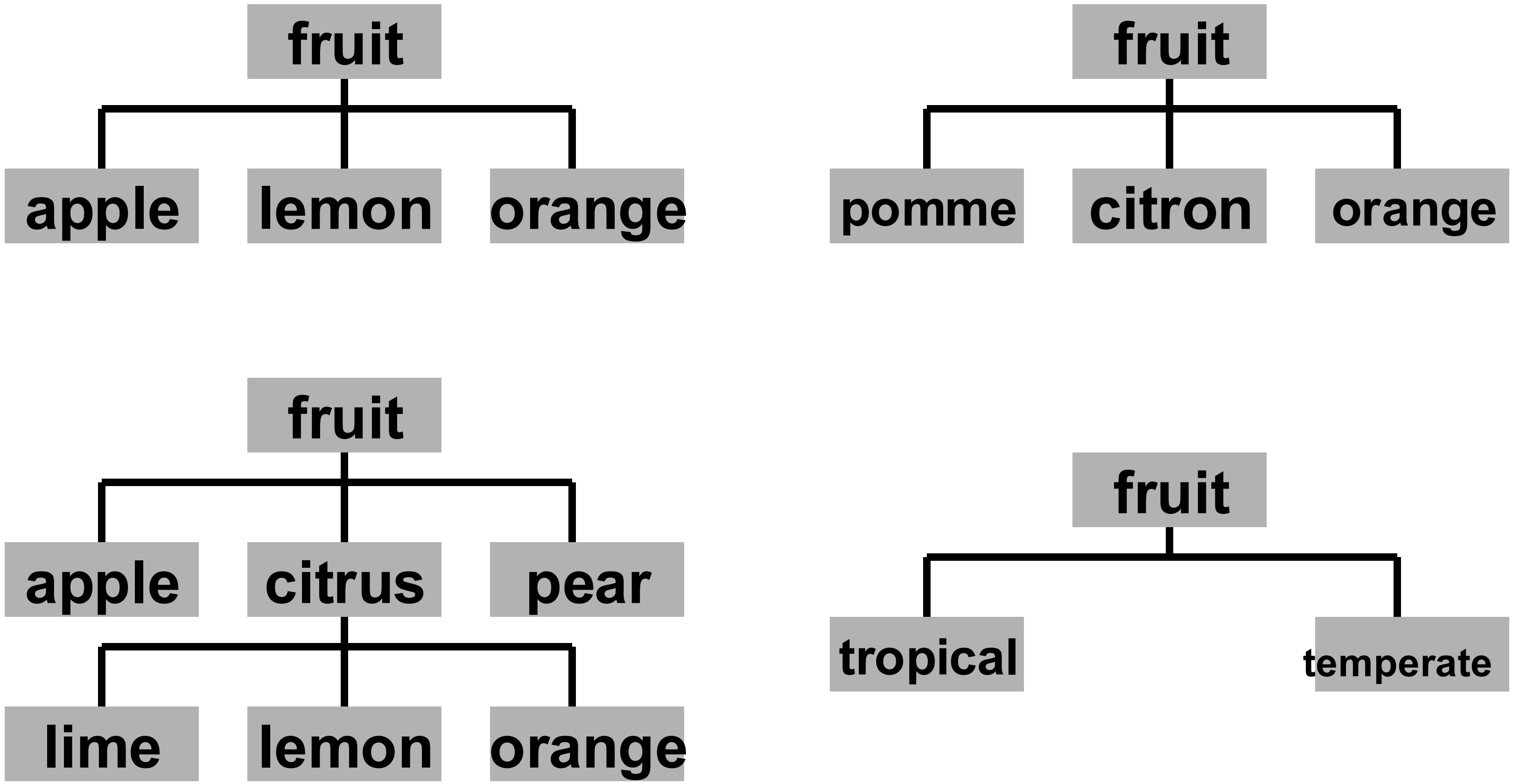
- Systems which communicate and work together must share an ontology.
- The shared ontology can be **implicit** or **explicit**.
- Implicit ontology are typically represented only by procedures and data structures.
- Explicit ontologies are (ideally) given a declarative representation in a well defined knowledge representation language.

⇒ *Explicit ontologies enable machine understanding*

Conceptualizations, Vocabularies and Axiomitization

- Three important aspects to explicit ontologies
  - **Conceptualization** involves the underlying model of the domain in terms of objects, attributes and relations.
  - **Vocabulary** involves assigning symbols or terms to refer to those objects, attributes and relations.
  - **Axiomitization** involves encoding rules and constraints which capture significant aspects of the domain model.
- Two ontologies may
  - be based on different conceptualizations
  - be based on the same conceptualization but use different vocabularies
  - differ in how much they attempt to axiomitize the ontologies

Simple examples



fruit

apple lemon orange

fruit

pomme citron orange

fruit

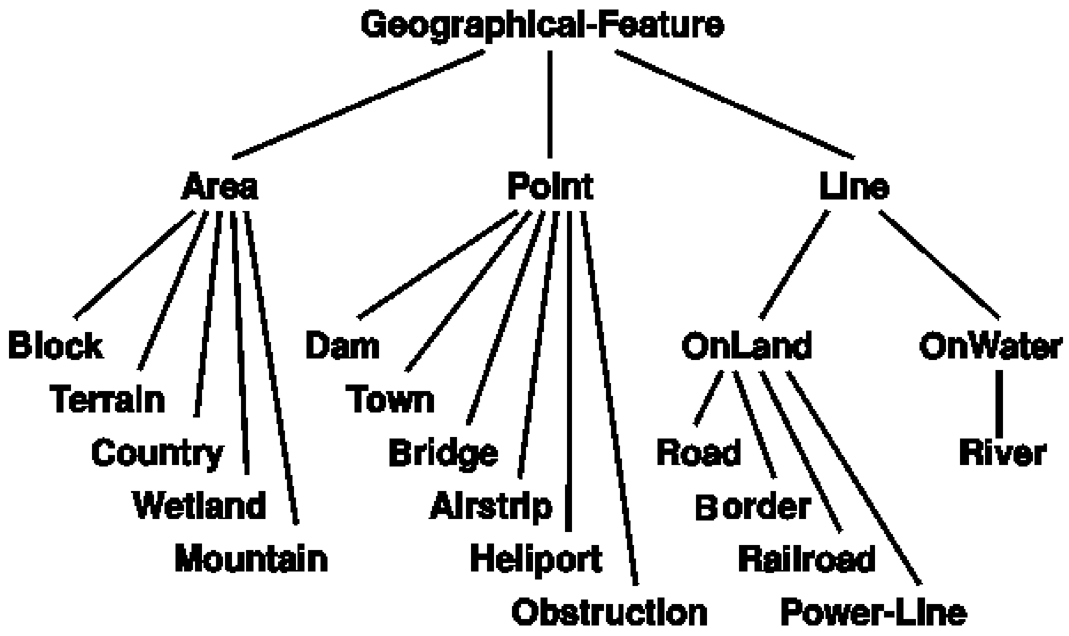
apple citrus pear

lime lemon orange

fruit

tropical temperate

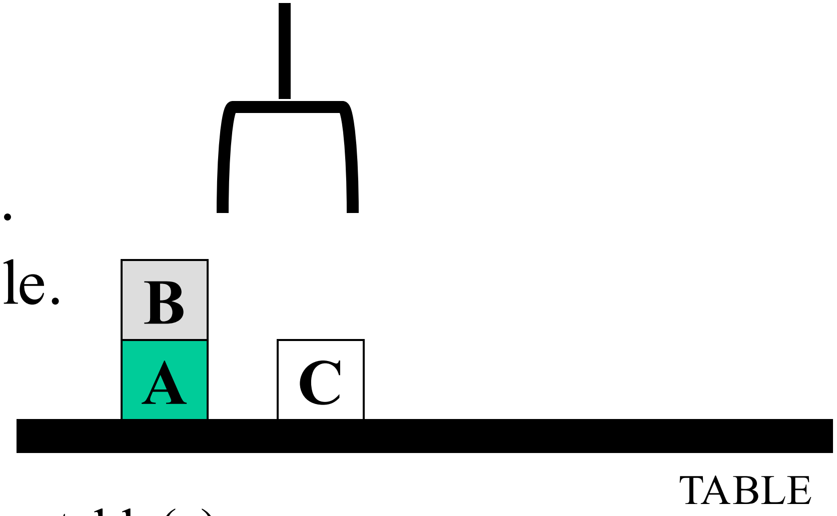
Top down vs. bottom up



- Philosophers build from the top down and are interested in capturing the most general concepts.
- Programmers tend to work from the bottom up, supporting a set of applications, with a little generality to help reuse and future development.
- Ex: CHAT-80 system (Periera and Warren, 1982) which answered NL questions about a geographic database.
  - Example of a **microworld** ontology supported NLP, query answering, and generation

Blocks world

- The **blocks world** is another microworld used often for NLP, vision, planning.
- It consists of a **table**, a set of **blocks** or different *shapes*, *sizes* and co*lors* and a robot **hand**.
- Some typical domain constraints:
  - Only one block can be on another block.
  - Any number of blocks can be on the table.
  - The hand can only hold one block.



- Typical representation:

block(a) color(a green) size(a,small) ontable(a) ontable(c)

block(b) color(b,gray) size(b,small) on(b,a) handempty

block(c) color(c,white) size(c,small) clear(b) clear(c)

$\forall$B1,B2,B3 on(B1,B2) $\wedge$ on(B3,B2) $\wedge$ B2$\neq$table $\Rightarrow$ B1=B3

…

Big Ontologies

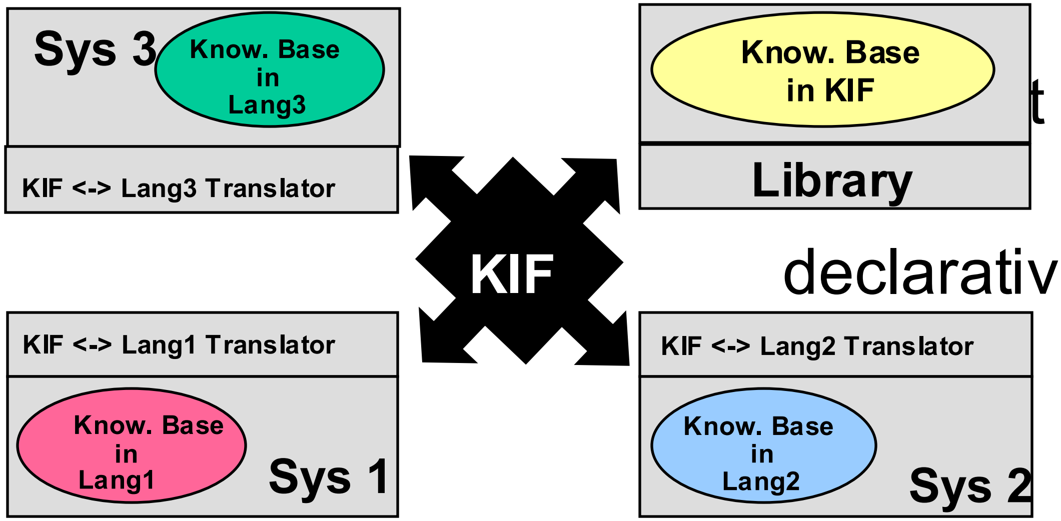
- There are several large, general ontologies that are freely available.
- Some examples are:
  - **Cyc -** O(10^4) terms and O(10^6) sentences
  - **OpenCyc –** an open source version of Cyc
  - **WordNet** - a large, on-line lexical reference system (~50K words)
  - **IEEE SUMO** – Suggested Upper Merged Ontology
  - **World Fact Book --** 5Meg of KIF sentences!
  - **UMLS** - NLM's Unified Medical Language System
- See http://www.cs.utexas.edu/users/mfkb/related.html for more

Ontology Conclusions

- Shared ontologies are essential for agent communication and knowledge sharing
- Ontology tools and standards are important
  - Ontolingua, OKBC and Protege are good examples
  - XML, RDF, DAML+OIL may be a next step
- Some large general ontologies are available
  - Cyc, SUMO, World Fact Book, WordNet, …
- For more information…
  - **http://www.cyc.com/** -- for CYC and OpenCYC
  - **http://suo.ieee.org/** -- for Standard Upper Ontology
  - **http://www.daml.org** -- for DAML and web ontologies
  - **http://www.w3.org/2001/sw/** -- for OWL and web ontologies
  - http://www.cogsci.princeton.edu/~wn/ -- for wordnet
  - **http://www.jfsowa.com/ontology/** -- John Sowa's ontology site

3 Origins - Knowledge Sharing Effort (KSE)

Knowledge Interchange Format



- KIF ~ First order logic t theory
- An **interlingua** for encoded declarative knowledge
  - Takes translation among n from $O(n^2)$ to O(n)
- Common language for reusable knowledge
  - Implementation independent semantics
  - Highly expressive - can represent knowledge in typical application KBs.
  - Translatable - into and out of typical application languages
  - Human readable - good for publishing reference models and ontologies.
- Current specification at http://logic.stanford.edu/kif/ now being superceded by an effort to specify and standardize “Common Logic” (http://cl.tamu.edu/)

KIF Syntax and Semantics

- Extended version of first order predicate logic
- Simple list-based linear ASCII syntax, e.g.,

```
(forall ?x (=> (P ?x) (Q ?x)))
(exisits ?person (mother mary ?person))
(=> (apple ?x) (red ?x))
(<<= (father ?x ?y) (and (child ?x ?y) (male ?x))
```

- Model-theoretic semantics
- KIF includes an axiomatic specification of large function and relation vocabulary and a vocabulary for numbers, sets, and lists

KIF Software

- Several KIF-based reasoners in LISP are available from Stanford (e.g., EPILOG).
- IBM's ABE (Agent Building Environment) & RAISE reasoning engine use KIF as their external language.
- Stanford's Ontolingua uses KIF as its internal language and Protégé can export KIF sentences.
- Translators (partial) exist for a number of other KR languages, including LOOM, Classic, CLIPS, Prolog,...
- Parsers for KIF exist which take KIF strings into C++ or Java objects.
- KIF's standarization effort stalled in the mid 90's but has been revived as **Common Logic**

Common Logic

- See http://cl.tamu.edu/
- An effort to define and standardize
  - A very expressive interchange language between heterogeneous systems;
  - An ASCII and human readable "plain vanilla" notation for logic;
  - A kind of general-purpose logic kit, easily configurable to a variety of machine- and human- oriented uses while preserving semantic clarity.
- Driven (mostly) by the IEEE Standard Upper Ontology group
- To be potentially used:
  - for translation between other kinds of notations for people with different needs or preferences.
  - in expressing the SUO;
  - for defining the meaning (semantics?) of languages;
  - as input and output to various inference engines;
  - in the emerging internet ontology community;
  - as an exercise in writing a first-order model theory.

KQML

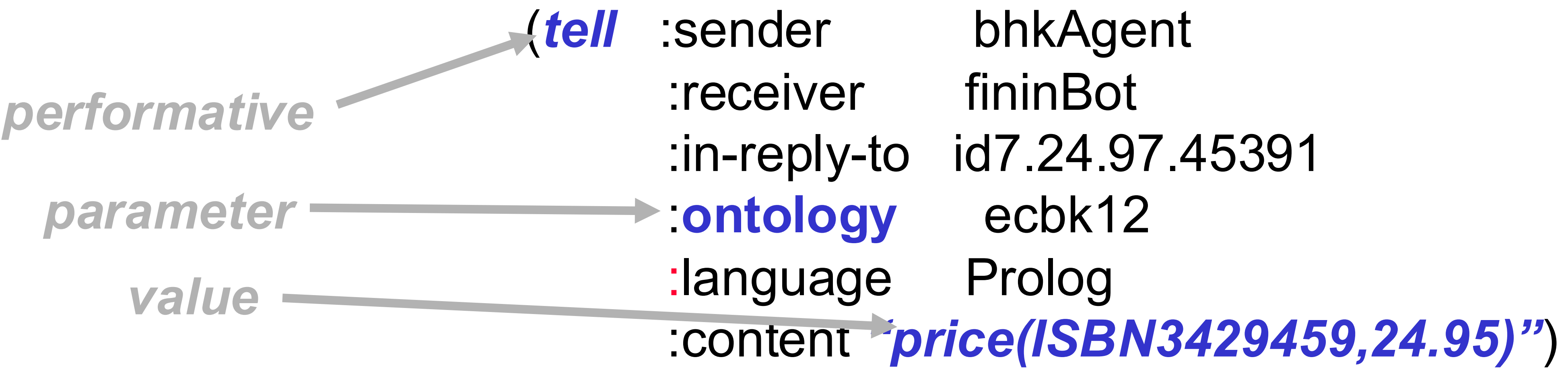
Knowledge Query and Manipulation Language

- KQML is a high-level, message-oriented, communication language and protocol for information exchange independent of content syntax and ontology.
- KQML is independent of
  - the transport mechanism (e.g., tcp/ip, email, corba objects, IIOP, etc.)
  - Independent of content language (e.g., KIF, SQL, STEP, Prolog, etc.)
  - Independent of the ontology assumed by the content.
- KQML includes primitive message types of particular interest to building interesting agent architectures (e.g., for mediators, sharing intentions, etc.)

KQML Specifications

- There are two KQML specification documents:
  - *Specification of the KQML Agent-Communication Language plus example agent policies and architectures*, The DARPA Knowledge Sharing Initiative, External Interfaces Working Group, 1993. http://www.cs.umbc.edu/papers/kqml93.pdf
  - *A Proposal for a new KQML Specification*, Yannis Labrou and Tim Finin, TR CS-97-03, Feb.1997, Computer Science and Electrical Engineering Department, University of Maryland Baltimore County, Baltimore, MD 21250. http://www.cs.umbc.edu/kqml/papers/kqml97.pdf
- There are also many dialects and “extended” versions of KQML plus lots of important concepts not addressed in either specification document (e.g., security).
- We’ll mostly focus on the 1997 document plus other ideas used in practice.

A KQML Message



```
(tell   :sender        bhkAgent
        :receiver      fininBot
        :in-reply-to   id7.24.97.45391
        :ontology      ecbk12
        :language      Prolog
        :content       “price(ISBN3429459,24.95)”)
```

Represents a single *speech act* or *performative*

ask, tell, reply, subscribe, achieve, monitor, ...

with an associated *semantics and protocol*

$tell(\ i,j,\ B_i\phi\ ) = fp[B_i\ B_i\phi \land \neg\ B_i(\ Bif_j\ B_i\phi \lor Uif_j\ B_i\phi\ )] \land re[B_j\ B_i\phi]$ ...

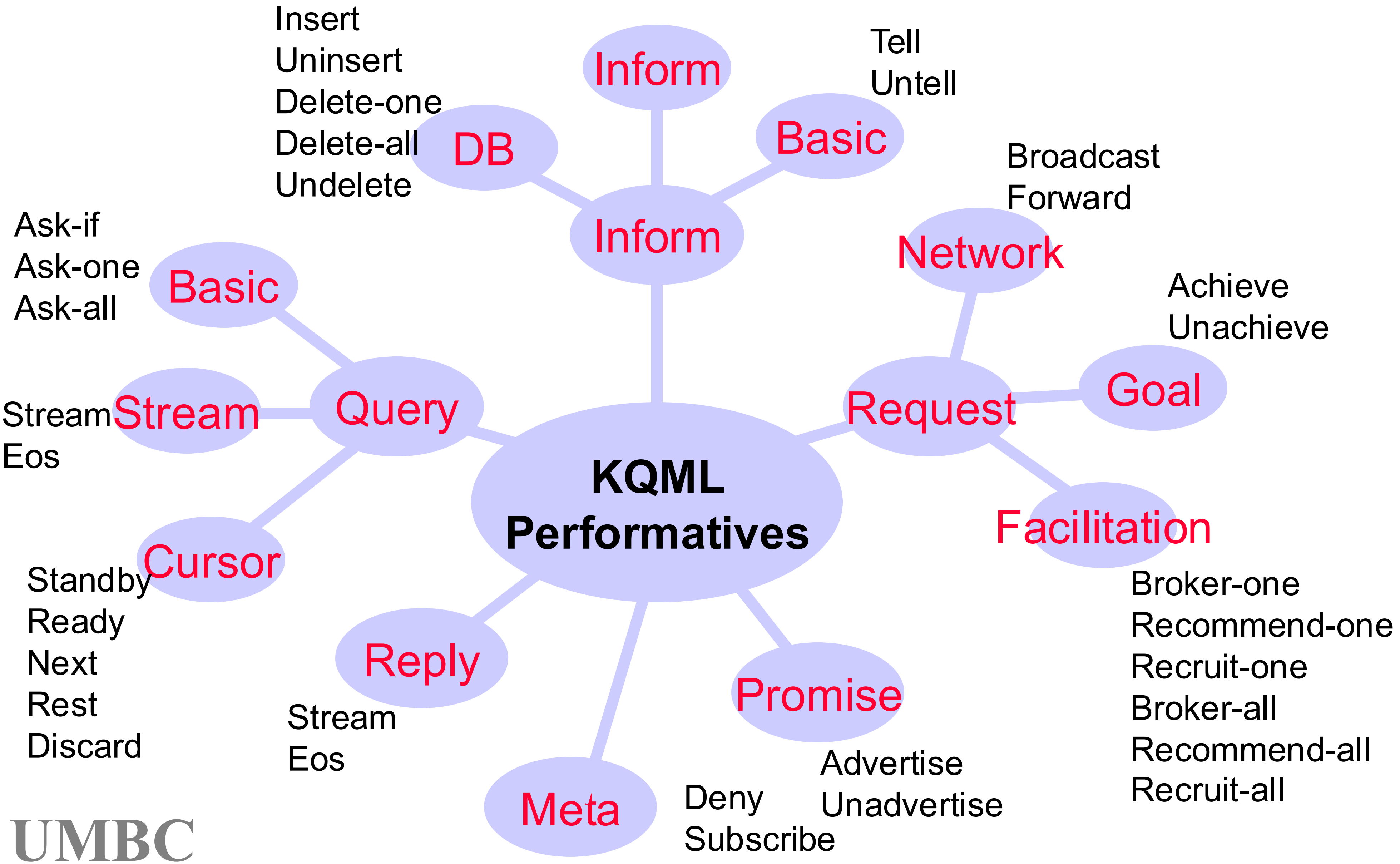
and a list of *attribute/value pairs*

:content, :language, :from, :in-reply-to

KQML Syntax

- KQML was originally defined as a language with a particular linear syntax which is based on Lisp.
- Alternate syntaxes have been used, e.g., based on SMTP, MIME, HTTP, etc.)
  - There are proposals for a meta-syntax that can support different syntactic dialects.
- KQML has also been mapped onto objects and passed from agent to agent as objects (e.g., if in the same memory space) or serialized objects.
- **KQML is not about syntax.**

Performatives (1997)



KQML Performatives

Inform

- Inform
- DB: Insert, Uninsert, Delete-one, Delete-all, Undelete
- Basic: Tell, Untell

Query

- Basic: Ask-if, Ask-one, Ask-all
- Stream: Stream, Eos
- Cursor: Standby, Ready, Next, Rest, Discard

Request

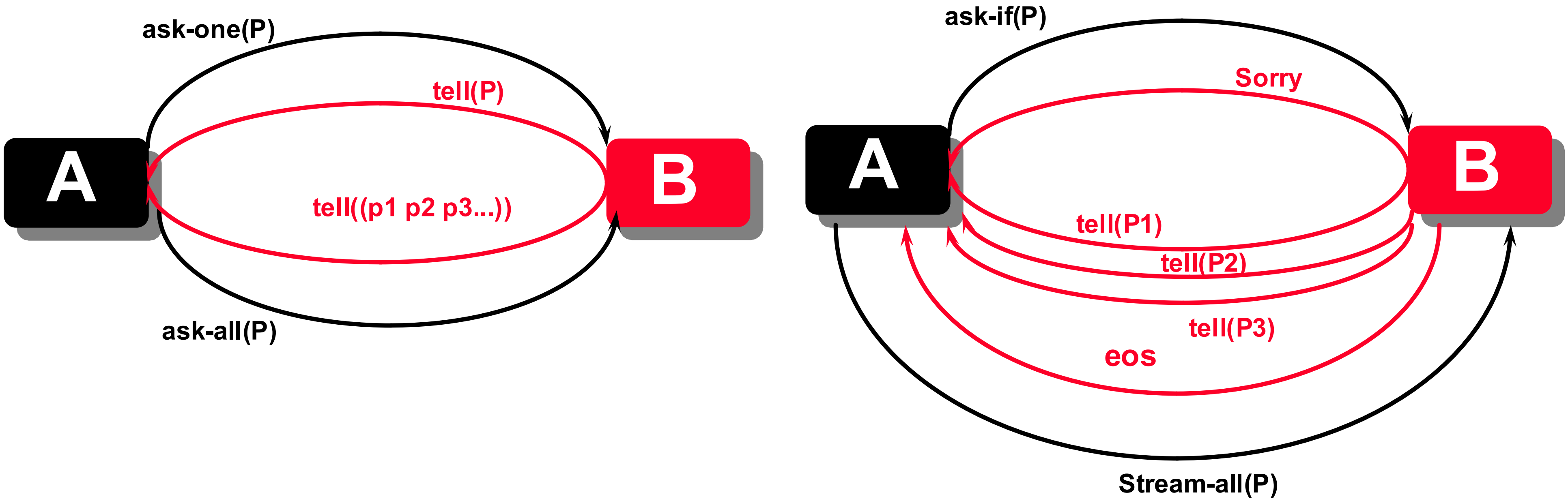
- Network: Broadcast, Forward
- Goal: Achieve, Unachieve
- Facilitation: Broker-one, Recommend-one, Recruit-one, Broker-all, Recommend-all, Recruit-all

Reply: Stream, Eos

Meta

Promise: Advertise, Unadvertise, Deny, Subscribe

Simple Query Performatives



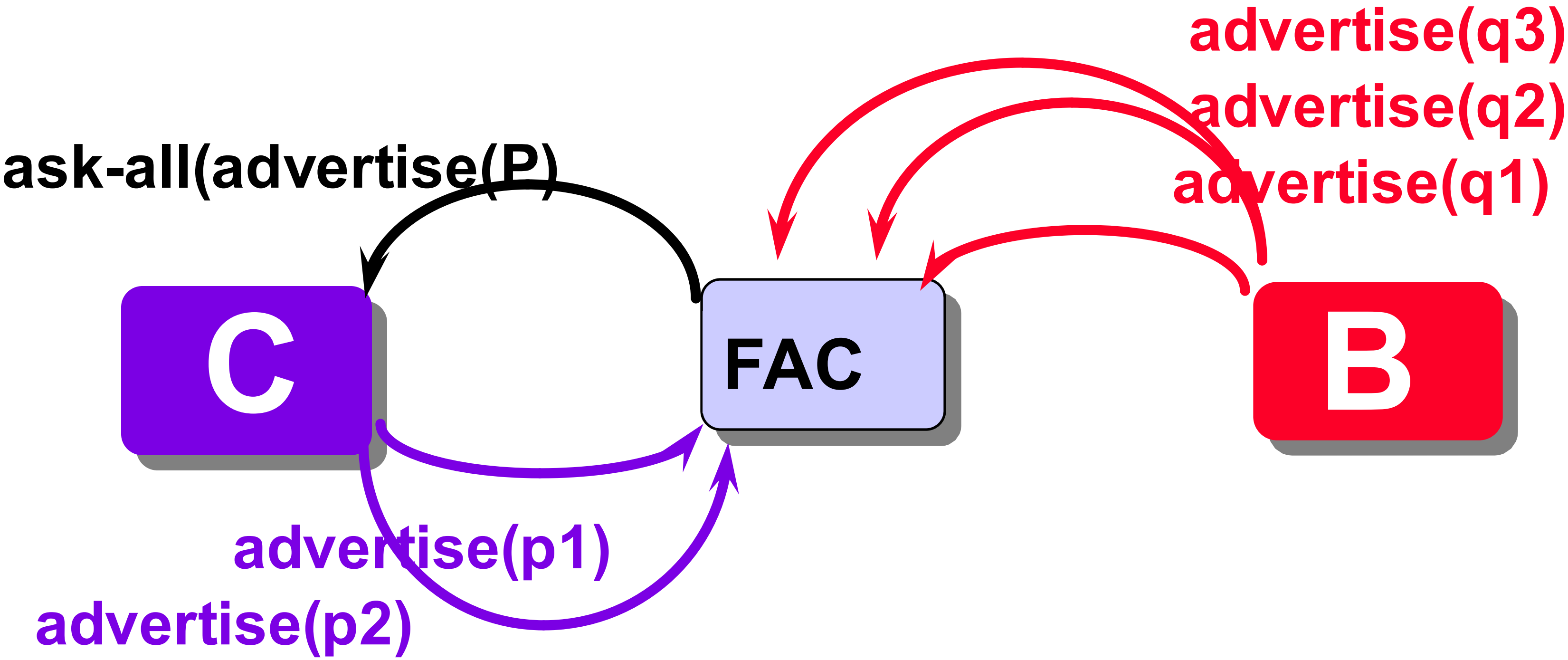
- The **ask-one**, **ask-all**, **ask-if**, and **stream-all** performatives provide a basic query mechanism.

Facilitation Services

Facilitators are a class of agents who

- traffic in meta-knowledge about other agents.
- provide communication services such as:
  - message forwarding and broadcasting
  - resource discovery
  - matchmaking
  - content-based routing
  - meta-knowledge queries
- Performatives of special interest to facilitators are
  - advertise, broker, recruit, recommend, forward, broadcast, etc.
- Brokers are generally considered to focus on matchmaking
- Facilitators can be intelligent or not
  - Intelligent facilitators use domain knowledge in matching services needs and offers.

Capability Description

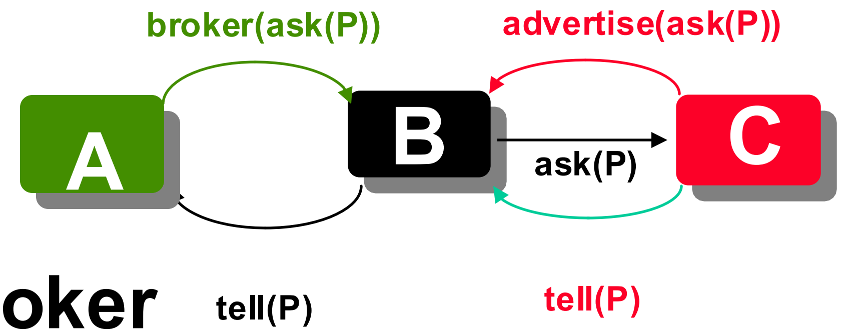


The **advertise** performative is used to describe the performatives an agent is prepared to accept.

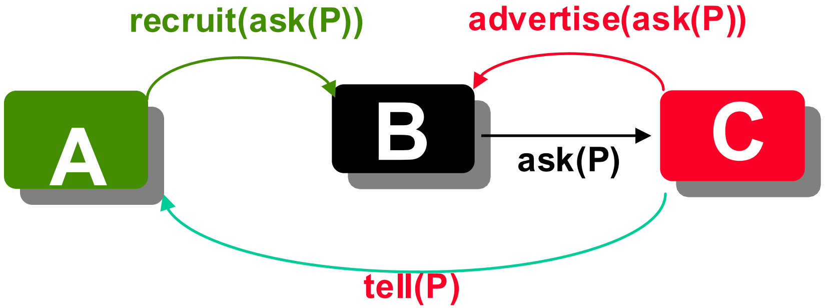
Facilitation Performatives

The three facilitation performatives come in a X-one and X-all versions:

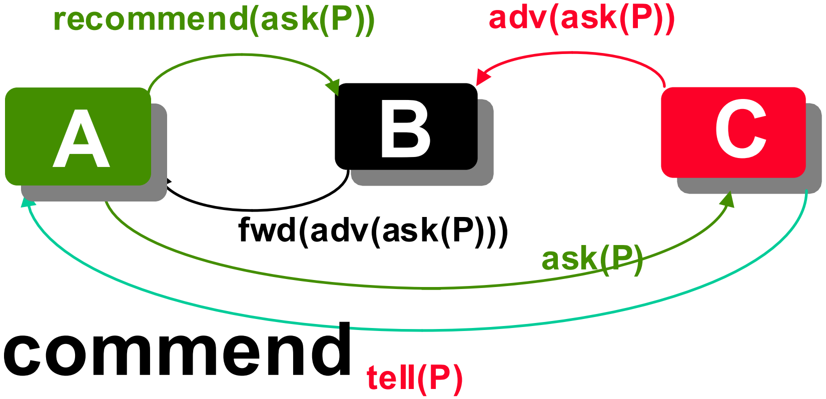
- Broker-one and broker-all
- Recruit-one and recruit-all
- recommend-one and recommend-all



Broker



Recruit



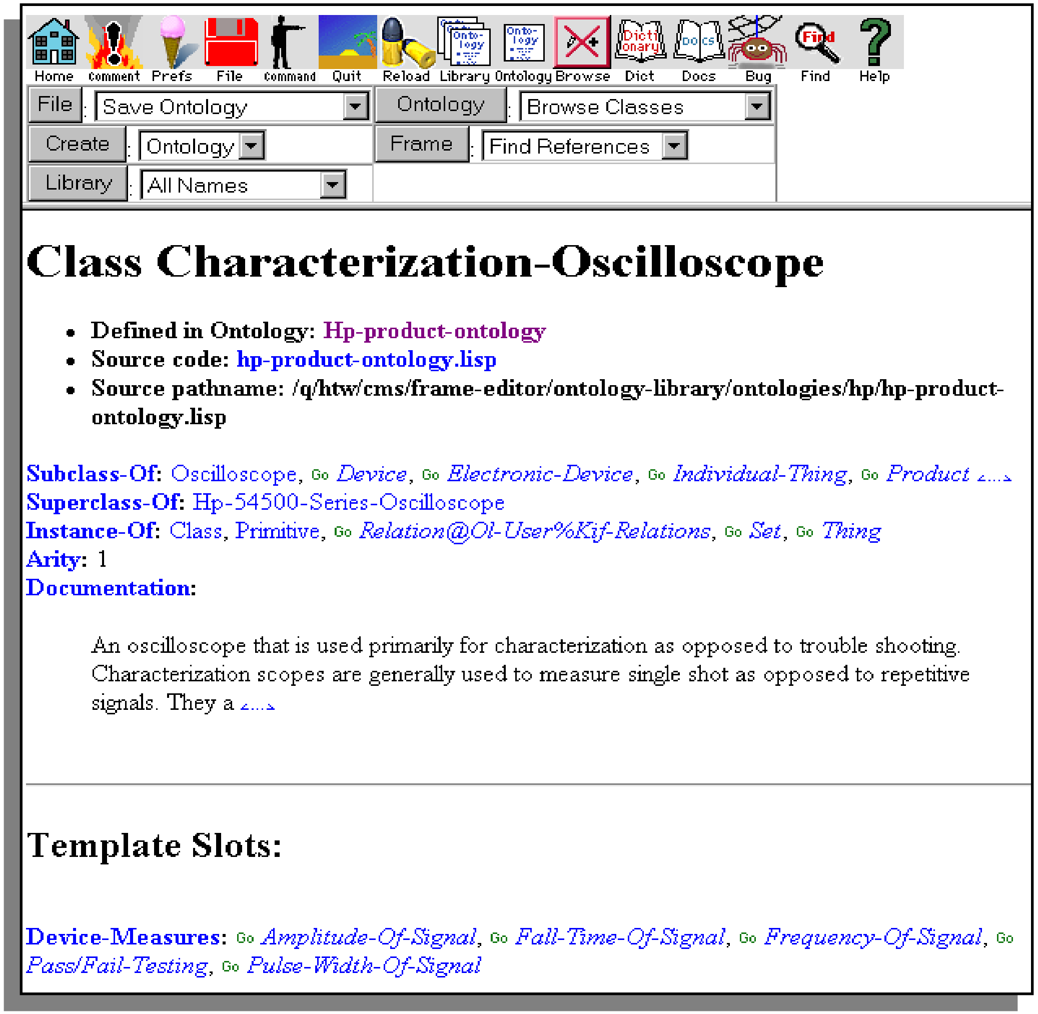
Recommend

KQML APIs and System Interfaces

- There have been dozens of APIs written for KQML
- Written in and for different languages
  - Lisp, Scheme, Prolog, C/C++, Java, CLIPS, Smalltalk, Tcl, Perl, ...
- And interfacing to may different systems
  - Loom, Cyc, SIMS (Information Integration), SIPE (Planning), Various Databases, …
- More recent is the appearance of KQML-speaking "*agent shells*", offering more than just an API.
- More on these later

Ontolingua - Language

- Ontolingua allows full KIF
  - 1st order logic with relation constants in domain of discourse
  - Extremely expressive
  - Too much for most users
  - Too much for most systems!
- Ontolingua provides an object-oriented projection
- Statements within the o-o sublanguage easy to make
  - But any statement is allowed
- Ontolingua separates representation from presentation



Class Characterization-Oscilloscope

- **Defined in Ontology: Hp-product-ontology**
- **Source code: hp-product-ontology.lisp**
- **Source pathname: /q/htw/cms/frame-editor/ontology-library/ontologies/hp/hp-product-ontology.lisp**

Subclass-Of: Oscilloscope, Go *Device*, Go *Electronic-Device*, Go *Individual-Thing*, Go *Product* ...
Superclass-Of: Hp-54500-Series-Oscilloscope
Instance-Of: Class, Primitive, Go *Relation@Ol-User%Kif-Relations*, Go *Set*, Go *Thing*
Arity: 1
Documentation:

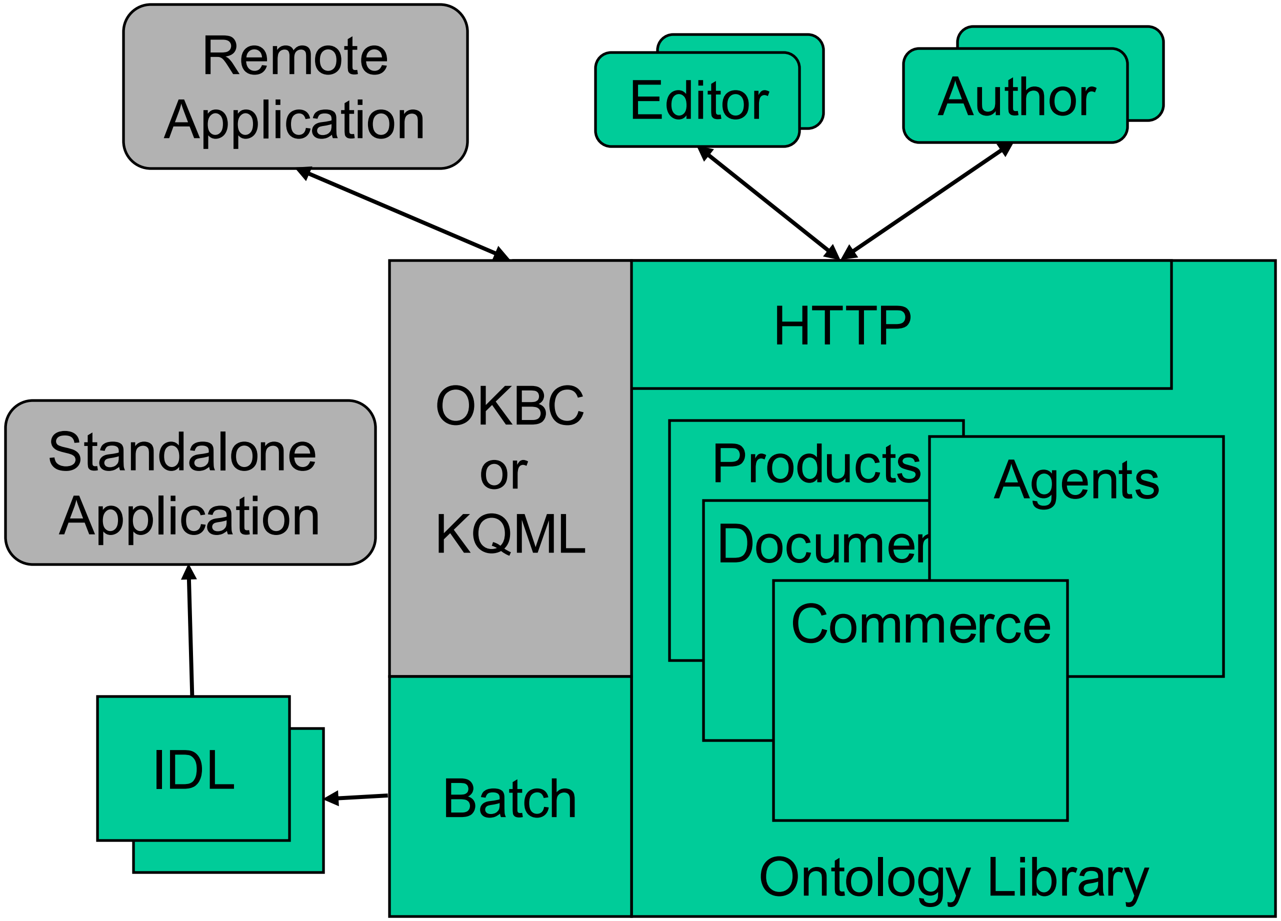
An oscilloscope that is used primarily for characterization as opposed to trouble shooting. Characterization scopes are generally used to measure single shot as opposed to repetitive signals. They a ...

Template Slots:

Device-Measures: Go *Amplitude-Of-Signal*, Go *Fall-Time-Of-Signal*, Go *Frequency-Of-Signal*, Go *Pass/Fail-Testing*, Go *Pulse-Width-Of-Signal*

Ontolingua - Architecture

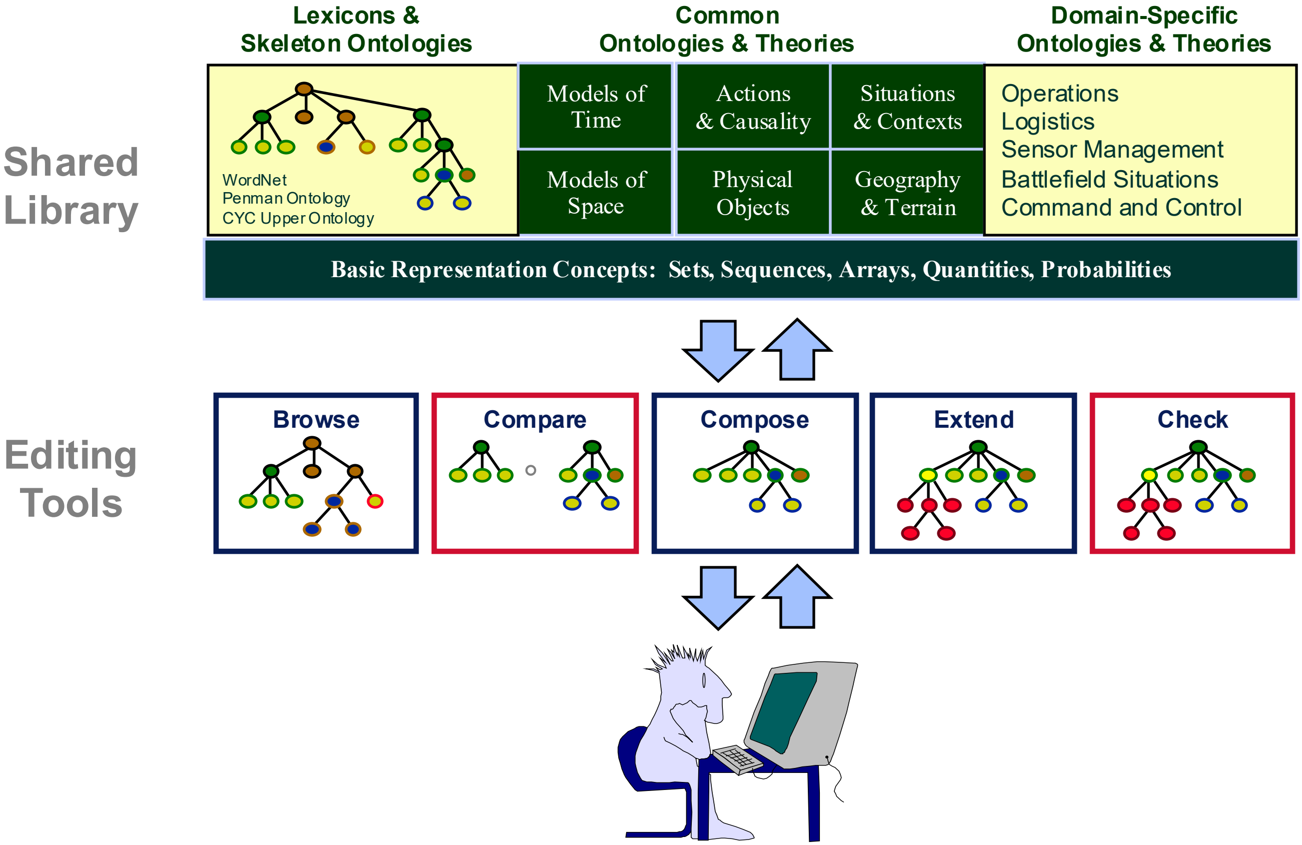
- Authors, editors, reviewers interaction via the web interface
- Applications interact via the OKBC or KQML interface
- Batch translation of ontologies supports the construction of stand-alone applications



Ontology Library and Editing Tools

Ontolingua is a language for building, publishing, and sharing ontologies.

- A web-based interface to a browser/editor server at *http://ontolingua.stanford.edu/* and mirror sites.
- Ontologies can be translated into a number of content languages, including KIF, LOOM, Prolog, CLIPS, etc.



UMBC
an Honors University in Maryland

4
Standardization
FIPA

What is FIPA

- The Foundation for Intelligent Physical Agents (FIPA) is a non-profit association.
- FIPA's purpose is to promote the success of emerging agent-based applications, services and equipment.
- FIPA's goal is pursued by making available in a timely manner, internationally agreed specifications that maximise interoperability across agent-based applications, services and equipment.
- http://fipa.org/

FIPA Standards

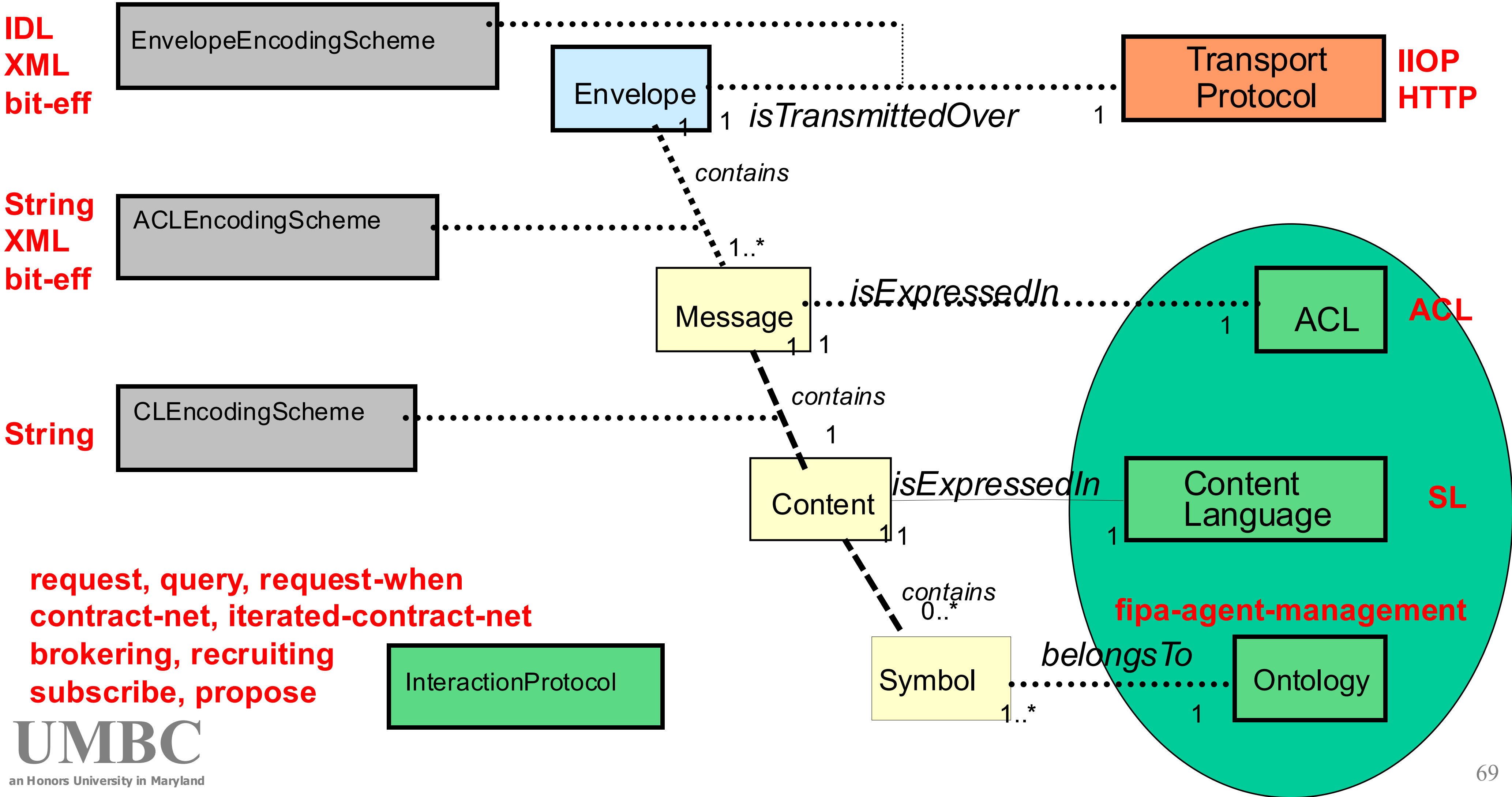
FIPA standards are organized into five areas

- Applications – application level standards, e.g., for mobile apps, network management
- Abstract architecture
- Agent communication – Communicative acts, content languages and interaction protocols
- Agent management
- Agent message transport – ACL representations, Envelope representations, Transport protocols

FIPA ACL specifications

- Library of communicative acts
- Library of conversation protocols
  - Expressed in AUML
- Library of content languages
  - SL (Semantic Language)
  - KIF (Knowledge Interchange Format)
  - RDF (Resource Description Framework)
  - CCL Constraint Choice Language
  - In theory, a FIPA ACL content language should support
    - Objects
    - Propositions
    - Actions
    - … or, suggest how such constructs are mapped to its own constructs

FIPA Standards Overview



IDL
XML
bit-eff

EnvelopeEncodingScheme

Envelope

1 1 *isTransmittedOver* 1

Transport Protocol

IIOP
HTTP

contains

String
XML
bit-eff

ACLEncodingScheme

1..*

Message

isExpressedIn

1 ACL

ACL

1 1

contains

String

CLEncodingScheme

1

Content

isExpressedIn

Content Language

SL

1 1 1

contains

0..*

fipa-agent-management

request, query, request-when
contract-net, iterated-contract-net
brokering, recruiting
subscribe, propose

InteractionProtocol

Symbol

belongsTo

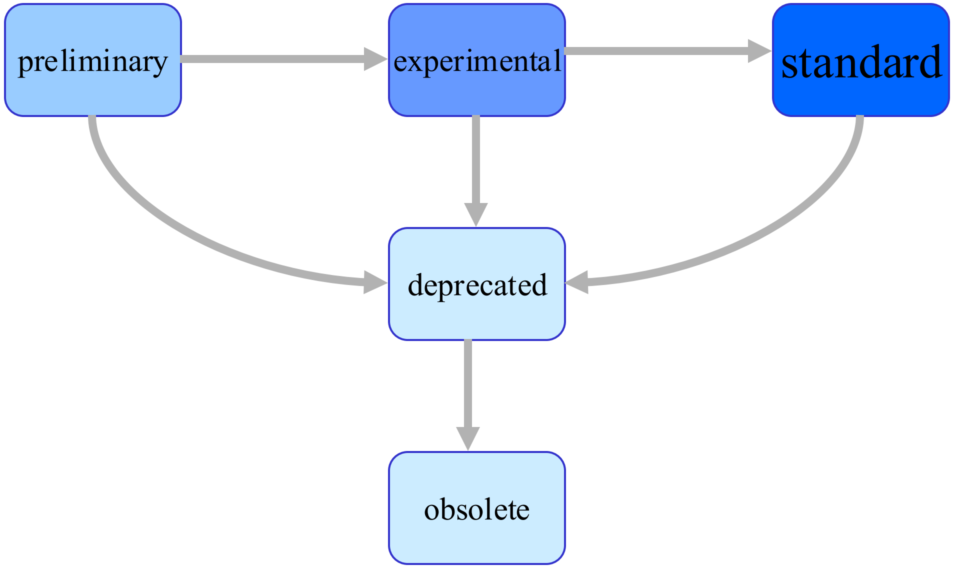
Ontology

1..* 1

UMBC
an Honors University in Maryland

FIPA specification lifecycle

- FIPA specifications are classified according to their position in the specification life cycle
- A specification progresses from its inception through to its ultimate resolution
- Stages
  - – Preliminary
  - – Experimental
  - – Standard
  - – Deprecated
  - – Obsolete



FIPA ACL

The FIPA Agent Communication Language

- Called FIPA ACL
- Based on speech acts
- Messages are actions (communicative actions or CAs)
- Communicative acts are described in both a narrative form and a formal semantics based on modal logic
- Syntax is similar to KQML
- Specification provides a *normative* description of high-level interaction protocols (aka conversations)

Communicative acts

- FIPA has a library of 22 CAs that includes the following
  - Accept-Proposal, Agree, Cancel, Call-for-Proposal, Confirm, Disconfirm, Failure, Inform, Inform-If, Inform-Ref, Not-Understood, Propagate, Propose, Proxy, Query-If, Query-Ref, Refuse, Reject-Proposal, Request, Request-When, Request-Whenever, Subscribe
- The semantics of the for primitive CAs (inform, request, confirm, disconfirm) are defined in SL
- The other CAs are defined using these.
- You can add new CAs by following this model.

Agent-Standardization - FIPA Cooperation between Agents

CAs for Information Exchange

- proposition or reference as content
- Basic CAs:
  - inform
  - query-ref
  - not-understood
- Advanced CAs:
  - inform-if, inform-ref
  - confirm, disconfirm
  - subscribe

Agent-Standardization - FIPA Cooperation between Agents

CAs for task delegation

- action-description as content
- Basic CAs:
  - request
  - agree
  - refuse
  - failure
  - not-understood
- Advanced CAs:
  - request-when, request-whenever
  - cancel

Agent-Standardization - FIPA
Cooperation between Agents

CAs for Negotiation

- action-description and proposition as content
- Initiating CA
  - cfp
- Negotiating CA
  - propose
- Closing CAs
  - accept-proposal
  - reject-proposal

Agent-Standardization - FIPA Cooperation between Agents

Example

```
(request
   :sender   (:name user_agent@bond.mchp.siemens.de:3410)
   :receiver (:name hilton_hotel@tcp://hilton.com:5001)
   :ontology fipa-pta
   :language SL
   :protocol fipa-request
   :content
   ( action hilton_hotel@tcp://hilton.com:5001
     ( book-hotel    (:arrival   04/07/1999) (:departure 12/07/1999)
      (:infos  (   ))
     )))
```

FIPA 99: other possibilities to define content!

Agent-Standardization - FIPA Cooperation between Agents

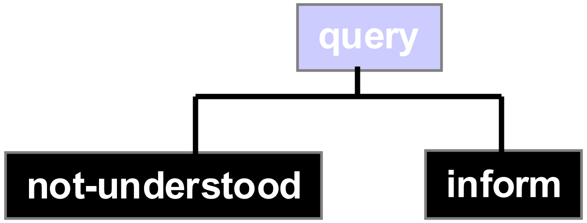
FIPA Cooperation

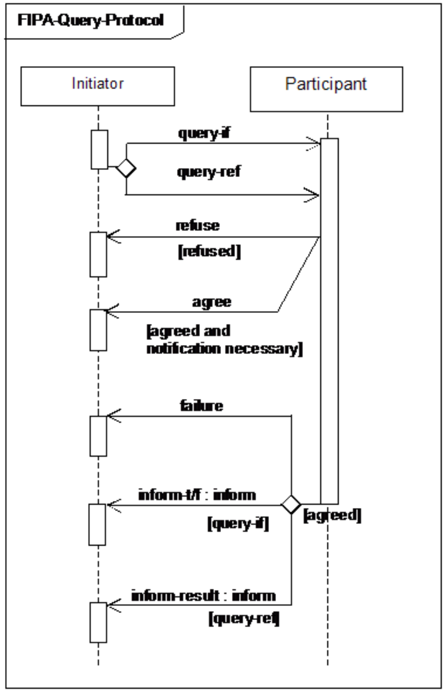
- CAs have their own formal semantics
  - difficult to implement
  - need not be implemented - agent must behave according to semantics
- Interaction protocols define structured conversations
  - based on CAs
  - basis for dialogues between agents
  - basic set of pre-defined IPs
  - own IPs can be defined

Major FIPA protocols

- **Basic protocols**
  - Request
  - Request-when
  - Query
- **Cooperation protocols**
  - Propose
  - Contract Net
  - Iterated contract net
- **Market mechanisms**
  - English auction
  - Dutch auction
- **Middle agent protocols**
  - Brokering
  - Recruiting
  - Subscribe

- Every ACL message is tagged with the protocol it assumes
- Protocols were originally specified using simple interaction diagrams and now AUML

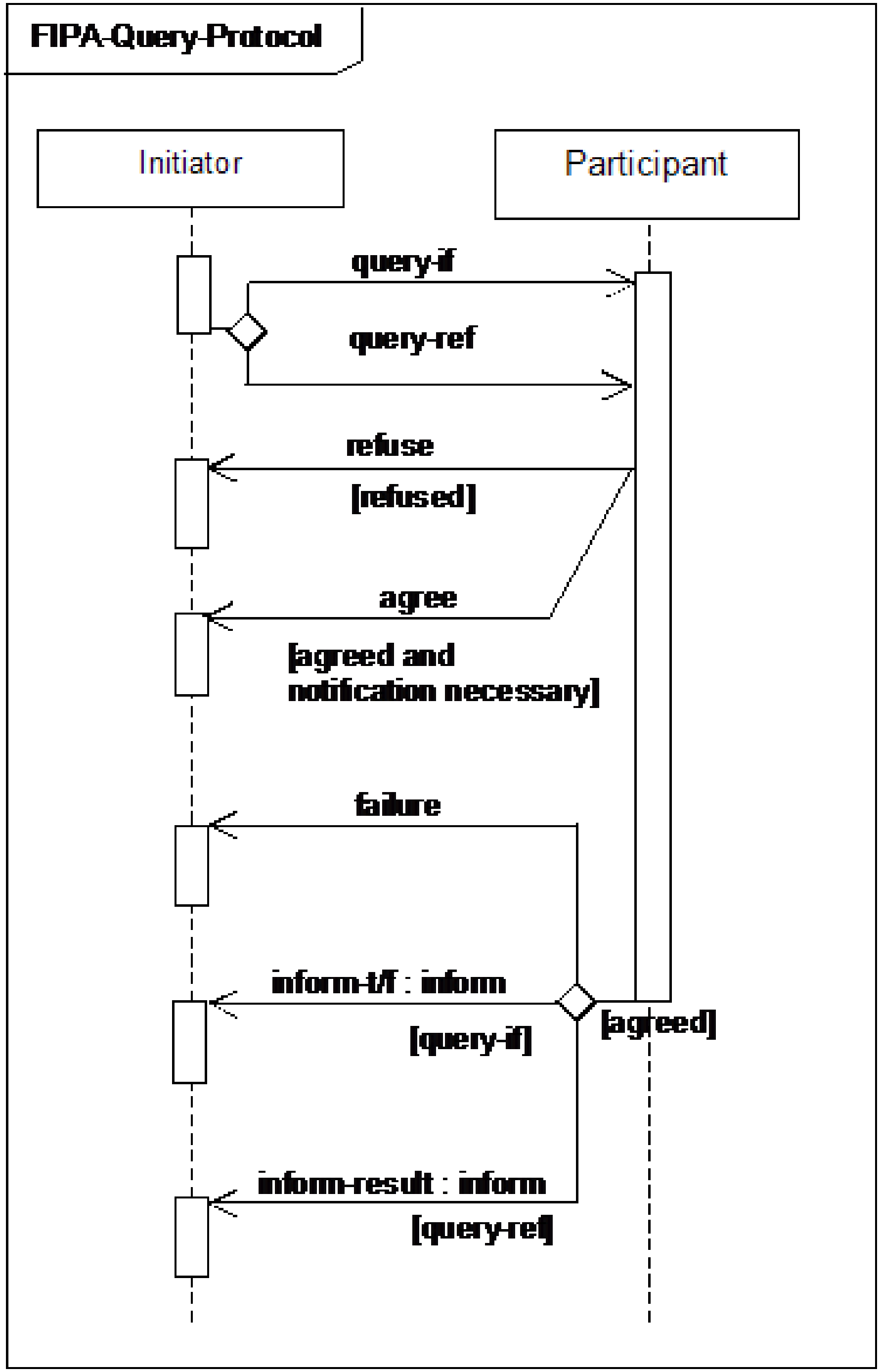




FIPA Query Interaction Protocol

Notes:

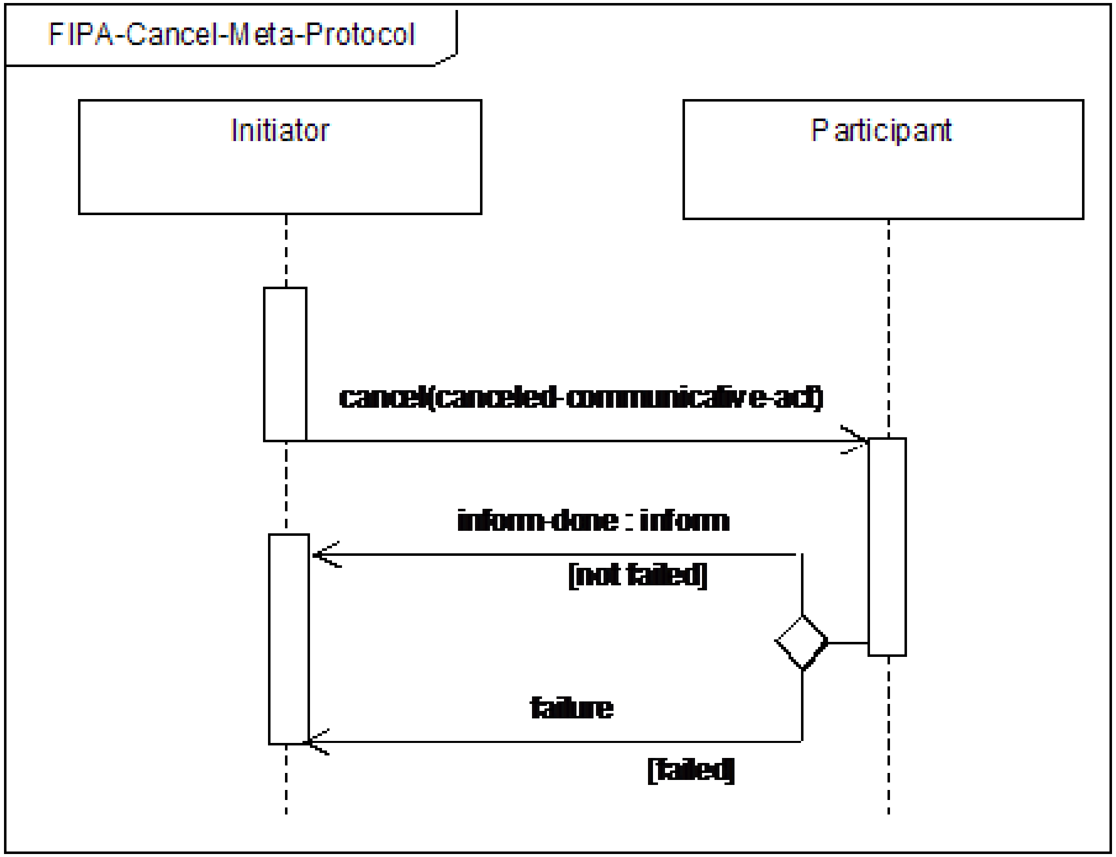
- Two kinds of queries
- Diamonds represent OR
- States are noted in brackets, e.g., [refused]
- The agree is optional
- The vertical dotted line is an agent *lifeline* and the rectangular box is a *thread*
- Two kinds of informs



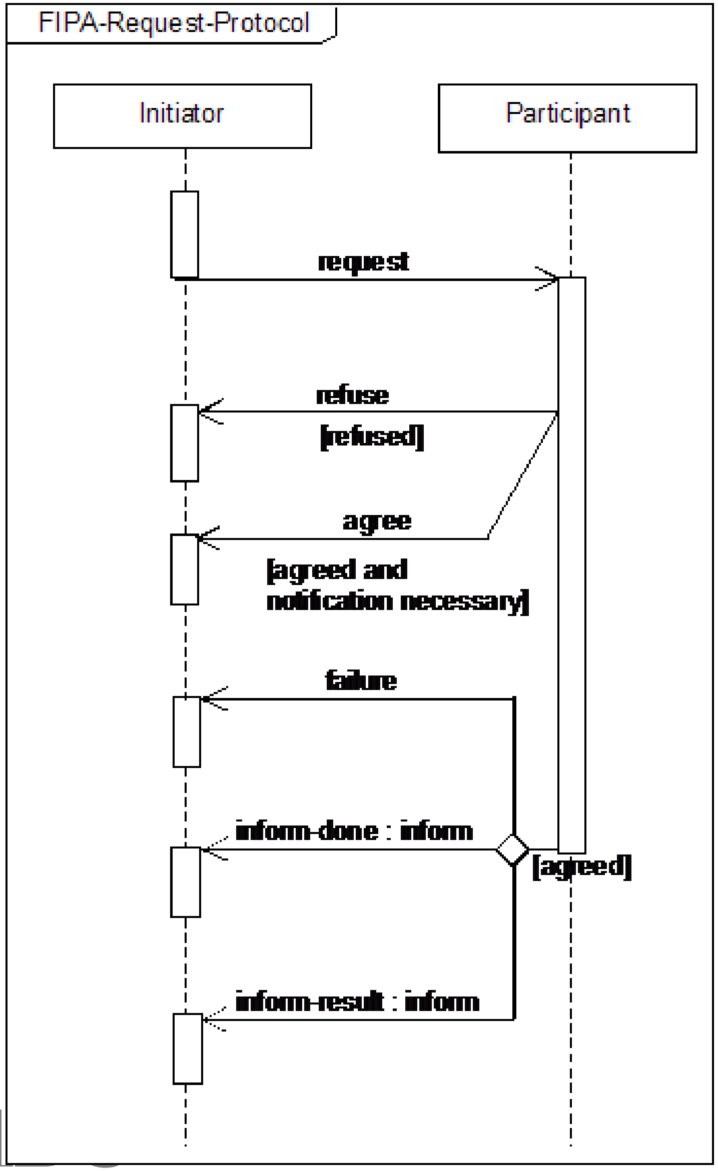
FIPA meta-protocols

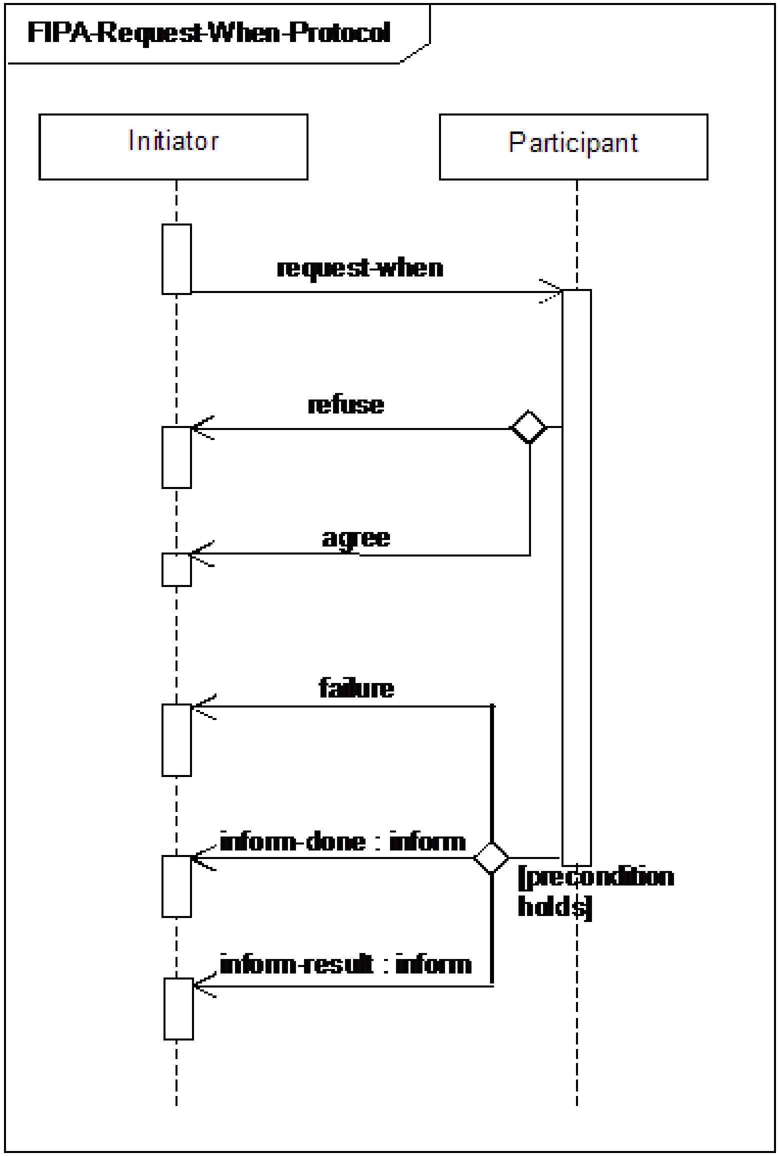
FIPA supports the notion of a meta protocol, two of which apply here.

- At any time, the receiver can reply with a **not-understood** CA, which terminates the interaction
- The sender can send a **cancel** CA following the protocol to the right.



FIPA request and request-when

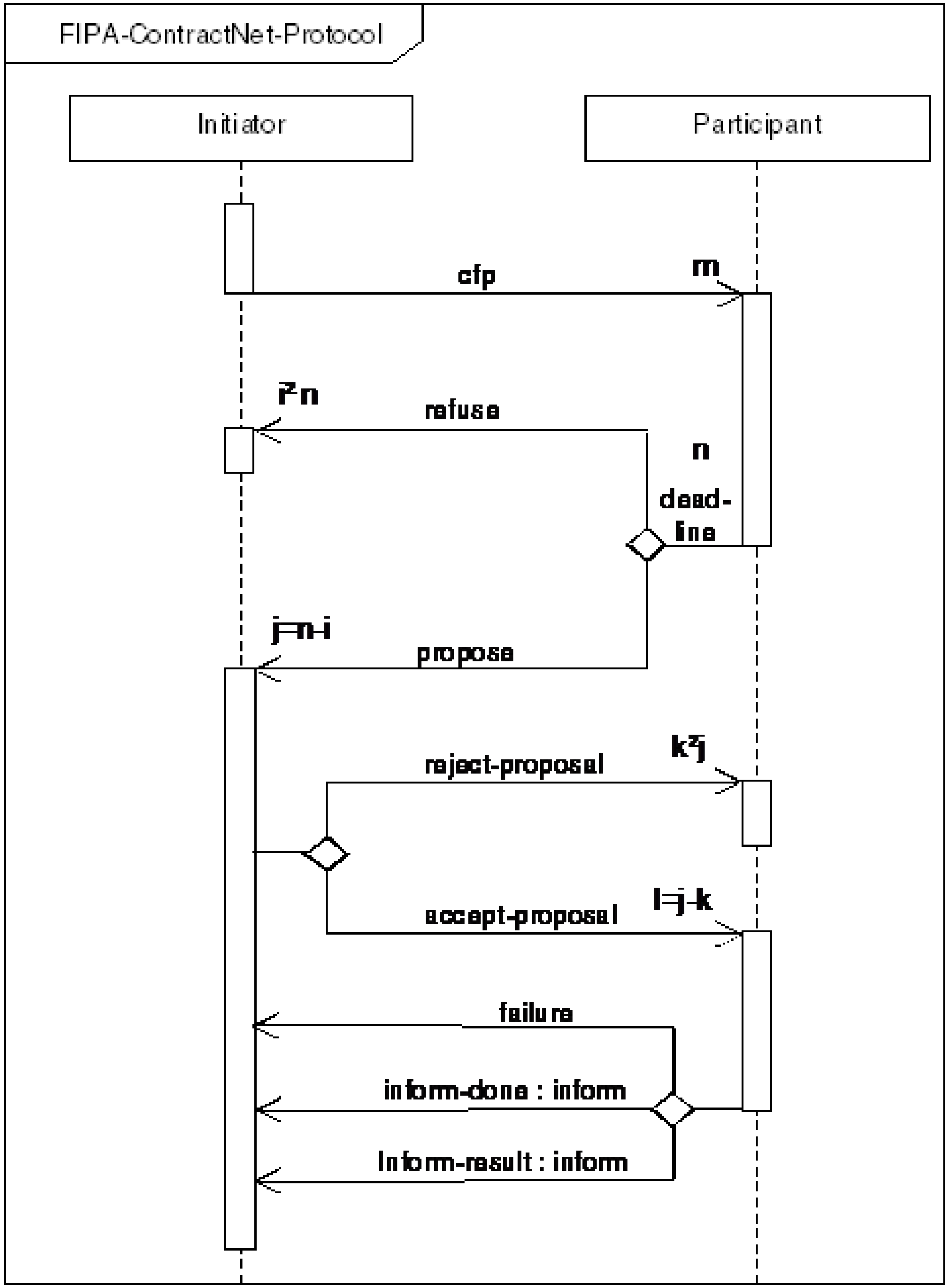




UMBC
an Honors University in Maryland

FIPA contract net

- Initiator sends a CFP to M recipients with a deadline
- I of which respond with a REFUSE an J of which with a PROPOSE.
- K of the proposals get a REJECT-PROPOSAL
- and L of them get an ACCEPT-PROPOSAL
- Those with accepted proposals respond with a FAILURE, a INFORM-DONE or an INFORM-RESULT.



UMBC
an Honors University in Maryland

FIPA content languages

- Like KQML, you can use virtually any content language you want.
- A FIPA-compliant must have a convention for representing all of the types of things you find in the message content, e.g.,
  - Propositions
  - Variables
  - Actions
  - Communicative acts

FIPA content languages

FIPA currently has four CLs in its specification library

- SL – semantic language. This is FIPA's original CL and is similar to KIF.
- CCL – constraint choice language. A language designed for CSP problems.
- KIF
- RDF

Others have been specified but not submitted to the approval process

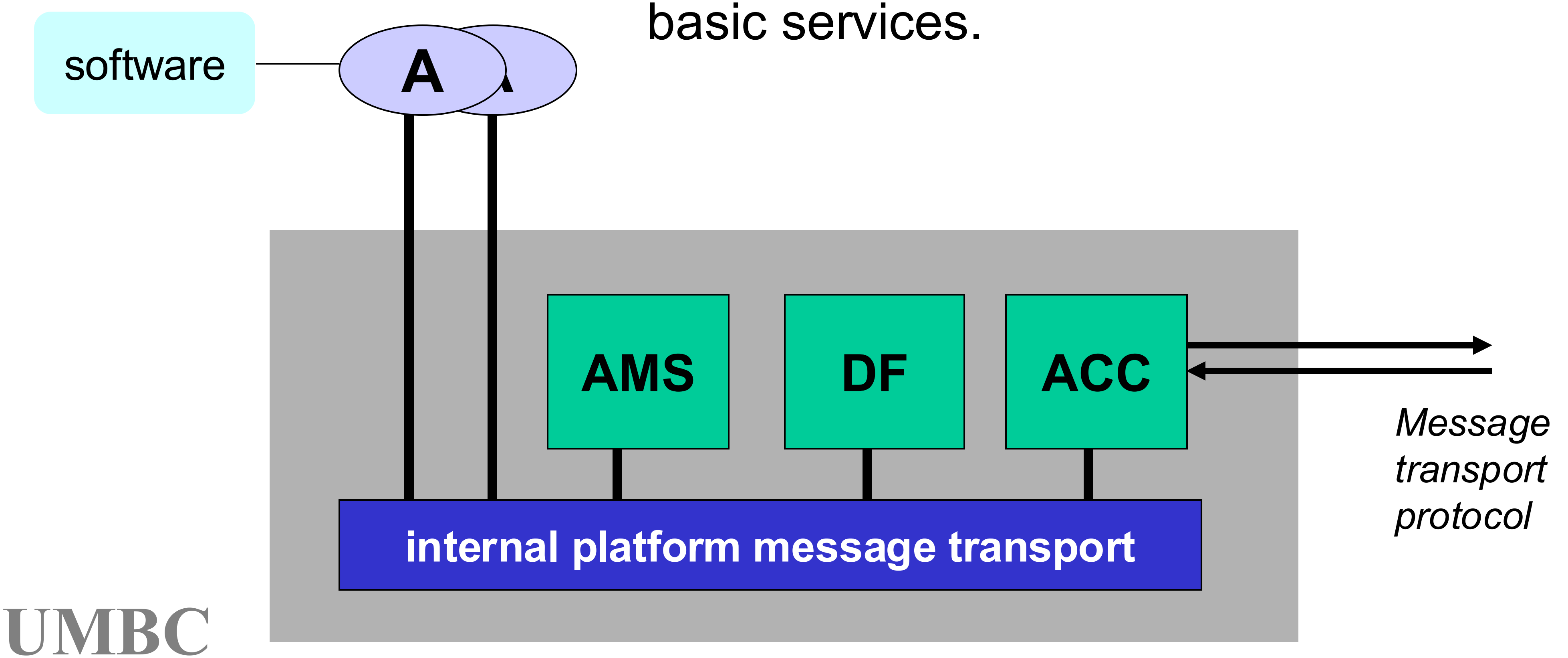
- OWL

FIPA

FIPA Agent Platform

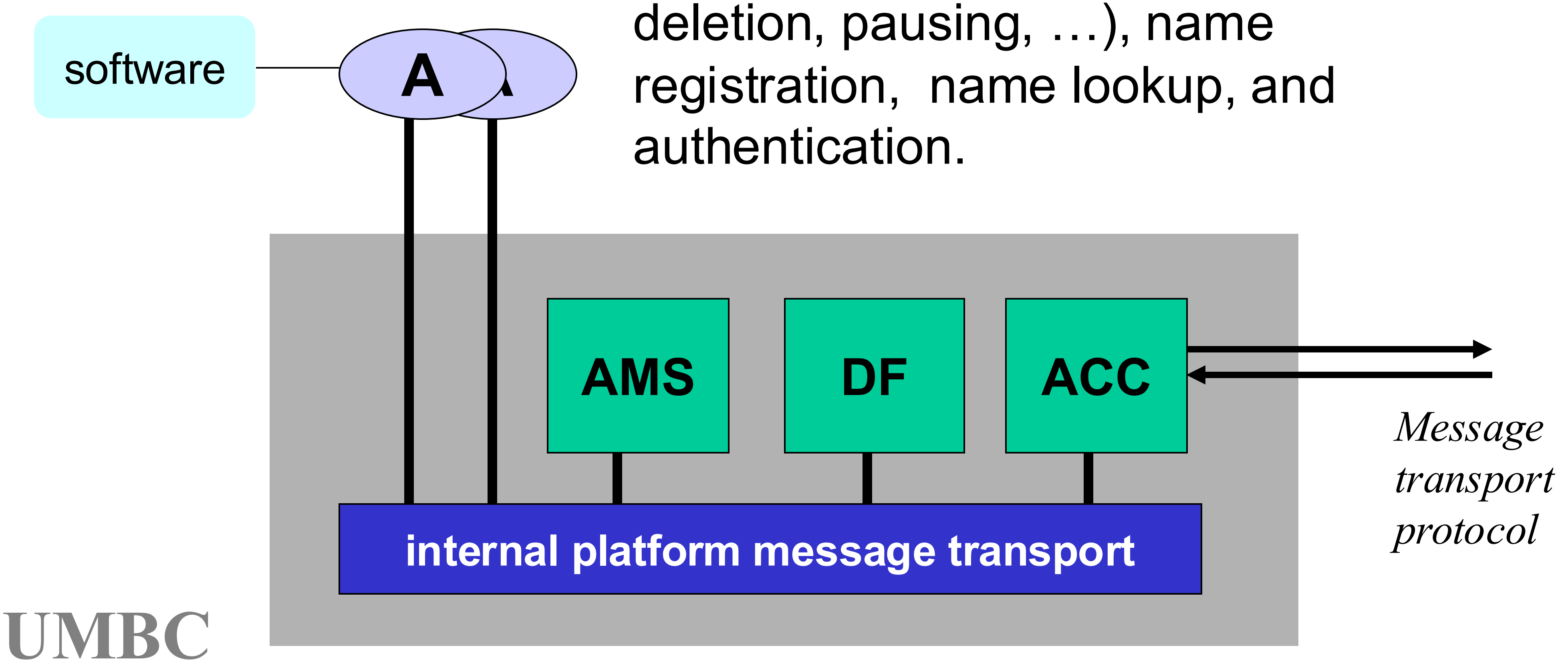
FIPA Agent Platform

Agents belong to one or more agent platforms which provide basic services.


software
A
A
AMS
DF
ACC
internal platform message transport
Message transport protocol

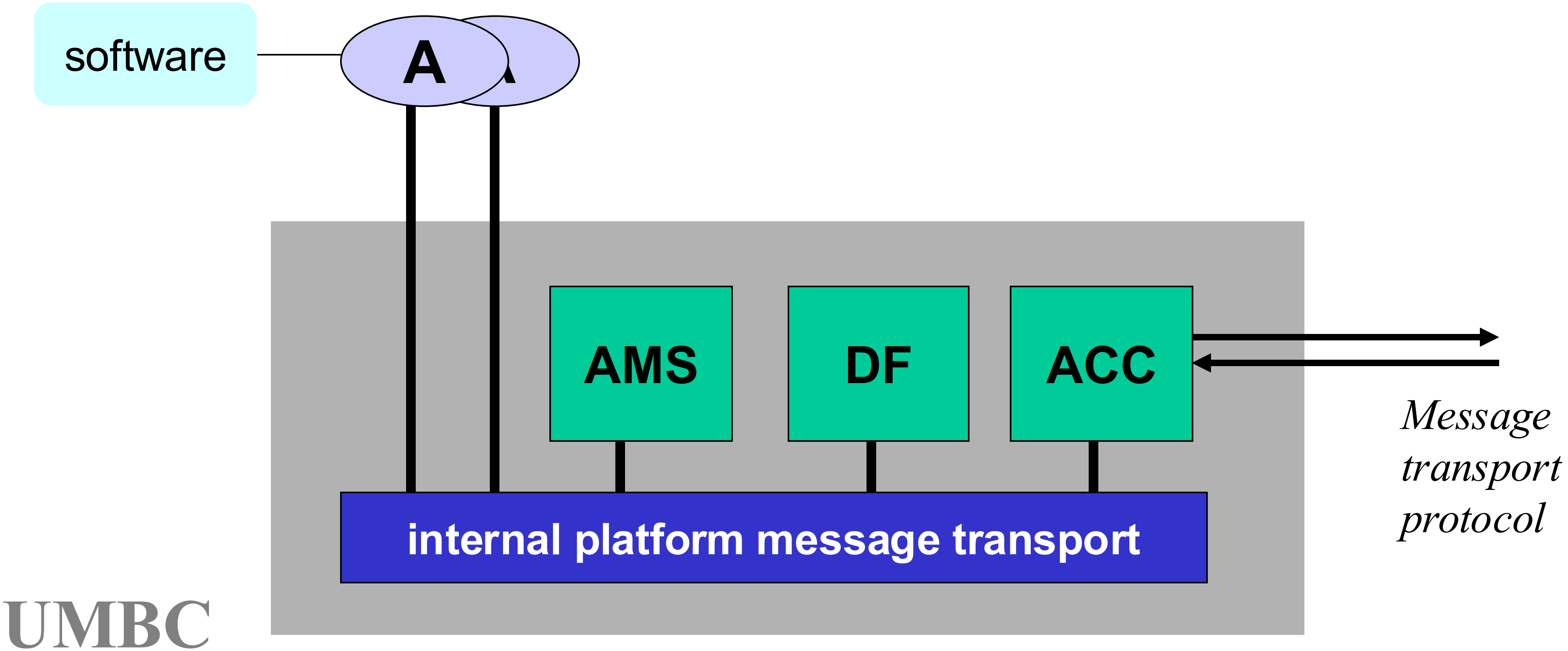
FIPA Agent Platform

The **AMS** (Agent Management System) provides services like lifecycle management (creation, deletion, pausing, …), name registration, name lookup, and authentication.


software
A
A
AMS
DF
ACC
internal platform message transport
Message transport protocol

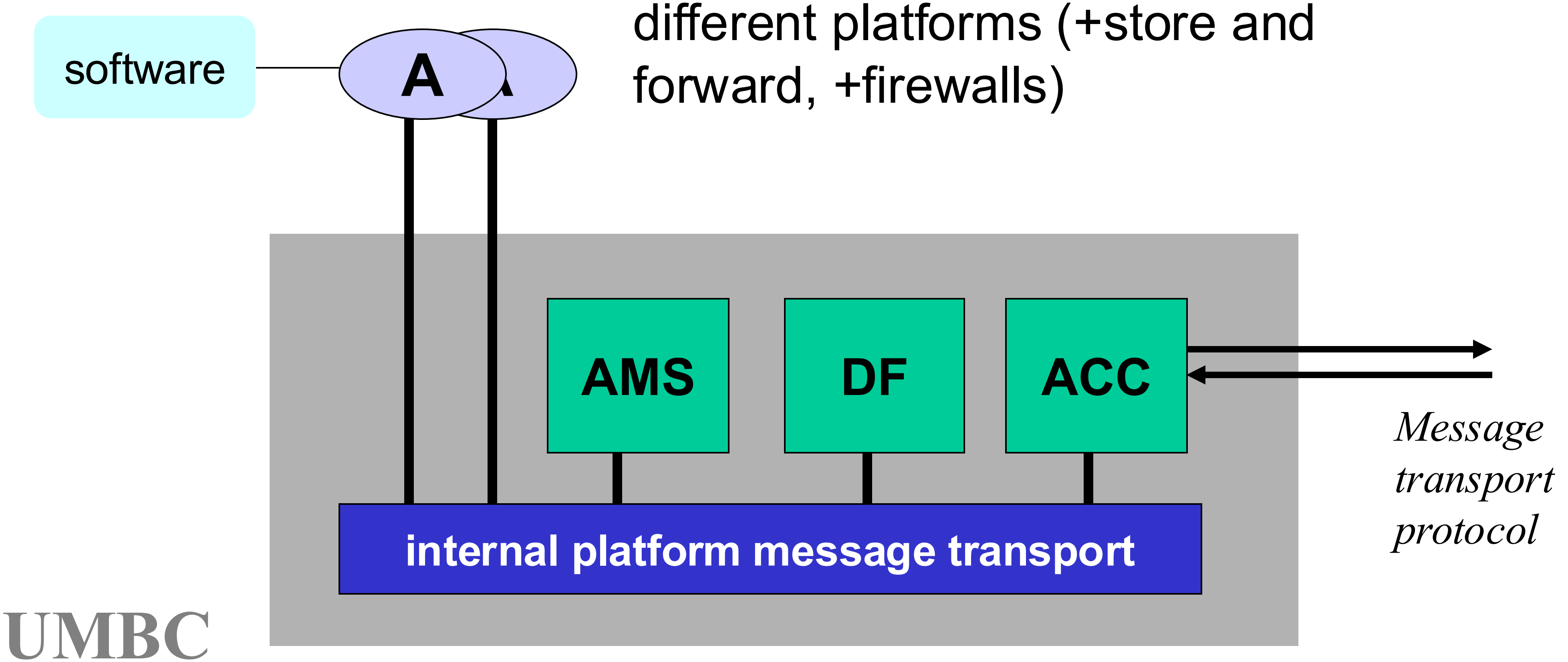
FIPA Agent Platform

The **DF** (Directory Facilitator) provides yellow pages services which describe the attributes and capabilities of agents in the platform.


software
A
A
AMS
DF
ACC
internal platform message transport
Message transport protocol

FIPA Agent Platform

The **ACC** (Agent Communication Channel) accepts and delivers message between agents on different platforms (+store and forward, +firewalls)

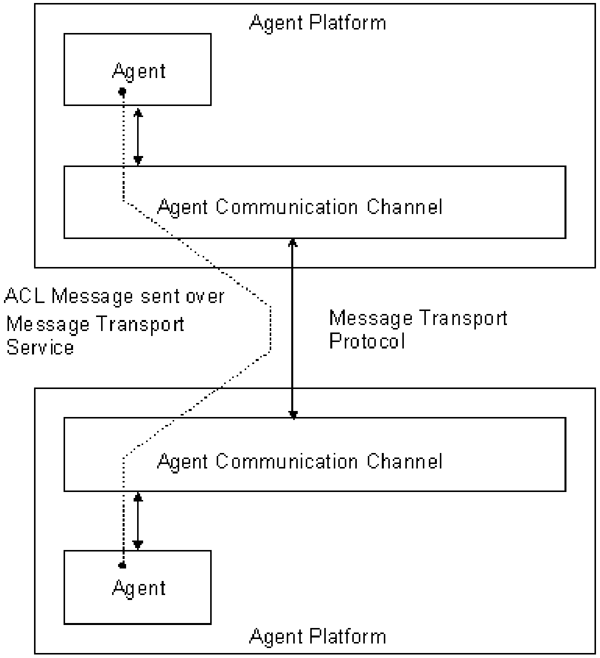

software
A
A
AMS
DF
ACC
internal platform message transport
Message transport protocol

UMBC
an Honors University in Maryland

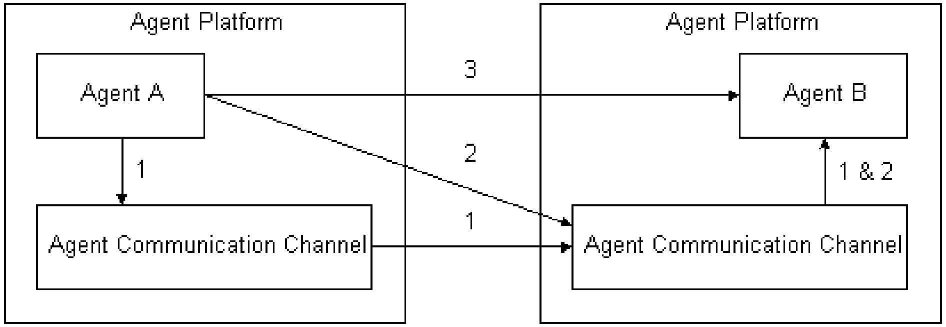
Message Transport

The message transport specifications include those for

- ACL representsatio encoded in …
  - Strings
  - XML
  - Bit efficient
- ACL message envelope encoded in …
  - XML
  - Bit efficient encoding
- Transport protocols over …
  - IIOP
  - HTTP
  - WAP


Agent Platform
Agent
Agent Communication Channel
ACL Message sent over Message Transport Service
Message Transport Protocol
Agent Communication Channel
Agent
Agent Platform

Overview of the Message Transport Model

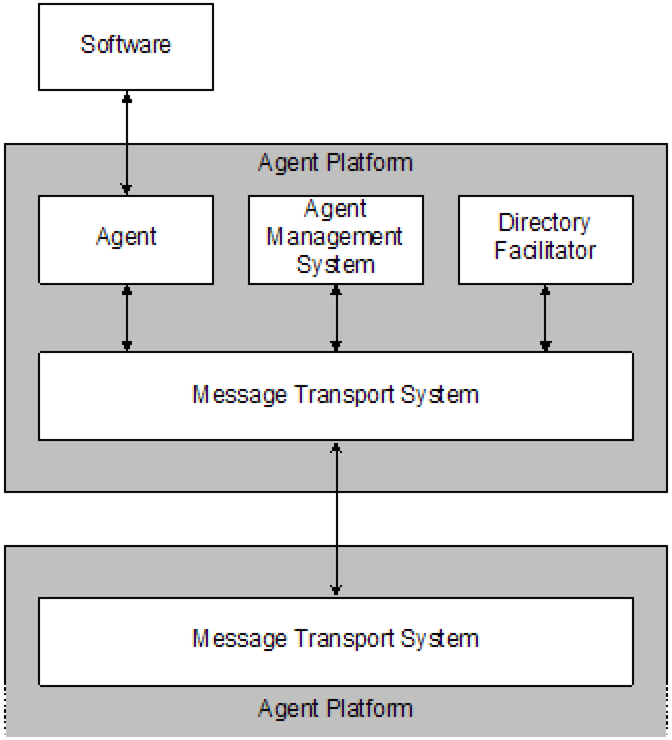


Three Methods of Communication between Agents on Different APs

FIPA agent management

FIPA Agent Management specifications deal with the control and management of agents within and across agent platforms, including

- Agent Naming
- Agent Management Service
  - Directory Facilitator
  - Agent Management System
  - Message Transport Service
- Agent Platform
  - Agent life-cycle
  - Agent registration
- Agent Management Ontology



UMBC
an Honors University in Maryland

Agent names

A FIPA agent identifier (AID) consists of

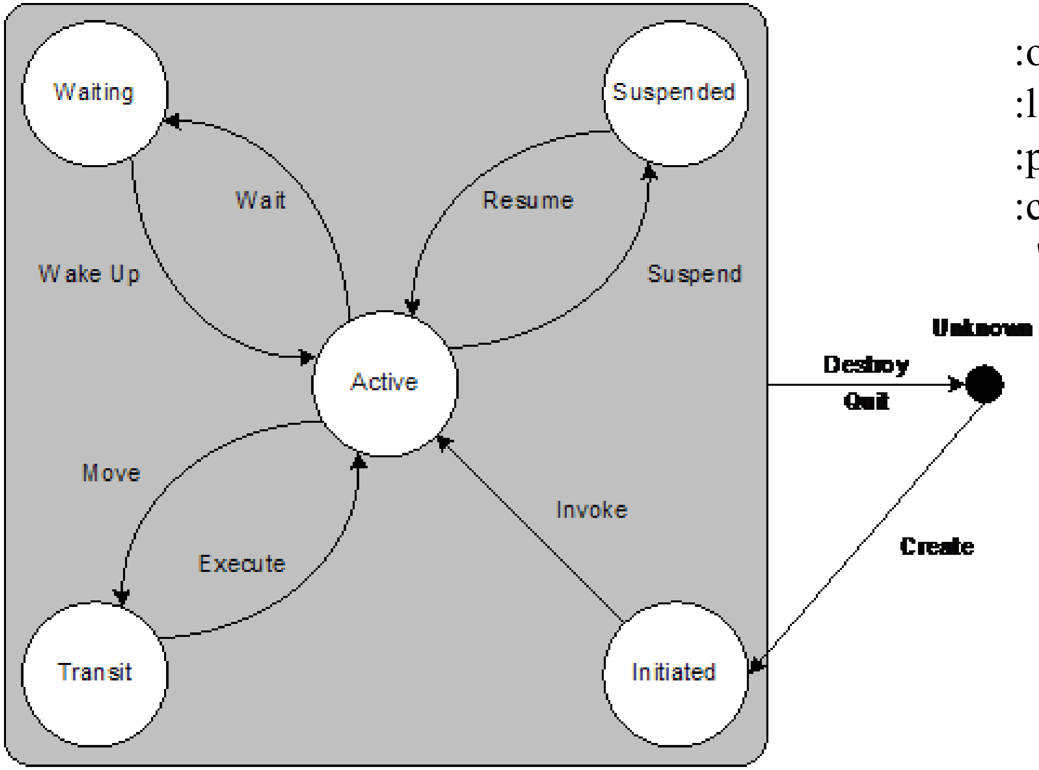
- Name : A Globally unique identifier (e.g. name @ platform-address)
- Addresses: A List of transport addresses (URLs)
- Resolver: AIDs of *name resolution service addresses* that can resolve the name into addresses.

Example:

```
(agent-identifier
  :name tagaMOA@taga.umbc.edu
  :addresses
   (sequence
     http://taga.umbc.edu/acc  iiop://taga.umbc.edu/ac http://taga.umbc.edu/acc))
```

Agent life cycle

An agent is in one of five possible states.



Agents can make requests to the AMS like …

```
(request
  :sender
    (agent-identifier
      :name discovery-agent@bar.com
      :addresses (sequence iiop://bar.com/acc))
  :receiver (set
    (agent-identifier
      :name ams@foo.com
      :addresses (sequence iiop://foo.com/acc)))
  :ontology fipa-agent-management
  :language fipa-sl0
  :protocol fipa-request
  :content
    "((action
      (agent-identifier
        :name ams@foo.com
        :addresses (sequence iiop://foo.com/acc))
      (register
        (:ams-description
          :name
            (agent-identifier
              :name discovery-agent@bar.com
              :addresses (sequence iiop://bar.com/acc))
          ...)))")
```

FIPA Platform Implementations

Many platforms have been implemented

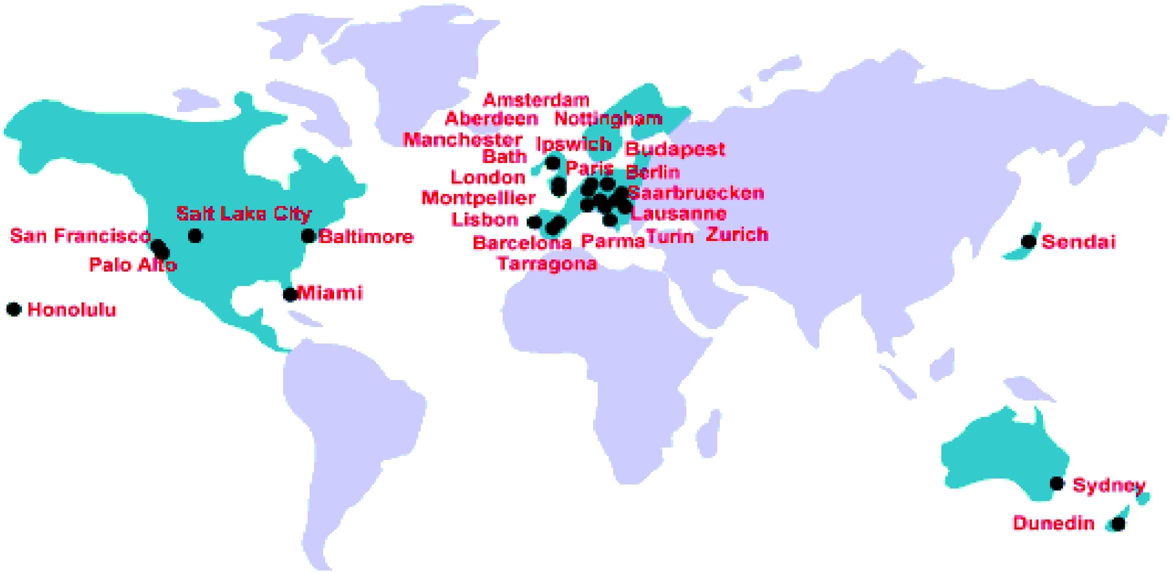
- –JADE/LEAP *(open source)*
- –FIPA-OS *(open source)*
- –*Java Agent Services API*
- –Zeus (BT)
- –Mecca (Siemens)
- –FIPA-SMART (Spawar)
- –Comtec *(in KAWA, a scheme dialect in Java)*
- –Fujitsu *(in April, a logic programming language)*
- –Jack intelligent agents
- –Grasshopper

and interoperability demonstrated.

agentcities

agentcities

- See http://agentcities.com/ http://agentcities.net/
- A network of FIPA platforms
- Each offers a set of services
- Sample services
  - Ping
  - Weather



=> See AAMAS'03 Workshop *Agentcities: Challenges in open agent environments*

5
ACL
Semantics

Outline

- Cohen & Levesque
  - Theory of Rational Agency
  - Cohen & Levesque on ACL Semantics
- KQML Semantics (Labrou)
- FIPA ACL Semantics
- Comparing ACL semantics approaches & Comments

The Cohen & Levesque Approach

- Most attempts for semantics for ACL descend from the work of Cohen & Levesque (C&L)
- Intention = Choice + Commitment
- Integration of Agent Theory and Semantics of Communication Primitives
- A (partial) theory of rational agency
- Possible-worlds semantics

Commitments and Intentions

- Internal Commitment:
  - (P-GOAL x p q) =

    (1) (BEL x $\neg$p) $\wedge$

    (2) (GOAL x (LATER p)) $\wedge$

    (3) [KNOW x (PRIOR [(BEL x p) $\vee$ (BEL x □$\neg$p) $\vee$ (BEL x □$\neg$q)]

    $\neg$ [GOAL x (LATER p)])]

  meaning

  "(1) agent x believes p is currently false

  (2) chooses that it be true later

  (3) and x knows that before abandoning that choice, he must either believe it is true, or that it will never be true, or that some q (an escape clause) is false"

Intention

The p of P-GOAL

What has just happened, i.e., x believes that a will happen next and a did happen

- (INTEND x a q) =
  (P-GOAL x [DONE x (BEL x (HAPPENS a))?;a] q)
  - x has the persistent goal of reaching a state at which it believes that a will happen, after which (state) a does happen
- Intending is a special kind of commitment
- The agent is committed to arriving at a state in which he is about to do the intended action next
- Thus an agent cannot be committed to doing something accidentally or unknowingly
- “I intend for the sun to rise tomorrow” vs
  “I intend to get an “A” in this course”

Semantics for INFORM

- {INFORM speaker listener e p} =

  {ATTEMPT speaker listener e
  (know listener p)
  [BMB listener speaker
  (P-GOAL speaker (KNOW listener (KNOW speaker P)))]}

The “honest effort”

- An INFORM is defined as an attempt in which to make an “honest effort”, the speaker is committed to making public that he is committed to the listener’s knowing that he (the speaker) knows p.

ACL Semantics

KQML Semantics

Which Agent States? (Labrou 1996)

- ***Preconditions*** indicate the necessary state for an agent in order to send a performative and for the receiver to accept it and successfully process it.
- ***Postconditions*** describe the states of both interlocutors after the successful utterance of a performative (by the sender) and after the receipt and processing (but before a counter utterance) of a message (by the receiver).
- ***Preconditions*** indicate what can be assumed to be the state of the interlocutors involved in an exchange. Similarly, the *postconditions* are taken to describe the states of the interlocutors assuming the successful performance of the communication primitive

Semantics for TELL

TELL(A,B,X)

- A states to B that A believes X to be true (for A).
- bel(A,X)
- Pre(A): bel(A,X) $\wedge$ know(A,want(B,know(B,S)))

  where S may be bel(B,X) or NOT(bel(B,X))

  Pre(B): intend(B,know(B,S))
- Post(A): know(A,know(B,bel(A,X)))

  Post(B): know(B,bel(A,X))
- Completion: know(B,bel(A,X))
- The completion condition and postconditions hold unless a SORRY or ERROR suggests B's inability to properly acknowledge the TELL.

ACL Semantics

FIPA ACL Semantics

Outline of FIPA ACL Semantics

- A primitive’s meaning is defined in terms of FPs and REs
- The Feasibility Preconditions of a CA define the conditions that ought to be true before an agent may plan to execute the CA
- The Rational Effect is the effect that an agent hopes to bring about by performing an action (but with no guarantee that the effect will be achieved)
- The FPs and the REs involve agents state descriptions that are given in SL

An example of FIPA ACL semantics (inform)

<*i*, inform(*j*, ϕ)>

FP: $B_i\phi \wedge \neg B_i(Bif_j\phi \vee Uif_j\phi)$

RE: $B_j\phi$

Agent i informs agent j that (it is true that) it is raining today:

```
(inform
    :sender i
    :receiver j
    :content "weather(today,raining)"
    :language Prolog
    :ontology weather42)
```

Another example of FIPA ACL semantics (request)

<*i*, request(*j*, a)>

FP: FP(a) [i\j] $\wedge$ B_i Agent(*j*, a) $\wedge$ $\neg B_i$ I_j Done(a)

RE: Done(a)

Agent i requests j to open a file:

```
(request
   :sender i
   :receiver j
   :content "open \"db.txt\" for input"
   :language vb)
```

How do KQML and FIPA ACL differ?

- Different semantics; mapping of KQML performatives to FIPA primitives and vice versa is a futile exercise.
- Different treatment of the “administration primitives”; in FIPA ACL register, unregister, etc., are treated as requests for action with reserved (natural language) meaning
- No “facilitation primitives”, e.g., broker, recommend, recruit, etc., in FIPA ACL
- Reserved content language: a very murky issue ...

Shortcomings of Current ACL semantics

- Intentional level description: which mental attitudes, what definitions?
- Problems with mental attitudes: from theory to practice
- Can all desirable communication primitives be modeled after speech acts? Should they?
- Flexible description of agents' capabilities and advertising of such capabilities.
- How can we test an agent's compliance with the ACL?
- Ease of extending an ACL

Addressing the shortcomings of the semantics with conversations

- Both KQML and FIPA ACL include specifications for conversations (or conversation protocols)
- Conversations are not part of the semantic definition of the ACL
- Conversations shift the focus to an agent’s observable behavior
- Programmers might find conversations more useful than formal semantics
- The meaning of primitives is often context/situation dependant and conversations can accommodate context

Conversations: A new trend

- A conversation is a sequence of messages that agents “agree” to observe, when engaging in message exchange.
- A conversation is focusing on the observed actions of an agent’s communicative behavior
- Both KQML and FIPA ACLs have taken an interest in conversations (protocols).
- Conversation may well be the only enforceable compliance-checking for communicating agents.

Conversations

- Conversations define allowed/useful/desirable sequences of messages for particular tasks and indicate where/how messages “fit” in exchanges.
- Desiderata:
  - Allow more intuitive and convenient method for handling messages in context.
  - Through conversation composition, scale to varying levels of granularity.
  - Provide conversation management independent of agent implementation.
  - Facilitate communication through conversation sharing.

Conversation Policies Regulate Other Features of Agent Communication

- Conversation management assumptions
  - Exception handling
  - Initiation, termination, interruption
  - Concurrency and turn-taking
  - Clarification, repair, insertion sequences
- Uptake acknowledgment assumptions
- Pragmatic principles
  - Preferences on semantically equivalent messages
- Task-specific sequences with guaranteed properties
  - Fairness

The Role of Conversation Policies

- Modern ACLs are powerful enough to:
  - Encompass several different ways to achieve the same goal
  - Achieve several different goals with the same message
- If there was just one way to achieve any goal, CPs would not be needed.
- Conversation Policies define conventional ways to ***constrain*** the expressive power of an ACL
  - Reduce the depth of modeling of other agents (and *v.v.*); and publicly expose the agent's goals.
  - Reduce uncertainty about the next conversational move
- What kind of constraints have we developed?

Alternatives

- Alternative ACLs
- Alternatives to ACLs

Other Agent Communication Frameworks

There are many other ACL based systems, none as established as FIPA

- CoABS Grid
  - Developed by DARPA CoABS program, based on Jini using some FIPA and KQML components
- Cougaar
  - A DARPA sponsored open source agent infrastructure
- Java Agent Services
  - Java APIs based on FIPA's abstract architecture
- Open Agent Architecture
  - Developed at SRI

Alternatives to ACLs

- There are many alternatives to using ACLs for communicating and sharing information.
- From oldest to newest...
  - Natural language (Espanol)
  - Database languages (SQL, …)
  - Domain dependant (EDI, …)
  - Distributed object systems (CORBA, ...)
  - OKBC
  - Service languages (e-speak, BizTalk, …)
  - P2P and Grid computing
  - Web languages (XML, RDF, DAML+OIL)
- One size won’t fit all, so we need to appreciate the strengths and weaknesses.
- We will also see mixing, matching and morphing

Thoughts on alternatives to ACLs

- One size won't fit all
- General purpose vs. specialized languages
- Things will continue to evolve
- Industry has not invested in ACLs, preferring home-grown approaches (.NET, Java) and ad-hoc solutions, often based on XML (e.g., WSDL).
- Advice: if you are implementing a single application, before you go with an ACL, convince yourself that it's not right for a mature alternative, such as:

  (1) a database approach

  (2) a distributed objects approach

6
ACL’s and Semantic Web Motivation

UMBC

an Honors University in Maryland

"XML is Lisp's bastard nephew, with uglier syntax and no semantics. Yet XML is poised to enable the creation of a Web of data that dwarfs anything since the Library at Alexandria."

-- Philip Wadler, *Et tu XML? The fall of the relational empire*, VLDB, Rome, September 2001.

“The web has made people smarter. We need to understand how to use it to make machines smarter, too.”

-- Michael I. Jordan, paraphrased
from a talk at AAAI, July 2002
by Michael Jordan (UC Berkeley)

“The Semantic Web will globalize KR, just as the WWW globalize hypertext”

-- Tim Berners-Lee

“The multi-agent systems paradigm and the web both emerged around 1990. One has succeeded beyond imagination and the other has not yet made it out of the lab.”

-- Anonymous, 2001

IOHO

- The web is like a universal acid, eating through and consuming everything it touches.
  - Web principles and technologies are equally good for wireless/pervasive computing.
- The semantic web is our first serious attempt to provide semantics for XML sublanguages.
- The semantic web will provide mechanisms for people and machines (agents, programs, CGI scripts) to come together.
  - Solving the symbol grounding problem?

Semantic web and agents

- The hope is that the semantic web and agents will be a perfect marriage.
  - Agents need content
  - The semantic web needs actors and service providers
- SW languages support distributed, multi-authored ontologies.
- Using content languages based on XML and backed by standards organizations (W3C) will aid acceptance by industry.
- Next: overview of Semantic Web

Using XML to describe ACL messages

- Both KQML and FIPA ACL are using a LISP-like syntax to describe properly-formed ACL messages
- ACL messages have “deep” semantics (KR-like) than account for the Communicative Act, the Sender and the Receiver
- The deep semantics, in the case of FIPA ACL are described in SL
- A ACL message as a syntactic object has parameters that are not accounted for in the semantics (language, ontology, in-reply-to, etc.)

Using XML to describe ACL messages (continued)

- Syntactically, ACL messages introduce pragmatic elements and a particular syntax useful for parsing and routing.
- The syntactic form (e.g., LISP-like) need not be unique.
- Syntactically, ACL messages can be thought as having an "abstract syntax".
- The abstract syntax "allows" for multiple **syntactic representations** or **encodings**
- Examples of encodings are: Lisp-like balanced parenthesis list, XML or even a Java structure

Advantages of XML-encodings

- Parsing ACL messages is a big overhead of agent development.
- The XML encoding is easier to develop parsers for:
  - one can use off-the-shelf XML parsers
  - a modified DTD does not mean re-writing the parser
- ACL messages are more WWW-friendly
  - easier integration with web-based technologies
  - potential for taking advantages of WWW-solutions to outstanding ACL issues (e.g., security)
- ACL messages introduce a pragmatics layer that is unaccounted at the semantic level
- Using XML, helps better address these pragmatic aspects through the use of links. Links point to additional information.
  - links can assist with ontological problems (defining and sharing ontologies)
  - links can point to agent capability and identity information, protocols, even semantics.

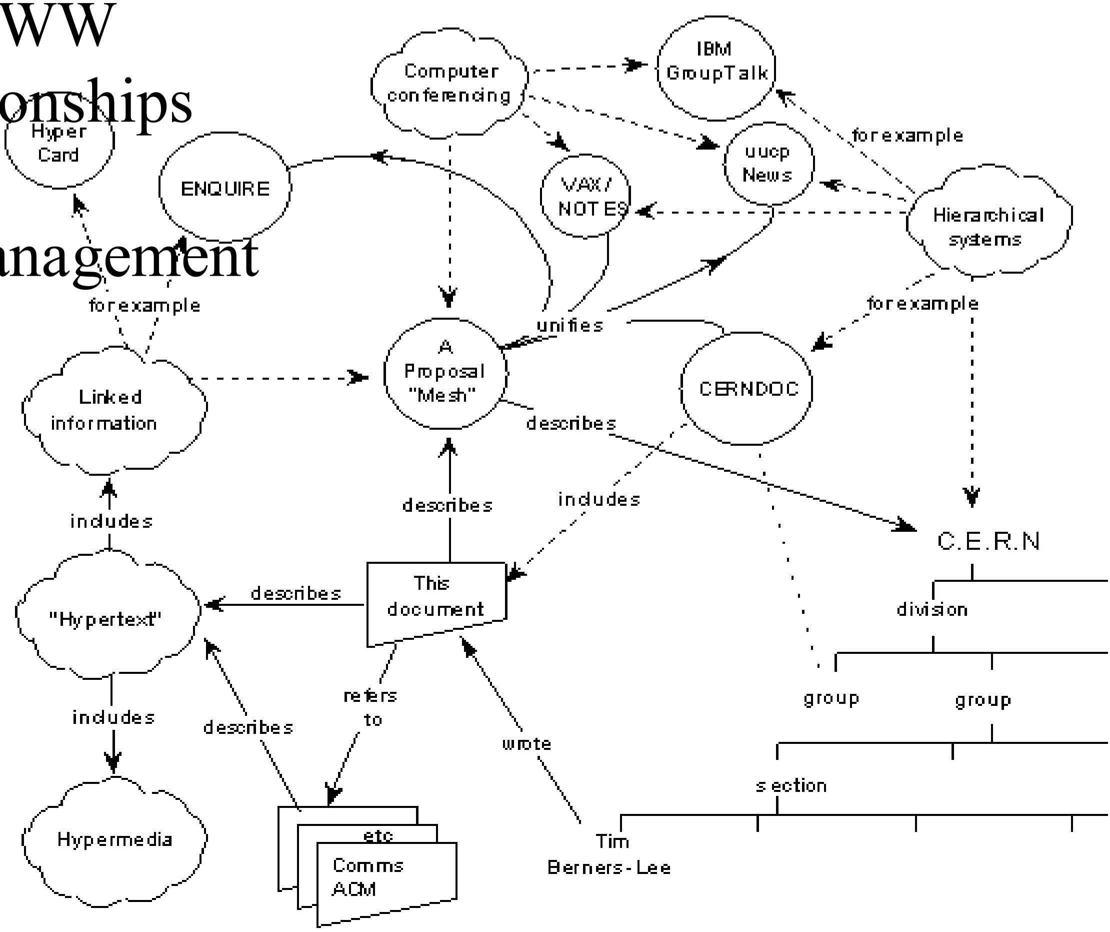
Disadvantages

- XML encodings tend to be
  - Much more verbose
  - Somewhat harder for people to read (perhaps because of their verbosity)
  - Lacking in some common, useful syntactic components, such as variables, operators, lists, …
- These deficiencies are probably all related.
- XML is no one’s favorite syntax
- Which is a reason why it’s a good interlingua

7
Semantic
Web

Origins of the Semantic Web

Tim Berners-Lee's original 1989 WWW proposal described a web of relationships among named objects that unified many info. management tasks.



Capsule history

- Guha designed MCF (~94)
- XML+MCF=>RDF (~96)
- RDF+OO=>RDFS (~99)
- RDFS+KR=>DAML+OIL (00)
- W3C's SW activity (01)
- W3C's OWL (02?)

http://www.w3.org/History/1989/proposal.html

W3C's Semantic Web Goals

- **Focus on machine consumption:**

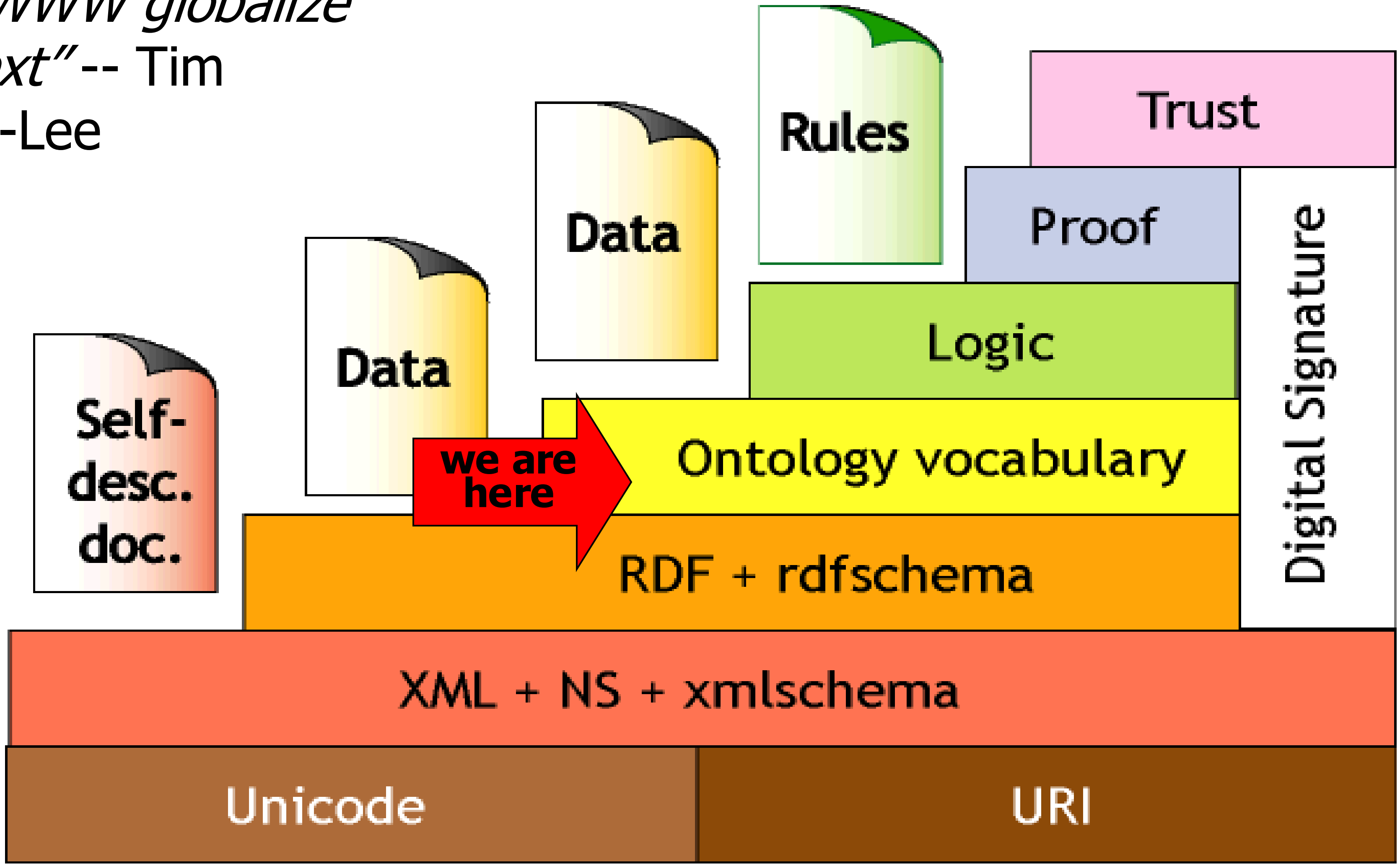
  "The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation." -- Berners-Lee, Hendler and Lassila, The Semantic Web, Scientific American, 2001

- **The current Web *stores* things whereas the semantic Web *does* things.**

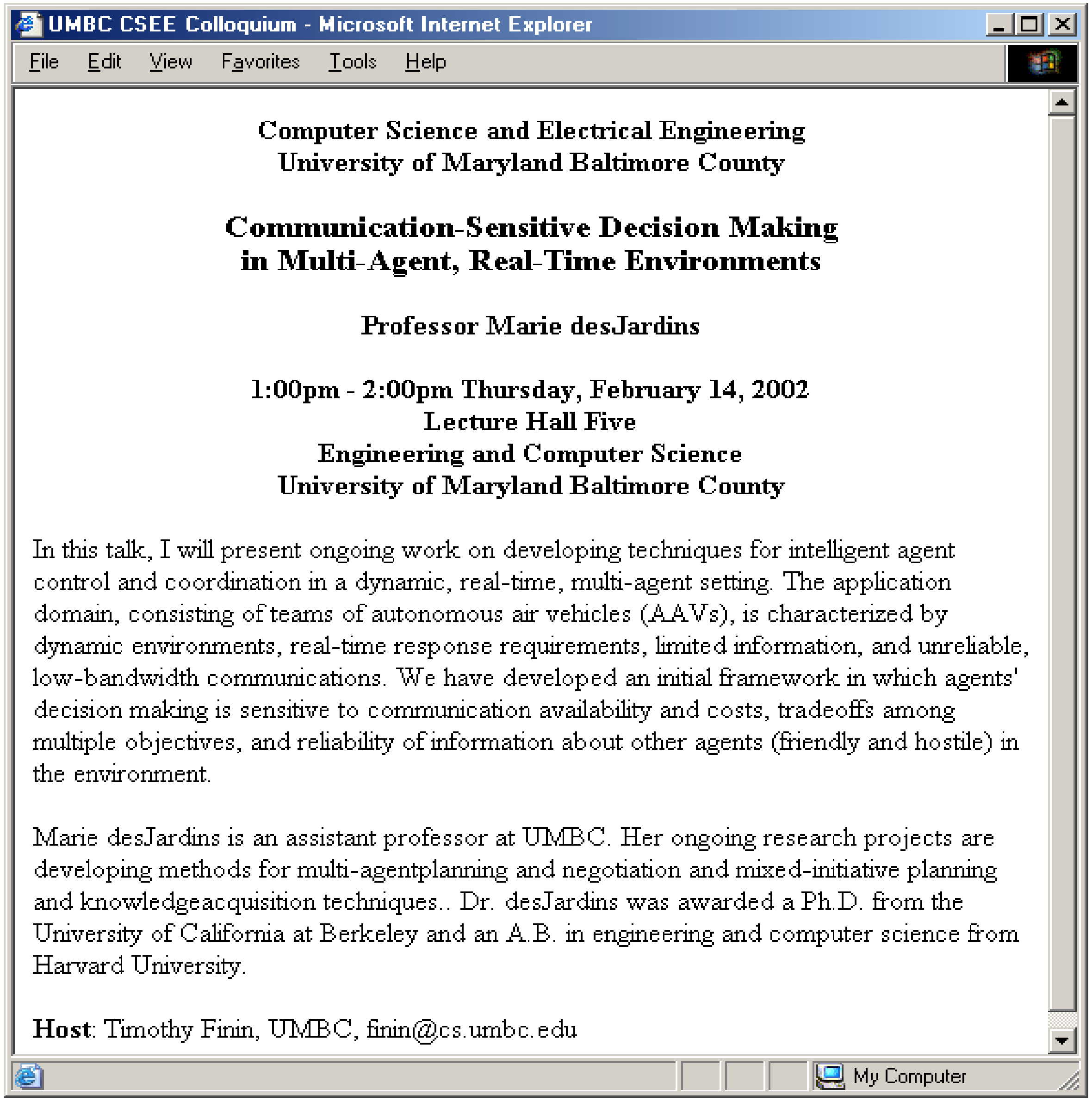
=> *Good match for agents!*

TBL’s semantic web vision

"The Semantic Web will globalize KR, just as the WWW globalize hypertext" -- Tim Berners-Lee



Why is this hard?



UMBC CSEE Colloquium - Microsoft Internet Explorer
File Edit View Favorites Tools Help
My Computer

Computer Science and Electrical Engineering
University of Maryland Baltimore County

Communication-Sensitive Decision Making
in Multi-Agent, Real-Time Environments

Professor Marie desJardins

1:00pm - 2:00pm Thursday, February 14, 2002
Lecture Hall Five
Engineering and Computer Science
University of Maryland Baltimore County

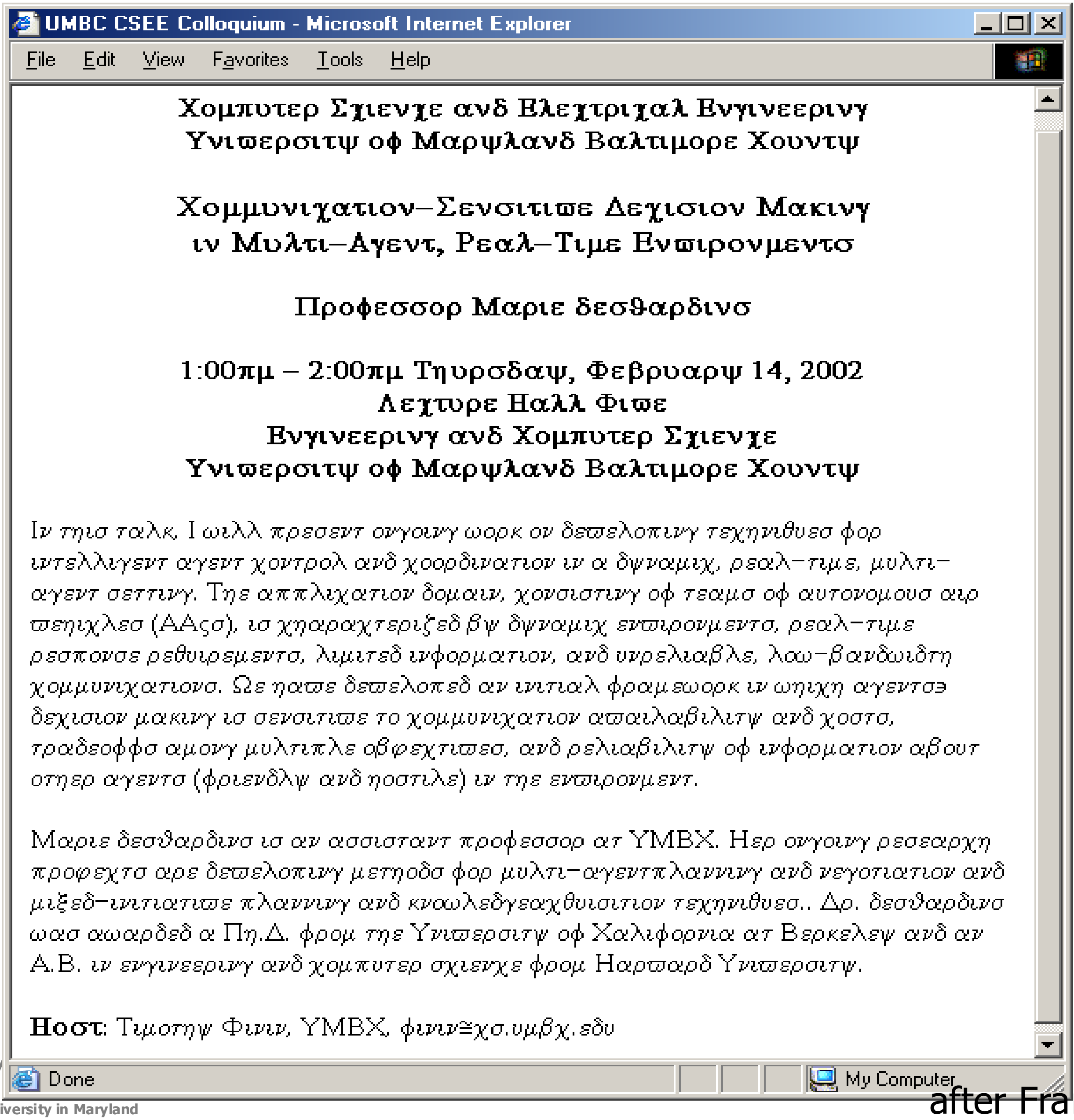
In this talk, I will present ongoing work on developing techniques for intelligent agent control and coordination in a dynamic, real-time, multi-agent setting. The application domain, consisting of teams of autonomous air vehicles (AAVs), is characterized by dynamic environments, real-time response requirements, limited information, and unreliable, low-bandwidth communications. We have developed an initial framework in which agents' decision making is sensitive to communication availability and costs, tradeoffs among multiple objectives, and reliability of information about other agents (friendly and hostile) in the environment.

Marie desJardins is an assistant professor at UMBC. Her ongoing research projects are developing methods for multi-agentplanning and negotiation and mixed-initiative planning and knowledgeacquisition techniques.. Dr. desJardins was awarded a Ph.D. from the University of California at Berkeley and an A.B. in engineering and computer science from Harvard University.

Host: Timothy Finin, UMBC, finin@cs.umbc.edu

after Frank van Harmelen and Jim Hendler

What a web page looks like to a machine…



Χομπυτερ Σχιενχε ανδ Ελεχτριχαλ Ενγινεερινγ
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ

Χομμυνιχατιον–Σενσιτιϖε Δεχισιον Μακινγ
ιν Μυλτι–Αγεντ, Ρεαλ–Τιμε Ενϖιρονμεντσ

Προφεσσορ Μαριε δεσϑαρδινσ

1:00πμ – 2:00πμ Τηυρσδαψ, Φεβρυαρψ 14, 2002
Λεχτυρε Ηαλλ Φιϖε
Ενγινεερινγ ανδ Χομπυτερ Σχιενχε
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ

Ιν τηισ ταλκ, Ι ωιλλ πρεσεντ ονγοινγ ωορκ ον δεϖελοπινγ τεχηνιθυεσ φορ ιντελλιγεντ αγεντ χοντρολ ανδ χοορδινατιον ιν α δψναμιχ, ρεαλ−τιμε, μυλτι−αγεντ σεττινγ. Τηε αππλιχατιον δομαιν, χονσιστινγ οφ τεαμσ οφ αυτονομουσ αιρ ϖεηιχλεσ (ΑΑςσ), ισ χηαραχτεριζεδ βψ δψναμιχ ενϖιρονμεντσ, ρεαλ−τιμε ρεσπονσε ρεθυιρεμεντσ, λιμιτεδ ινφορματιον, ανδ υνρελιαβλε, λοω−βανδωιδτη χομμυνιχατιονσ. Ωε ηαϖε δεϖελοπεδ αν ινιτιαλ φραμεωορκ ιν ωηιχη αγεντσ∋ δεχισιον μακινγ ισ σενσιτιϖε το χομμυνιχατιον αϖαιλαβιλιτψ ανδ χοστσ, τραδεοφφσ αμονγ μυλτιπλε οβφεχτιϖεσ, ανδ ρελιαβιλιτψ οφ ινφορματιον αβουτ οτηερ αγεντσ (φριενδλψ ανδ ηοστιλε) ιν τηε ενϖιρονμεντ.

Μαριε δεσϑαρδινσ ισ αν ασσισταντ προφεσσορ ατ ΥΜΒΧ. Ηερ ονγοινγ ρεσεαρχη προφεχτσ αρε δεϖελοπινγ μετηοδσ φορ μυλτι−αγεντπλαννινγ ανδ νεγοτιατιον ανδ μιξεδ−ινιτιατιϖε πλαννινγ ανδ κνοωλεδγεαχθυισιτιον τεχηνιθυεσ.. Δρ. δεσϑαρδινσ ωασ αωαρδεδ α Πη.Δ. φρομ τηε Υνιϖερσιτψ οφ Χαλιφορνια ατ Βερκελεψ ανδ αν Α.Β. ιν ενγινεερινγ ανδ χομπυτερ σχιενχε φρομ Ηαρϖαρδ Υνιϖερσιτψ.

Ηοστ: Τιμοτηψ Φινιν, ΥΜΒΧ, φινιν≅χσ.υμβχ.εδυ

And understanding natural language is easier than images!

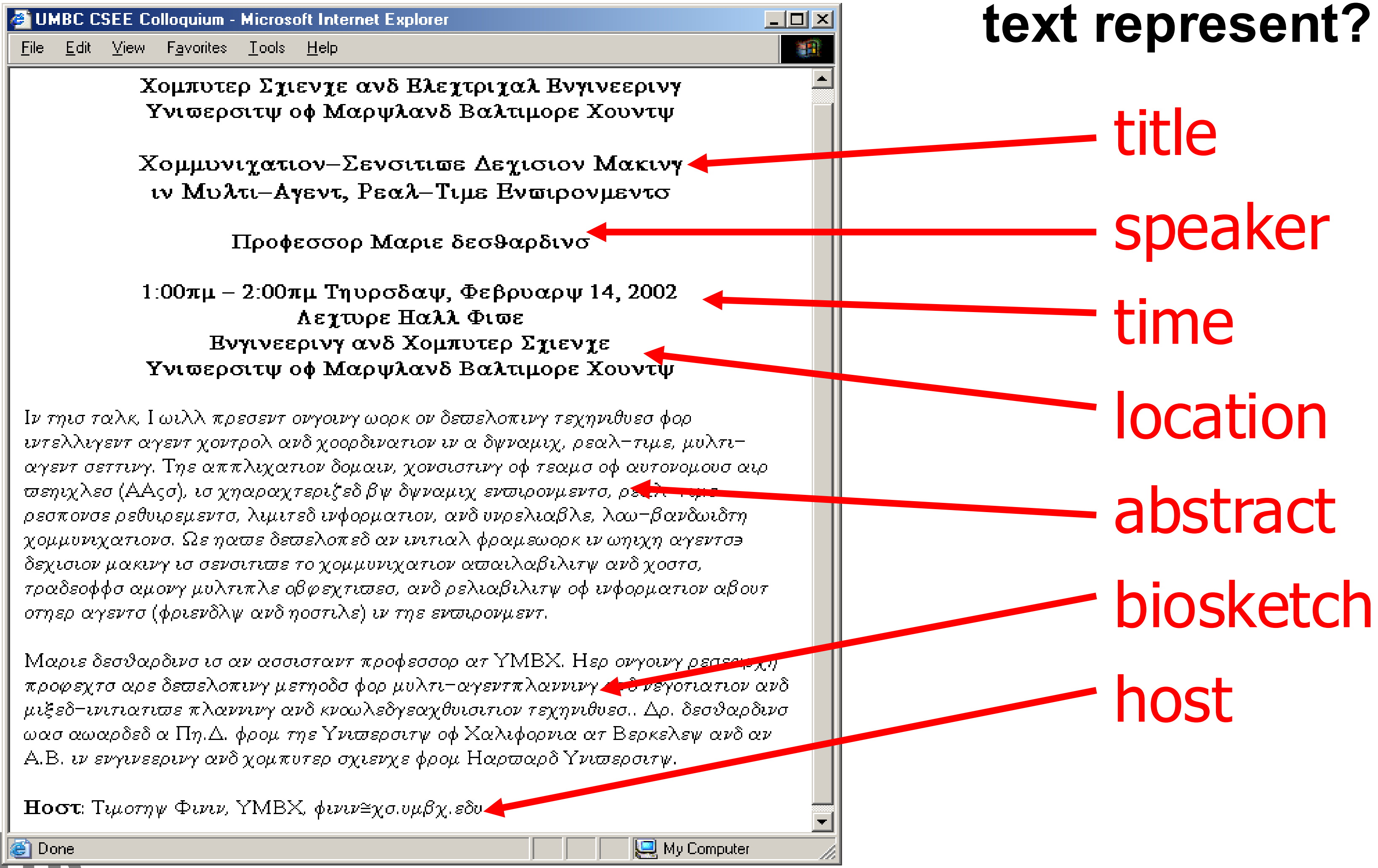
***"Webscraping"* is mostly done by hand crafted rules or rules generated by supervised learning**

Either way, the rules can break when the page structure changes.

after Frank van Harmelen and Jim Hendler

OK, so HTML is not helpful

Could we tell the machine what the different parts of the text represent?



Χομπυτερ Σχιενχε ανδ Ελεχτριχαλ Ενγινεερινγ
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ

Χομμυνιχατιον–Σενσιτιϖε Δεχισιον Μακινγ
ιν Μυλτι–Αγεντ, Ρεαλ–Τιμε Ενϖιρονμεντσ

Προφεσσορ Μαριε δεσϑαρδινσ

1:00πμ – 2:00πμ Τηυρσδαψ, Φεβρυαρψ 14, 2002
Λεχτυρε Ηαλλ Φιϖε
Ενγινεερινγ ανδ Χομπυτερ Σχιενχε
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ

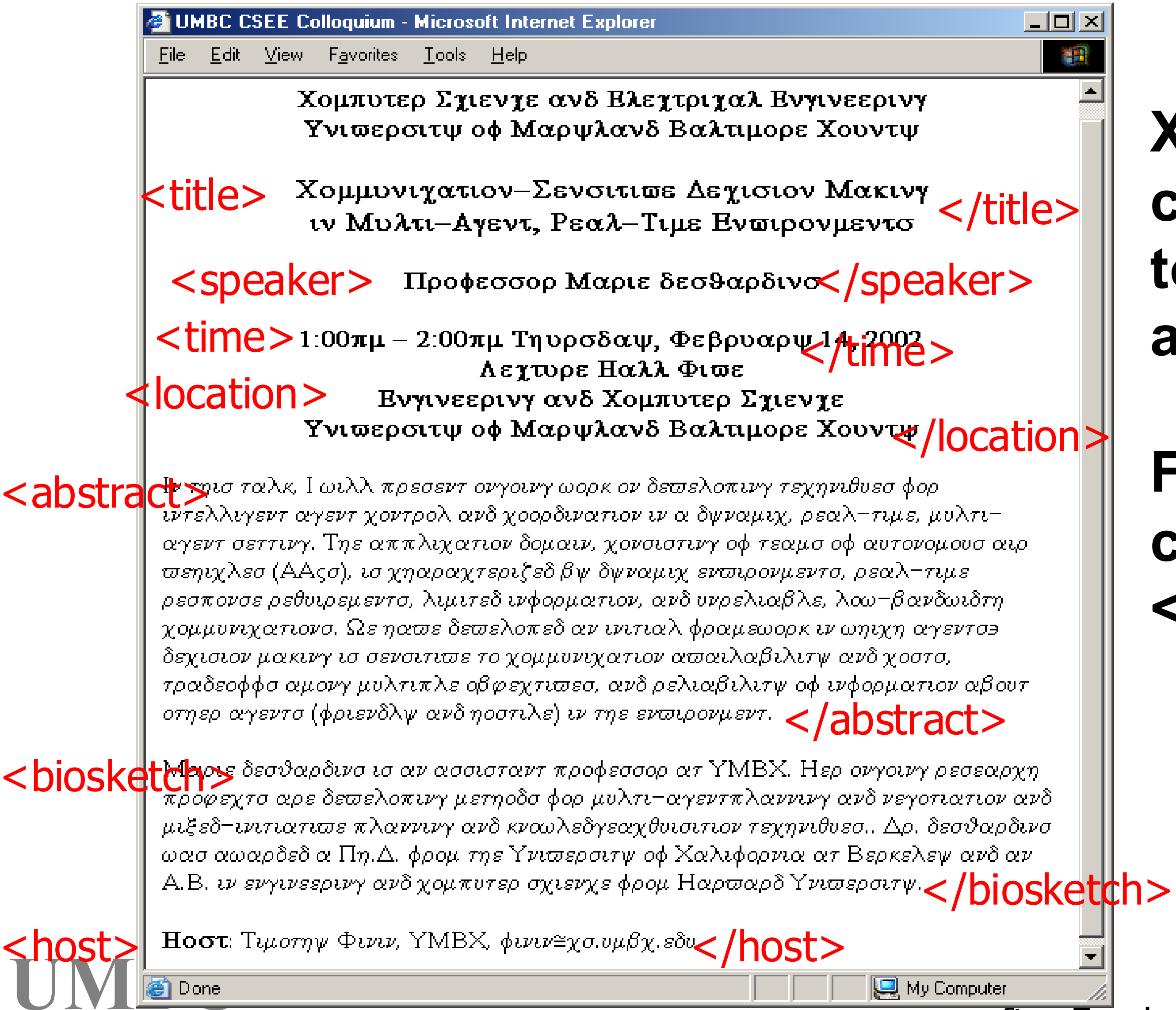
Ιν τηισ ταλκ, Ι ωιλλ πρεσεντ ονγοινγ ωορκ ον δεϖελοπινγ τεχηνιθυεσ φορ ιντελλιγεντ αγεντ χοντρολ ανδ χοορδινατιον ιν α δψναμιχ, ρεαλ–τιμε, μυλτι–αγεντ σεττινγ. Τηε αππλιχατιον δομαιν, χονσιστινγ οφ τεαμσ οφ αυτονομουσ αιρ ϖεηιχλεσ (ΑΑςσ), ισ χηαραχτεριζεδ βψ δψναμιχ ενϖιρονμεντσ, ρεαλ–τιμε ρεσπονσε ρεθυιρεμεντσ, λιμιτεδ ινφορματιον, ανδ υνρελιαβλε, λοω–βανδωιδτη χομμυνιχατιονσ. Ωε ηαϖε δεϖελοπεδ αν ινιτιαλ φραμεωορκ ιν ωηιχη αγεντσ δεχισιον μακινγ ισ σενσιτιϖε το χομμυνιχατιον αϖαιλαβιλιτψ ανδ χοστσ, τραδεοφφσ αμονγ μυλτιπλε οβϕεχτιϖεσ, ανδ ρελιαβιλιτψ οφ ινφορματιον αβουτ οτηερ αγεντσ (φριενδλψ ανδ ηοστιλε) ιν τηε ενϖιρονμεντ.

Μαριε δεσϑαρδινσ ισ αν ασσισταντ προφεσσορ ατ ΥΜΒΧ. Ηερ ονγοινγ ρεσεαρχη προϕεχτσ αρε δεϖελοπινγ μετηοδσ φορ μυλτι–αγεντπλαννινγ ανδ νεγοτιατιον ανδ μιξεδ–ινιτιατιϖε πλαννινγ ανδ κνοωλεδγεαχθυισιτιον τεχηνιθυεσ.. Δρ. δεσϑαρδινσ ωασ αωαρδεδ α Πη.Δ. φρομ τηε Υνιϖερσιτψ οφ Χαλιφορνια ατ Βερκελεψ ανδ αν Α.Β. ιν ενγινεερινγ ανδ χομπυτερ σχιενχε φρομ Ηαρϖαρδ Υνιϖερσιτψ.

Ηοστ: Τιμοτηψ Φινιν, ΥΜΒΧ, φινιν≅χσ.υμβχ.εδυ

after Frank van Harmelen and Jim Hendler

XML to the rescue?



UMBC CSEE Colloquium - Microsoft Internet Explorer

Χομπυτερ Σχιενχε ανδ Ελεχτριχαλ Ενγινεερινγ
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ

<title> Χομμυνιχατιον–Σενσιτιϖε Δεχισιον Μακινγ
ιν Μυλτι–Αγεντ, Ρεαλ–Τιμε Ενϖιρονμεντσ </title>

<speaker> Προφεσσορ Μαριε δεσϑαρδινσ</speaker>

<time>1:00πμ – 2:00πμ Τηυρσδαψ, Φεβρυαρψ 14, 2002</time>
Λεχτυρε Ηαλλ Φιϖε
<location> Ενγινεερινγ ανδ Χομπυτερ Σχιενχε
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ</location>

<abstract>Ιν τηισ ταλκ, Ι ωιλλ πρεσεντ ονγοινγ ωορκ ον δεϖελοπινγ τεχηνιθυεσ φορ ιντελλιγεντ αγεντ χοντρολ ανδ χοορδινατιον ιν α δψναμιχ, ρεαλ–τιμε, μυλτι–αγεντ σεττινγ. Τηε αππλιχατιον δομαιν, χονσιστινγ οφ τεαμσ οφ αυτονομουσ αιρ ϖεηιχλεσ (ΑΑςσ), ισ χηαραχτεριζεδ βψ δψναμιχ ενϖιρονμεντσ, ρεαλ–τιμε ρεσπονσε ρεθυιρεμεντσ, λιμιτεδ ινφορματιον, ανδ υνρελιαβλε, λοω–βανδωιδτη χομμυνιχατιονσ. Ωε ηαϖε δεϖελοπεδ αν ινιτιαλ φραμεωορκ ιν ωηιχη αγεντσ∋ δεχισιον μακινγ ισ σενσιτιϖε το χομμυνιχατιον αϖαιλαβιλιτψ ανδ χοστσ, τραδεοφφσ αμονγ μυλτιπλε οβφεχτιϖεσ, ανδ ρελιαβιλιτψ οφ ινφορματιον αβουτ οτηερ αγεντσ (φριενδλψ ανδ ηοστιλε) ιν τηε ενϖιρονμεντ. </abstract>

<biosketch>Μαριε δεσϑαρδινσ ισ αν ασσισταντ προφεσσορ ατ ΥΜΒΧ. Ηερ ονγοινγ ρεσεαρχη προφεχτσ αρε δεϖελοπινγ μετηοδσ φορ μυλτι–αγεντπλαννινγ ανδ νεγοτιατιον ανδ μιξεδ–ινιτιατιϖε πλαννινγ ανδ κνοωλεδγεαχθυισιτιον τεχηνιθυεσ.. Δρ. δεσϑαρδινσ ωασ αωαρδεδ α Πη.Δ. φρομ τηε Υνιϖερσιτψ οφ Χαλιφορνια ατ Βερκελεψ ανδ αν Α.Β. ιν ενγινεερινγ ανδ χομπυτερ σχιενχε φρομ Ηαρϖαρδ Υνιϖερσιτψ.</biosketch>

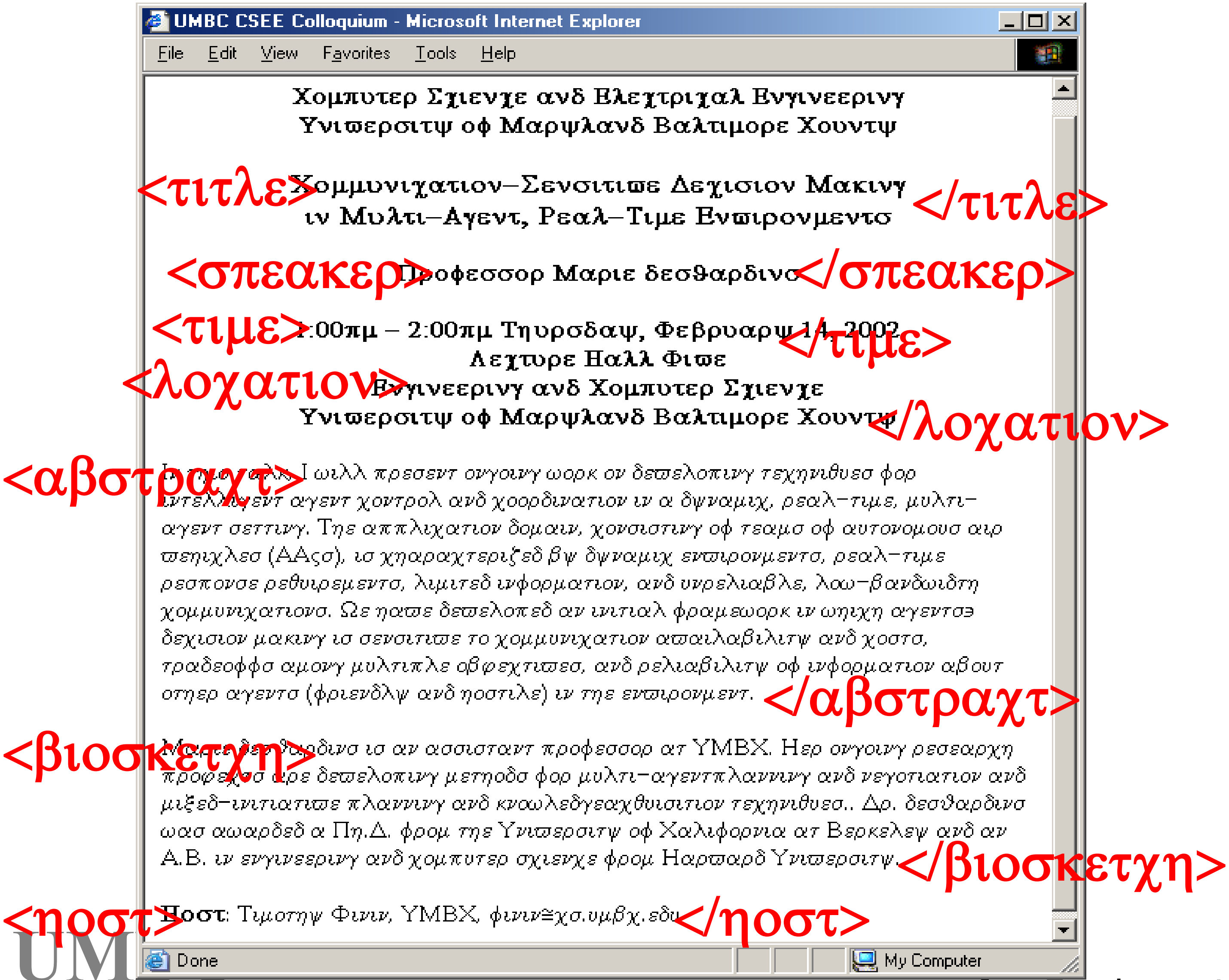
<host>**Ηοστ**: Τιμοτηψ Φινιν, ΥΜΒΧ, φινιν≅χσ.υμβχ.εδυ</host>

XML fans propose creating a XML tag set to use for each application.

For talks, we can choose <title>, <speaker>, etc.

after Frank van Harmelen and Jim Hendler

XML ≠ machine accessible meaning


UMBC CSEE Colloquium - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Done
My Computer

Χομπυτερ Σχιενχε ανδ Ελεχτριχαλ Ενγινεερινγ
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ

<τιτλε>Χομμυνιχατιον–Σενσιτιϖε Δεχισιον Μακινγ
ιν Μυλτι–Αγεντ, Ρεαλ–Τιμε Ενϖιρονμεντσ</τιτλε>

<σπεακερ>Προφεσσορ Μαριε δεσΘαρδινσ</σπεακερ>

<τιμε>1:00πμ – 2:00πμ Τηυρσδαψ, Φεβρυαρψ 14, 2002</τιμε>
Λεχτυρε Ηαλλ Φιϖε
<λοχατιον>Ενγινεερινγ ανδ Χομπυτερ Σχιενχε
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ</λοχατιον>

<αβστραχτ>Ιν τηισ ταλκ, Ι ωιλλ πρεσεντ ονγοινγ ωορκ ον δεϖελοπινγ τεχηνιθυεσ φορ ιντελλιγεντ αγεντ χοντρολ ανδ χοορδινατιον ιν α δψναμιχ, ρεαλ–τιμε, μυλτι–αγεντ σεττινγ. Τηε αππλιχατιον δομαιν, χονσιστινγ οφ τεαμσ οφ αυτονομουσ αιρ ϖεηιχλεσ (ΑΑςσ), ισ χηαραχτεριζεδ βψ δψναμιχ ενϖιρονμεντσ, ρεαλ–τιμε ρεσπονσε ρεθυιρεμεντσ, λιμιτεδ ινφορματιον, ανδ υνρελιαβλε, λοω–βανδωιδτη χομμυνιχατιονσ. Ωε ηαϖε δεϖελοπεδ αν ινιτιαλ φραμεωορκ ιν ωηιχη αγεντσ∋ δεχισιον μακινγ ισ σενσιτιϖε το χομμυνιχατιον αϖαιλαβιλιτψ ανδ χοστσ, τραδεοφφσ αμονγ μυλτιπλε οβφεχτιϖεσ, ανδ ρελιαβιλιτψ οφ ινφορματιον αβουτ οτηερ αγεντσ (φριενδλψ ανδ ηοστιλε) ιν τηε ενϖιρονμεντ.</αβστραχτ>

<βιοσκετχη>Μαριε δεσΘαρδινσ ισ αν ασσισταντ προφεσσορ ατ ΥΜΒΧ. Ηερ ονγοινγ ρεσεαρχη προφεχτσ αρε δεϖελοπινγ μετηοδσ φορ μυλτι–αγεντπλαννινγ ανδ νεγοτιατιον ανδ μιξεδ–ινιτιατιϖε πλαννινγ ανδ κνοωλεδγεαχθυισιτιον τεχηνιθυεσ.. Δρ. δεσΘαρδινσ ωασ αωαρδεδ α Πη.Δ. φρομ τηε Υνιϖερσιτψ οφ Χαλιφορνια ατ Βερκελεψ ανδ αν Α.Β. ιν ενγινεερινγ ανδ χομπυτερ σχιενχε φρομ Ηαρϖαρδ Υνιϖερσιτψ.</βιοσκετχη>

<ηοστ>**Ηοστ**: Τιμοτηψ Φινιν, ΥΜΒΧ, φινιν≅χσ.υμβχ.εδυ</ηοστ>

But, to your machine, the tags still look like this....

The tag names carry no meaning.

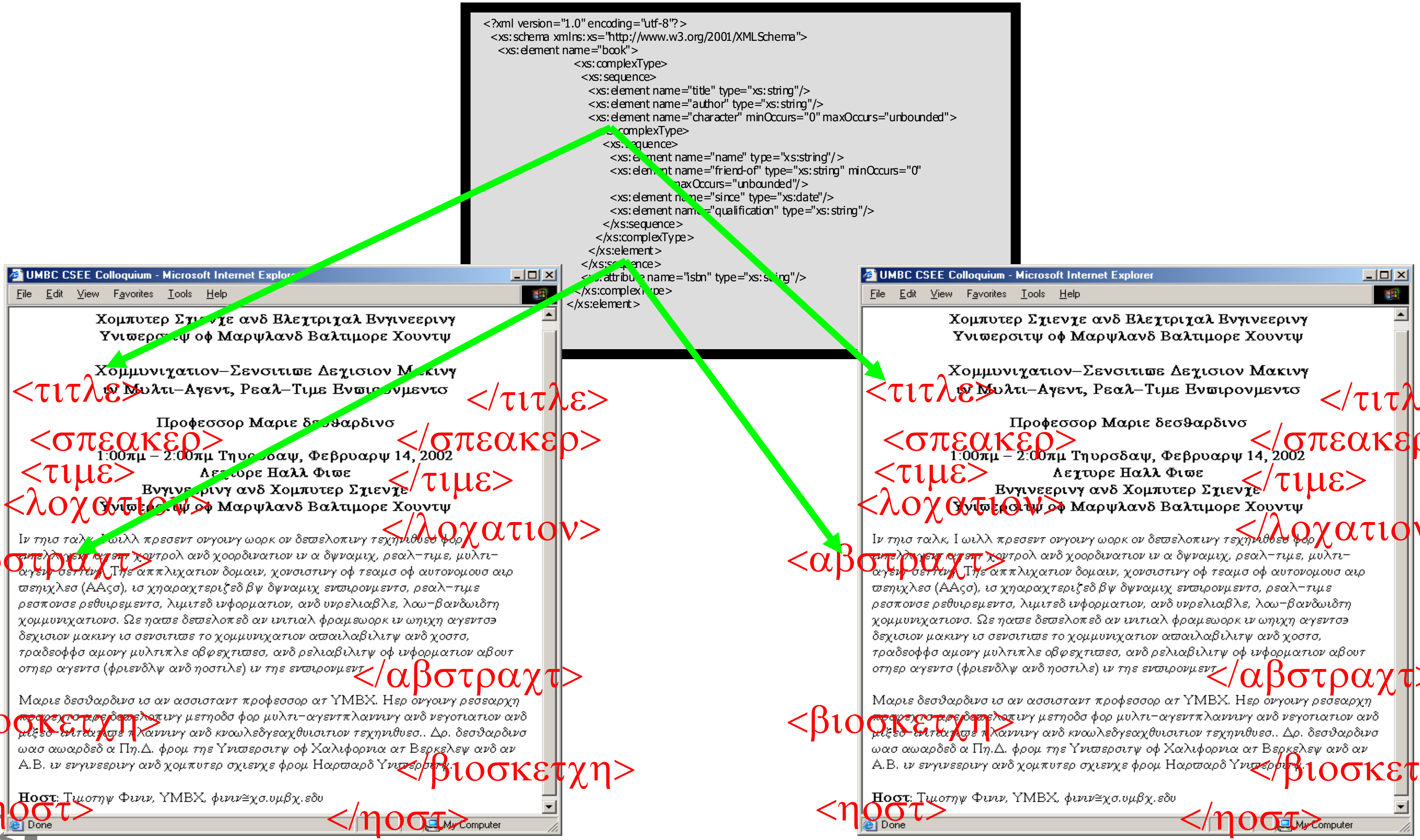
XML DTDs and Schemas have little or no semantics.

after Frank van Harmelen and Jim Hendler

XML Schema helps

XML Schemas provide a simple mechanism to define shared vocabularies.

XML Schema file



```
<?xml version="1.0" encoding="utf-8"?>
  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:element name="book">
                      <xs:complexType>
                        <xs:sequence>
                          <xs:element name="title" type="xs:string"/>
                          <xs:element name="author" type="xs:string"/>
                          <xs:element name="character" minOccurs="0" maxOccurs="unbounded">
                            <xs:complexType>
                              <xs:sequence>
                                <xs:element name="name" type="xs:string"/>
                                <xs:element name="friend-of" type="xs:string" minOccurs="0"
                                                maxOccurs="unbounded"/>
                                <xs:element name="since" type="xs:date"/>
                                <xs:element name="qualification" type="xs:string"/>
                              </xs:sequence>
                            </xs:complexType>
                          </xs:element>
                        </xs:sequence>
                        <xs:attribute name="isbn" type="xs:string"/>
                      </xs:complexType>
                    </xs:element>
```

Χομπυτερ Σχιενχε ανδ Ελεχτριχαλ Ενγινεερινγ
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ

Χομμυνιχατιον–Σενσιτιϖε Δεχισιον Μακινγ
ιν Μυλτι–Αγεντ, Ρεαλ–Τιμε Ενϖιρονμεντσ

Προφεσσορ Μαριε δεσϑαρδινσ

1:00πμ – 2:00πμ Τηυρσδαψ, Φεβρυαρψ 14, 2002
Λεχτυρε Ηαλλ Φιϖε
Ενγινεερινγ ανδ Χομπυτερ Σχιενχε
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ

Ιν τηισ ταλκ, Ι ωιλλ πρεσεντ ονγοινγ ωορκ ον δεϖελοπινγ τεχηνιθυεσ φορ ιντελλιγεντ αγεντ χοντρολ ανδ χοορδινατιον ιν α δψναμιχ, ρεαλ–τιμε, μυλτι–αγεντ σεττινγ. Τηε αππλιχατιον δομαιν, χονσιστινγ οφ τεαμσ οφ αυτονομουσ αιρ ϖεηιχλεσ (ΑΑςσ), ισ χηαραχτεριζεδ βψ δψναμιχ ενϖιρονμεντσ, ρεαλ–τιμε ρεσπονσε ρεθυιρεμεντσ, λιμιτεδ ινφορματιον, ανδ υνρελιαβλε, λοω–βανδωιδτη χομμυνιχατιονσ. Ωε ηαϖε δεϖελοπεδ αν ινιτιαλ φραμεωορκ ιν ωηιχη αγεντσ∋ δεχισιον μακινγ ισ σενσιτιϖε το χομμυνιχατιον αϖαιλαβιλιτψ ανδ χοστσ, τραδεοφφσ αμονγ μυλτιπλε οβϕεχτιϖεσ, ανδ ρελιαβιλιτψ οφ ινφορματιον αβουτ οτηερ αγεντσ (φριενδλψ ανδ ηοστιλε) ιν τηε ενϖιρονμεντ.

Μαριε δεσϑαρδινσ ισ αν ασσισταντ προφεσσορ ατ ΥΜΒΧ. Ηερ ονγοινγ ρεσεαρχη προϕεχτσ αρε δεϖελοπινγ μετηοδσ φορ μυλτι–αγεντπλαννινγ ανδ νεγοτιατιον ανδ μιξεδ–ινιτιατιϖε πλαννινγ ανδ κνοωλεδγεαχθυισιτιον τεχηνιθυεσ.. Δρ. δεσϑαρδινσ ωασ αωαρδεδ α Πη.Δ. φρομ τηε Υνιϖερσιτψ οφ Χαλιφορνια ατ Βερκελεψ ανδ αν Α.Β. ιν ενγινεερινγ ανδ χομπυτερ σχιενχε φρομ Ηαρϖαρδ Υνιϖερσιτψ.

Ηοστ: Τιμοτηψ Φινιν, ΥΜΒΧ, φινιν≅χσ.υμβχ.εδυ

Χομπυτερ Σχιενχε ανδ Ελεχτριχαλ Ενγινεερινγ
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ

Χομμυνιχατιον–Σενσιτιϖε Δεχισιον Μακινγ
ιν Μυλτι–Αγεντ, Ρεαλ–Τιμε Ενϖιρονμεντσ

Προφεσσορ Μαριε δεσϑαρδινσ

1:00πμ – 2:00πμ Τηυρσδαψ, Φεβρυαρψ 14, 2002
Λεχτυρε Ηαλλ Φιϖε
Ενγινεερινγ ανδ Χομπυτερ Σχιενχε
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ

Ιν τηισ ταλκ, Ι ωιλλ πρεσεντ ονγοινγ ωορκ ον δεϖελοπινγ τεχηνιθυεσ φορ ιντελλιγεντ αγεντ χοντρολ ανδ χοορδινατιον ιν α δψναμιχ, ρεαλ–τιμε, μυλτι–αγεντ σεττινγ. Τηε αππλιχατιον δομαιν, χονσιστινγ οφ τεαμσ οφ αυτονομουσ αιρ ϖεηιχλεσ (ΑΑςσ), ισ χηαραχτεριζεδ βψ δψναμιχ ενϖιρονμεντσ, ρεαλ–τιμε ρεσπονσε ρεθυιρεμεντσ, λιμιτεδ ινφορματιον, ανδ υνρελιαβλε, λοω–βανδωιδτη χομμυνιχατιονσ. Ωε ηαϖε δεϖελοπεδ αν ινιτιαλ φραμεωορκ ιν ωηιχη αγεντσ∋ δεχισιον μακινγ ισ σενσιτιϖε το χομμυνιχατιον αϖαιλαβιλιτψ ανδ χοστσ, τραδεοφφσ αμονγ μυλτιπλε οβϕεχτιϖεσ, ανδ ρελιαβιλιτψ οφ ινφορματιον αβουτ οτηερ αγεντσ (φριενδλψ ανδ ηοστιλε) ιν τηε ενϖιρονμεντ.

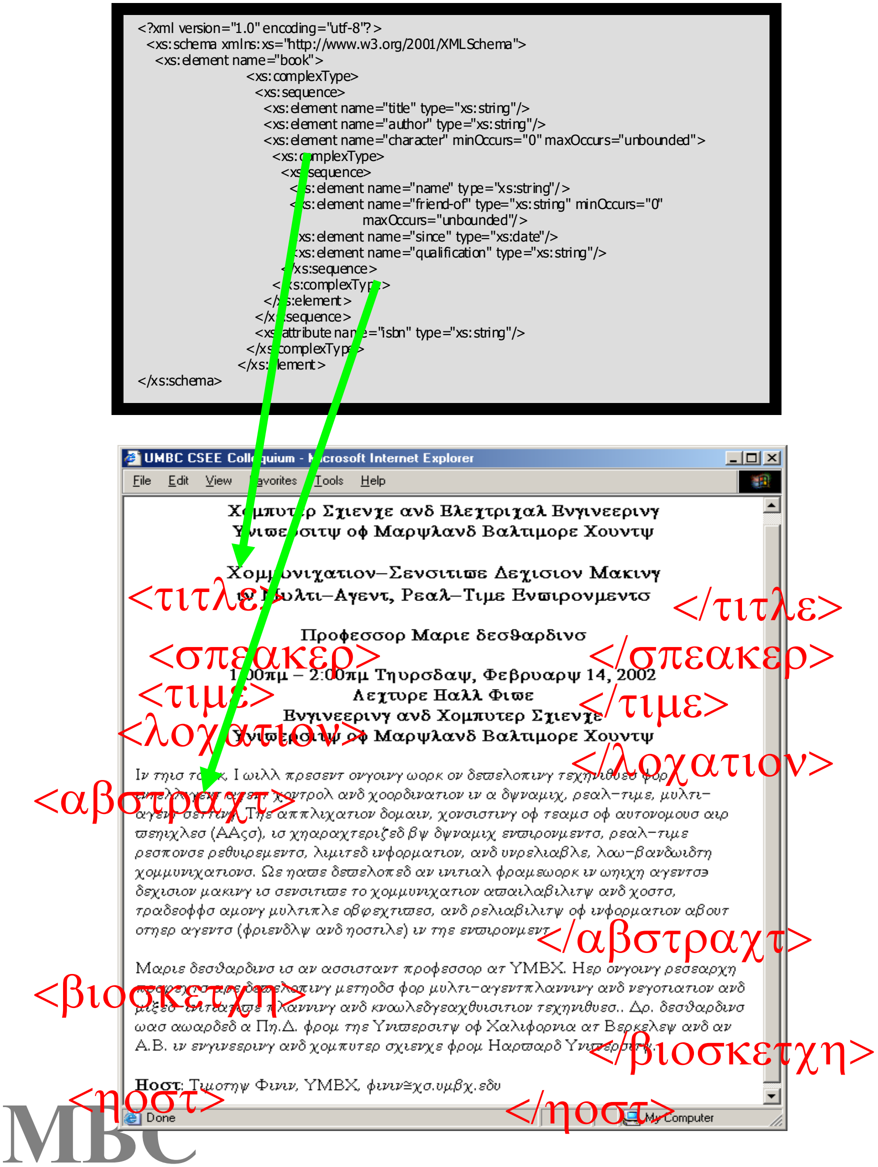
Μαριε δεσϑαρδινσ ισ αν ασσισταντ προφεσσορ ατ ΥΜΒΧ. Ηερ ονγοινγ ρεσεαρχη προϕεχτσ αρε δεϖελοπινγ μετηοδσ φορ μυλτι–αγεντπλαννινγ ανδ νεγοτιατιον ανδ μιξεδ–ινιτιατιϖε πλαννινγ ανδ κνοωλεδγεαχθυισιτιον τεχηνιθυεσ.. Δρ. δεσϑαρδινσ ωασ αωαρδεδ α Πη.Δ. φρομ τηε Υνιϖερσιτψ οφ Χαλιφορνια ατ Βερκελεψ ανδ αν Α.Β. ιν ενγινεερινγ ανδ χομπυτερ σχιενχε φρομ Ηαρϖαρδ Υνιϖερσιτψ.

Ηοστ: Τιμοτηψ Φινιν, ΥΜΒΧ, φινιν≅χσ.υμβχ.εδυ

after Frank van Harmelen and Jim Hendler

But there are many schemas

XML Schema file 1



```
<?xml version="1.0" encoding="utf-8"?>
  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:element name="book">
                    <xs:complexType>
                      <xs:sequence>
                        <xs:element name="title" type="xs:string"/>
                        <xs:element name="author" type="xs:string"/>
                        <xs:element name="character" minOccurs="0" maxOccurs="unbounded">
                          <xs:complexType>
                            <xs:sequence>
                              <xs:element name="name" type="xs:string"/>
                              <xs:element name="friend-of" type="xs:string" minOccurs="0"
                                          maxOccurs="unbounded"/>
                              <xs:element name="since" type="xs:date"/>
                              <xs:element name="qualification" type="xs:string"/>
                            </xs:sequence>
                          </xs:complexType>
                        </xs:element>
                      </xs:sequence>
                      <xs:attribute name="isbn" type="xs:string"/>
                    </xs:complexType>
                  </xs:element>
</xs:schema>
```

Χομπυτερ Σχιενχε ανδ Ελεχτριχαλ Ενγινεερινγ
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ

Χομμυνιχατιον–Σενσιτιϖε Δεχισιον Μακινγ
ιν Μυλτι–Αγεντ, Ρεαλ–Τιμε Ενϖιρονμεντσ

Προφεσσορ Μαριε δεσΘαρδινσ

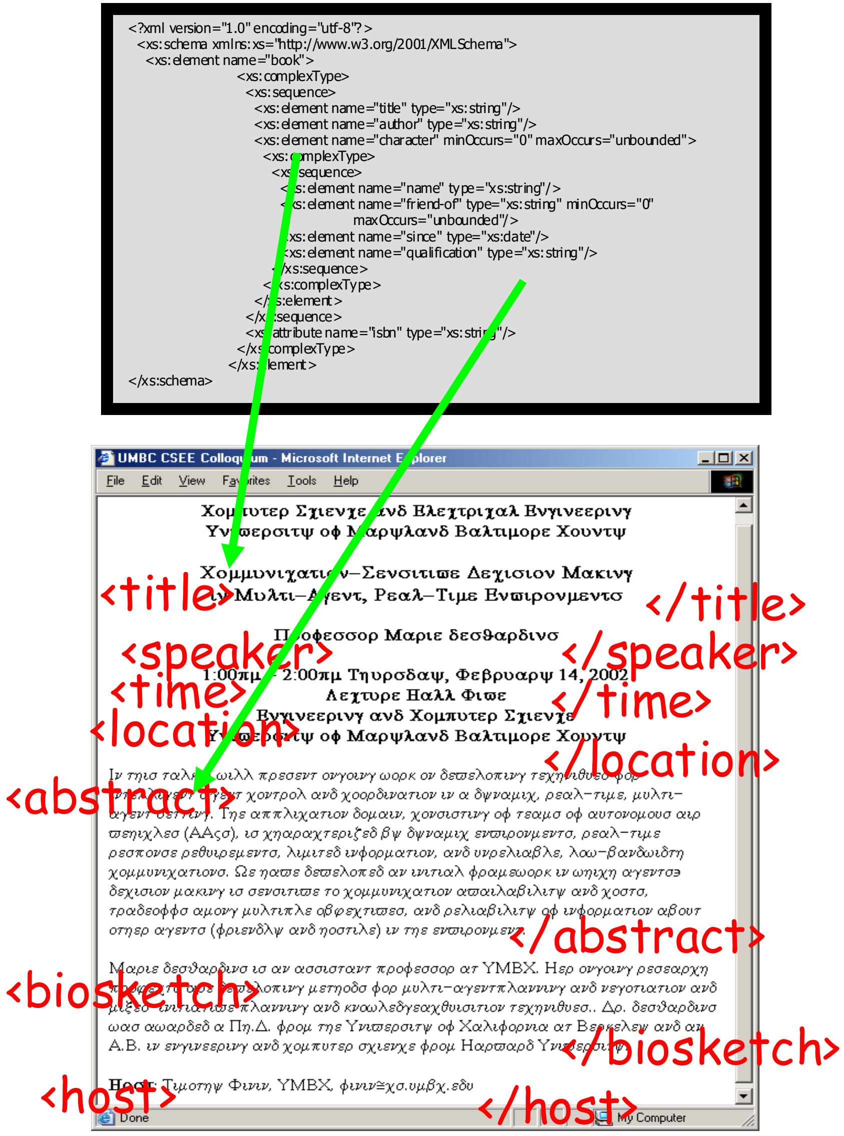
1:00πμ – 2:00πμ Τηυρσδαψ, Φεβρυαρψ 14, 2002
Λεχτυρε Ηαλλ Φιϖε
Ενγινεερινγ ανδ Χομπυτερ Σχιενχε
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ

Ιν τηισ ταλκ, Ι ωιλλ πρεσεντ ονγοινγ ωορκ ον δεϖελοπινγ τεχηνιθυεσ φορ ιντελλιγεντ αγεντ χοντρολ ανδ χοορδινατιον ιν α δψναμιχ, ρεαλ–τιμε, μυλτι–αγεντ σεττινγ. Τηε αππλιχατιον δομαιν, χονσιστινγ οφ τεαμσ οφ αυτονομουσ αιρ ϖεηιχλεσ (ΑΑςσ), ισ χηαραχτεριζεδ βψ δψναμιχ ενϖιρονμεντσ, ρεαλ–τιμε ρεσπονσε ρεθυιρεμεντσ, λιμιτεδ ινφορματιον, ανδ υνρελιαβλε, λοω–βανδωιδτη χομμυνιχατιονσ. Ωε ηαϖε δεϖελοπεδ αν ινιτιαλ φραμεωορκ ιν ωηιχη αγεντσ δεχισιον μακινγ ισ σενσιτιϖε το χομμυνιχατιον αϖαιλαβιλιτψ ανδ χοστσ, τραδεοφφσ αμονγ μυλτιπλε οβϕεχτιϖεσ, ανδ ρελιαβιλιτψ οφ ινφορματιον αβουτ οτηερ αγεντσ (φριενδλψ ανδ ηοστιλε) ιν τηε ενϖιρονμεντ.

Μαριε δεσΘαρδινσ ισ αν ασσισταντ προφεσσορ ατ ΥΜΒΧ. Ηερ ονγοινγ ρεσεαρχη περσπεχτιϖε ηασ δεϖελοπινγ μετηοδσ φορ μυλτι–αγεντπλαννινγ ανδ νεγοτιατιον ανδ μιξεδ–ινιτιατιϖε πλαννινγ ανδ κνοωλεδγεαχθυισιτιον τεχηνιθυεσ.. Δρ. δεσΘαρδινσ ωασ αωαρδεδ α Πη.Δ. φρομ τηε Υνιϖερσιτψ οφ Χαλιφορνια ατ Βερκελεψ ανδ αν Α.Β. ιν ενγινεερινγ ανδ χομπυτερ σχιενχε φρομ Ηαρϖαρδ Υνιϖερσιτψ.

Ηοστ: Τιμοτηψ Φινιν, ΥΜΒΧ, φινιν≅χσ.υμβχ.εδυ

XML Schema file 42



```
<?xml version="1.0" encoding="utf-8"?>
  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:element name="book">
                    <xs:complexType>
                      <xs:sequence>
                        <xs:element name="title" type="xs:string"/>
                        <xs:element name="author" type="xs:string"/>
                        <xs:element name="character" minOccurs="0" maxOccurs="unbounded">
                          <xs:complexType>
                            <xs:sequence>
                              <xs:element name="name" type="xs:string"/>
                              <xs:element name="friend-of" type="xs:string" minOccurs="0"
                                          maxOccurs="unbounded"/>
                              <xs:element name="since" type="xs:date"/>
                              <xs:element name="qualification" type="xs:string"/>
                            </xs:sequence>
                          </xs:complexType>
                        </xs:element>
                      </xs:sequence>
                      <xs:attribute name="isbn" type="xs:string"/>
                    </xs:complexType>
                  </xs:element>
</xs:schema>
```

Χομπυτερ Σχιενχε ανδ Ελεχτριχαλ Ενγινεερινγ
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ

Χομμυνιχατιον–Σενσιτιϖε Δεχισιον Μακινγ
ιν Μυλτι–Αγεντ, Ρεαλ–Τιμε Ενϖιρονμεντσ

Προφεσσορ Μαριε δεσΘαρδινσ

1:00πμ – 2:00πμ Τηυρσδαψ, Φεβρυαρψ 14, 2002
Λεχτυρε Ηαλλ Φιϖε
Ενγινεερινγ ανδ Χομπυτερ Σχιενχε
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ

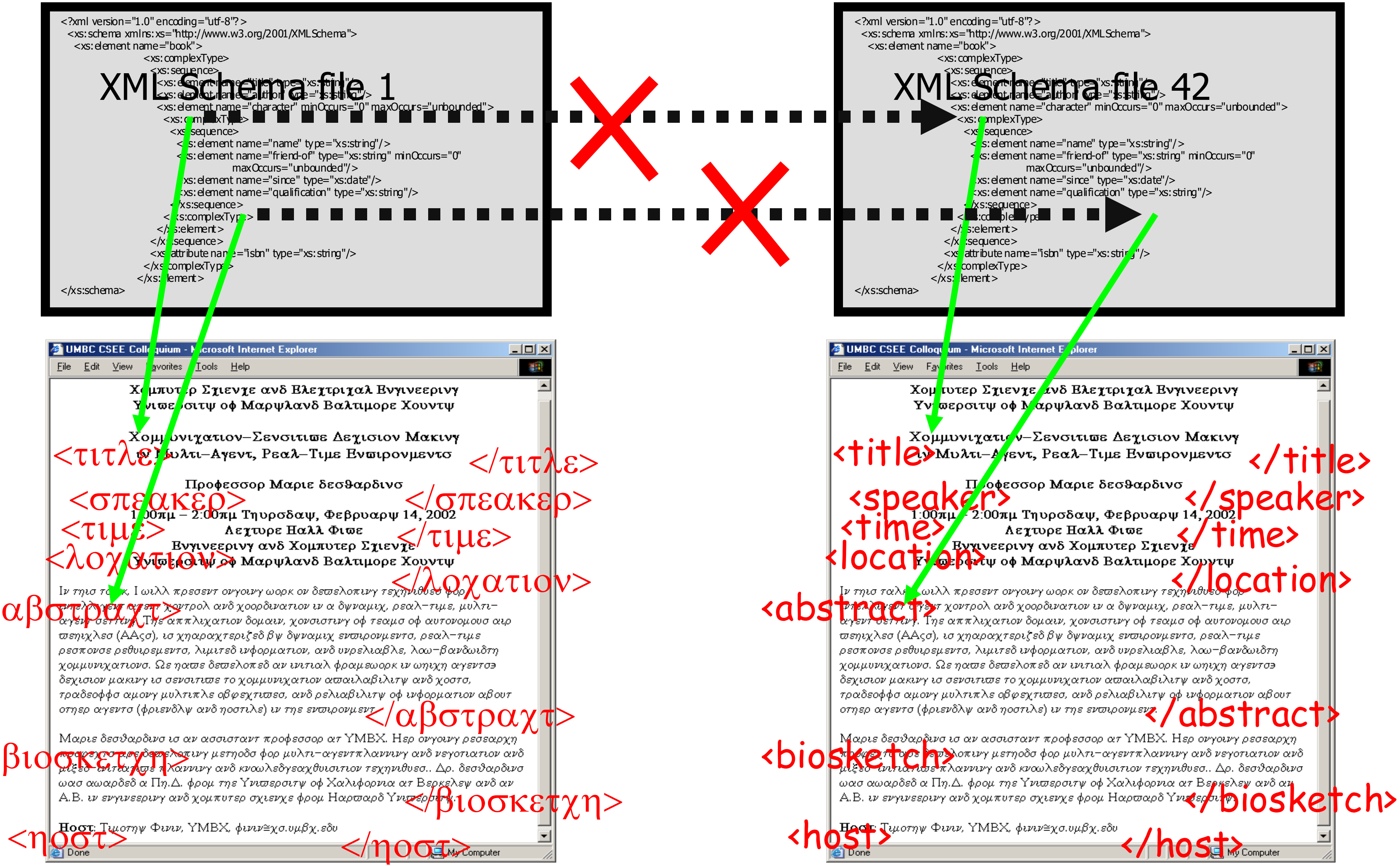
Ιν τηισ ταλκ, Ι ωιλλ πρεσεντ ονγοινγ ωορκ ον δεϖελοπινγ τεχηνιθυεσ φορ ιντελλιγεντ αγεντ χοντρολ ανδ χοορδινατιον ιν α δψναμιχ, ρεαλ–τιμε, μυλτι–αγεντ σεττινγ. Τηε αππλιχατιον δομαιν, χονσιστινγ οφ τεαμσ οφ αυτονομουσ αιρ ϖεηιχλεσ (ΑΑςσ), ισ χηαραχτεριζεδ βψ δψναμιχ ενϖιρονμεντσ, ρεαλ–τιμε ρεσπονσε ρεθυιρεμεντσ, λιμιτεδ ινφορματιον, ανδ υνρελιαβλε, λοω–βανδωιδτη χομμυνιχατιονσ. Ωε ηαϖε δεϖελοπεδ αν ινιτιαλ φραμεωορκ ιν ωηιχη αγεντσ δεχισιον μακινγ ισ σενσιτιϖε το χομμυνιχατιον αϖαιλαβιλιτψ ανδ χοστσ, τραδεοφφσ αμονγ μυλτιπλε οβϕεχτιϖεσ, ανδ ρελιαβιλιτψ οφ ινφορματιον αβουτ οτηερ αγεντσ (φριενδλψ ανδ ηοστιλε) ιν τηε ενϖιρονμεντ.

Μαριε δεσΘαρδινσ ισ αν ασσισταντ προφεσσορ ατ ΥΜΒΧ. Ηερ ονγοινγ ρεσεαρχη περσπεχτιϖε ηασ δεϖελοπινγ μετηοδσ φορ μυλτι–αγεντπλαννινγ ανδ νεγοτιατιον ανδ μιξεδ–ινιτιατιϖε πλαννινγ ανδ κνοωλεδγεαχθυισιτιον τεχηνιθυεσ.. Δρ. δεσΘαρδινσ ωασ αωαρδεδ α Πη.Δ. φρομ τηε Υνιϖερσιτψ οφ Χαλιφορνια ατ Βερκελεψ ανδ αν Α.Β. ιν ενγινεερινγ ανδ χομπυτερ σχιενχε φρομ Ηαρϖαρδ Υνιϖερσιτψ.

Ηοστ: Τιμοτηψ Φινιν, ΥΜΒΧ, φινιν≅χσ.υμβχ.εδυ

after Frank van Harmelen and Jim Hendler

There's no way to relate schema



```
<?xml version="1.0" encoding="utf-8"?>
 <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="book">
           <xs:complexType>
            <xs:sequence>
             <xs:element name="title" type="xs:string"/>
             <xs:element name="author" type="xs:string"/>
             <xs:element name="character" minOccurs="0" maxOccurs="unbounded">
              <xs:complexType>
               <xs:sequence>
                <xs:element name="name" type="xs:string"/>
                <xs:element name="friend-of" type="xs:string" minOccurs="0"
                         maxOccurs="unbounded"/>
                <xs:element name="since" type="xs:date"/>
                <xs:element name="qualification" type="xs:string"/>
               </xs:sequence>
              </xs:complexType>
             </xs:element>
            </xs:sequence>
            <xs:attribute name="isbn" type="xs:string"/>
           </xs:complexType>
          </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
 <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="book">
           <xs:complexType>
            <xs:sequence>
             <xs:element name="title" type="xs:string"/>
             <xs:element name="author" type="xs:string"/>
             <xs:element name="character" minOccurs="0" maxOccurs="unbounded">
              <xs:complexType>
               <xs:sequence>
                <xs:element name="name" type="xs:string"/>
                <xs:element name="friend-of" type="xs:string" minOccurs="0"
                         maxOccurs="unbounded"/>
                <xs:element name="since" type="xs:date"/>
                <xs:element name="qualification" type="xs:string"/>
               </xs:sequence>
              </xs:complexType>
             </xs:element>
            </xs:sequence>
            <xs:attribute name="isbn" type="xs:string"/>
           </xs:complexType>
          </xs:element>
</xs:schema>
```

Χομπυτερ Σχιενχε ανδ Ελεχτριχαλ Ενγινεερινγ
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ

Χομμυνιχατιον–Σενσιτιϖε Δεχισιον Μακινγ
ιν Μυλτι–Αγεντ, Ρεαλ–Τιμε Ενϖιρονμεντσ

Προφεσσορ Μαριε δεσϑαρδινσ

1:00πμ – 2:00πμ Τηυρσδαψ, Φεβρυαρψ 14, 2002
Λεχτυρε Ηαλλ Φιϖε
Ενγινεερινγ ανδ Χομπυτερ Σχιενχε
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ

Ιν τηισ ταλκ, Ι ωιλλ πρεσεντ ονγοινγ ωορκ ον δεϖελοπινγ τεχηνιθυεσ φορ ιντελλιγεντ αγεντ χοντρολ ανδ χοορδινατιον ιν α δψναμιχ, ρεαλ–τιμε, μυλτι–αγεντ σεττινγ. Τηε αππλιχατιον δομαιν, χονσιστινγ οφ τεαμσ οφ αυτονομουσ αιρ ϖεηιχλεσ (ΑΑςσ), ισ χηαραχτεριζεδ βψ δψναμιχ ενϖιρονμεντσ, ρεαλ–τιμε ρεσπονσε ρεθυιρεμεντσ, λιμιτεδ ινφορματιον, ανδ υνρελιαβλε, λοω–βανδωιδτη χομμυνιχατιονσ. Ωε ηαϖε δεϖελοπεδ αν ινιτιαλ φραμεωορκ ιν ωηιχη αγεντσ∋ δεχισιον μακινγ ισ σενσιτιϖε το χομμυνιχατιον αϖαιλαβιλιτψ ανδ χοστσ, τραδεοφφσ αμονγ μυλτιπλε οβϕεχτιϖεσ, ανδ ρελιαβιλιτψ οφ ινφορματιον αβουτ οτηερ αγεντσ (φριενδλψ ανδ ηοστιλε) ιν τηε ενϖιρονμεντ.

Μαριε δεσϑαρδινσ ισ αν ασσισταντ προφεσσορ ατ ΥΜΒΧ. Ηερ ονγοινγ ρεσεαρχη προϕεχτσ αρε δεϖελοπινγ μετηοδσ φορ μυλτι–αγεντπλαννινγ ανδ νεγοτιατιον ανδ μιξεδ–ινιτιατιϖε πλαννινγ ανδ κνοωλεδγεαχθυισιτιον τεχηνιθυεσ.. Δρ. δεσϑαρδινσ ωασ αωαρδεδ α Πη.Δ. φρομ τηε Υνιϖερσιτψ οφ Χαλιφορνια ατ Βερκελεψ ανδ αν Α.Β. ιν ενγινεερινγ ανδ χομπυτερ σχιενχε φρομ Ηαρϖαρδ Υνιϖερσιτψ.

Ηοστ: *Τιμοτηψ Φινιν, ΥΜΒΧ, φινιν≅χσ.υμβχ.εδυ*

Χομπυτερ Σχιενχε ανδ Ελεχτριχαλ Ενγινεερινγ
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ

Χομμυνιχατιον–Σενσιτιϖε Δεχισιον Μακινγ
ιν Μυλτι–Αγεντ, Ρεαλ–Τιμε Ενϖιρονμεντσ

Προφεσσορ Μαριε δεσϑαρδινσ

1:00πμ – 2:00πμ Τηυρσδαψ, Φεβρυαρψ 14, 2002
Λεχτυρε Ηαλλ Φιϖε
Ενγινεερινγ ανδ Χομπυτερ Σχιενχε
Υνιϖερσιτψ οφ Μαρψλανδ Βαλτιμορε Χουντψ

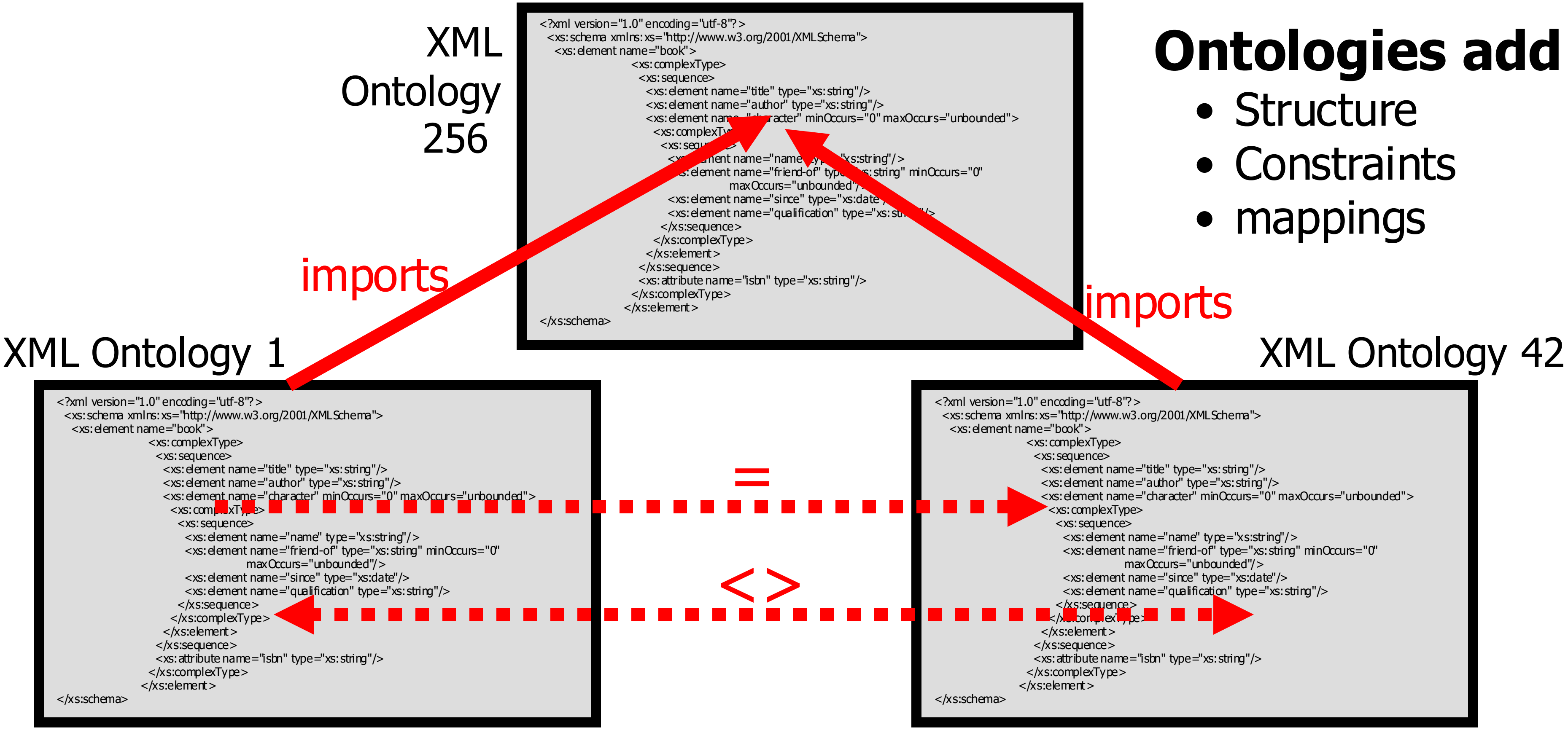
Ιν τηισ ταλκ, Ι ωιλλ πρεσεντ ονγοινγ ωορκ ον δεϖελοπινγ τεχηνιθυεσ φορ ιντελλιγεντ αγεντ χοντρολ ανδ χοορδινατιον ιν α δψναμιχ, ρεαλ–τιμε, μυλτι–αγεντ σεττινγ. Τηε αππλιχατιον δομαιν, χονσιστινγ οφ τεαμσ οφ αυτονομουσ αιρ ϖεηιχλεσ (ΑΑςσ), ισ χηαραχτεριζεδ βψ δψναμιχ ενϖιρονμεντσ, ρεαλ–τιμε ρεσπονσε ρεθυιρεμεντσ, λιμιτεδ ινφορματιον, ανδ υνρελιαβλε, λοω–βανδωιδτη χομμυνιχατιονσ. Ωε ηαϖε δεϖελοπεδ αν ινιτιαλ φραμεωορκ ιν ωηιχη αγεντσ∋ δεχισιον μακινγ ισ σενσιτιϖε το χομμυνιχατιον αϖαιλαβιλιτψ ανδ χοστσ, τραδεοφφσ αμονγ μυλτιπλε οβϕεχτιϖεσ, ανδ ρελιαβιλιτψ οφ ινφορματιον αβουτ οτηερ αγεντσ (φριενδλψ ανδ ηοστιλε) ιν τηε ενϖιρονμεντ.

Μαριε δεσϑαρδινσ ισ αν ασσισταντ προφεσσορ ατ ΥΜΒΧ. Ηερ ονγοινγ ρεσεαρχη προϕεχτσ αρε δεϖελοπινγ μετηοδσ φορ μυλτι–αγεντπλαννινγ ανδ νεγοτιατιον ανδ μιξεδ–ινιτιατιϖε πλαννινγ ανδ κνοωλεδγεαχθυισιτιον τεχηνιθυεσ.. Δρ. δεσϑαρδινσ ωασ αωαρδεδ α Πη.Δ. φρομ τηε Υνιϖερσιτψ οφ Χαλιφορνια ατ Βερκελεψ ανδ αν Α.Β. ιν ενγινεερινγ ανδ χομπυτερ σχιενχε φρομ Ηαρϖαρδ Υνιϖερσιτψ.

Ηοστ: *Τιμοτηψ Φινιν, ΥΜΒΧ, φινιν≅χσ.υμβχ.εδυ*

Either manually or automatically.
XML Schema is weak on semantics.

An Ontology level is needed



XML
Ontology
256

```
<?xml version="1.0" encoding="utf-8"?>
 <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="book">
            <xs:complexType>
             <xs:sequence>
              <xs:element name="title" type="xs:string"/>
              <xs:element name="author" type="xs:string"/>
              <xs:element name="character" minOccurs="0" maxOccurs="unbounded">
               <xs:complexType>
                <xs:sequence>
                 <xs:element name="name" type="xs:string"/>
                 <xs:element name="friend-of" type="xs:string" minOccurs="0"
                         maxOccurs="unbounded"/>
                 <xs:element name="since" type="xs:date"/>
                 <xs:element name="qualification" type="xs:string"/>
                </xs:sequence>
               </xs:complexType>
              </xs:element>
             </xs:sequence>
             <xs:attribute name="isbn" type="xs:string"/>
            </xs:complexType>
           </xs:element>
</xs:schema>
```

XML Ontology 1

```
<?xml version="1.0" encoding="utf-8"?>
 <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="book">
            <xs:complexType>
             <xs:sequence>
              <xs:element name="title" type="xs:string"/>
              <xs:element name="author" type="xs:string"/>
              <xs:element name="character" minOccurs="0" maxOccurs="unbounded">
               <xs:complexType>
                <xs:sequence>
                 <xs:element name="name" type="xs:string"/>
                 <xs:element name="friend-of" type="xs:string" minOccurs="0"
                         maxOccurs="unbounded"/>
                 <xs:element name="since" type="xs:date"/>
                 <xs:element name="qualification" type="xs:string"/>
                </xs:sequence>
               </xs:complexType>
              </xs:element>
             </xs:sequence>
             <xs:attribute name="isbn" type="xs:string"/>
            </xs:complexType>
           </xs:element>
</xs:schema>
```

XML Ontology 42

```
<?xml version="1.0" encoding="utf-8"?>
 <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="book">
            <xs:complexType>
             <xs:sequence>
              <xs:element name="title" type="xs:string"/>
              <xs:element name="author" type="xs:string"/>
              <xs:element name="character" minOccurs="0" maxOccurs="unbounded">
               <xs:complexType>
                <xs:sequence>
                 <xs:element name="name" type="xs:string"/>
                 <xs:element name="friend-of" type="xs:string" minOccurs="0"
                         maxOccurs="unbounded"/>
                 <xs:element name="since" type="xs:date"/>
                 <xs:element name="qualification" type="xs:string"/>
                </xs:sequence>
               </xs:complexType>
              </xs:element>
             </xs:sequence>
             <xs:attribute name="isbn" type="xs:string"/>
            </xs:complexType>
           </xs:element>
</xs:schema>
```

Ontologies add

- Structure
- Constraints
- mappings

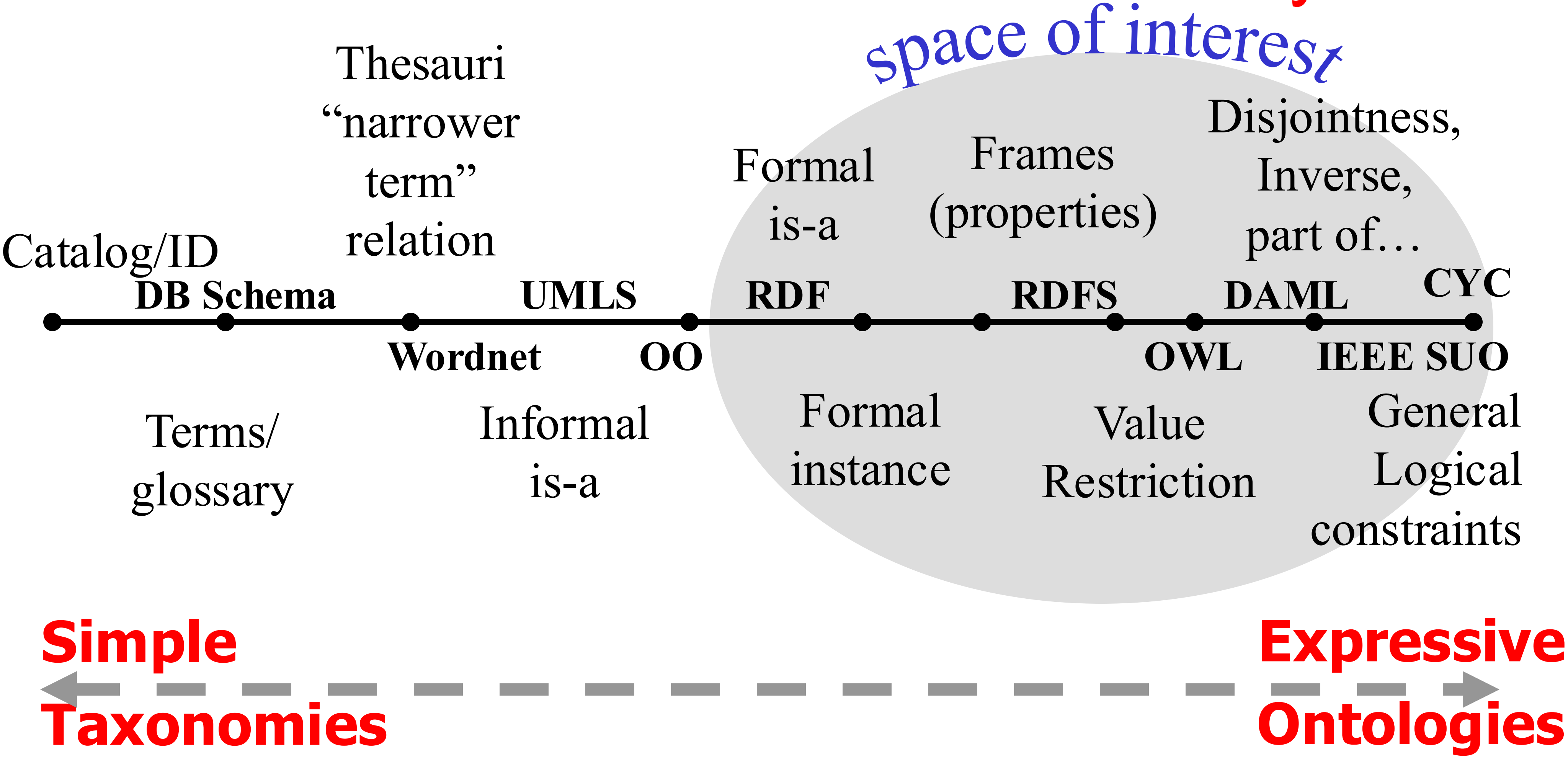
We need a way to define ontologies in XML

So we can relate them

So machines can understand (to some degree) their meaning

What kind of Ontologies?

from controlled vocabularies to Cyc



space of interest

Catalog/ID

Thesauri "narrower term" relation

Formal is-a

Frames (properties)

Disjointness, Inverse, part of…

DB Schema

UMLS

RDF

RDFS

DAML

CYC

Wordnet

OO

OWL

IEEE SUO

Terms/ glossary

Informal is-a

Formal instance

Value Restriction

General Logical constraints

Simple Taxonomies

Expressive Ontologies

After Deborah L. McGuinness (Stanford)

Dublin Core: an example of a simple ontology

- Developed by an OCLC workshop in Dublin ~95 as a metadata standard for digital library resources on web
  - 15 core attributes
  - http://dublincore.org/
- Neutral on representation
- Available as an RDF schema

http://purl.org/dc/elements/1.1/

15 DC elements

Content elements
- Coverage
- Description
- Relation
- Source
- Subject
- Title
- Type

Intellectual Property
- Contributor
- Creator
- Publisher
- Right

Instantiation
- Date
- Format
- Identifier
- Language

Cyc – an example of a complex ontology

OpenCyc.org

- Cyc is a large, general purpose ontology with reasoning engine developed since ~1983 by MCC and Cycorp
  - Cyc KB has > 100k terms.
  - Terms are axiomatized by > 1M handcrafted assertions
  - Cyc inference engine has > 500 heuristic level modules
- Goal: encode "common sense" knowledge for general applications (e.g., NLP)
- Available in DAML+OIL at http://opencyc.sourceforge.org/daml/cyc.daml

Today and tomorrow

- We are in a good position to use simple ontologies like DC today
  - This is happening (e.g., Adobe's XMP product)
- We hope to be able to make effective use ontologies like Cyc in the coming decade
  - There are skeptics …
  - It's a great research topic …
- The SW community has a roadmap and some experimental languages …

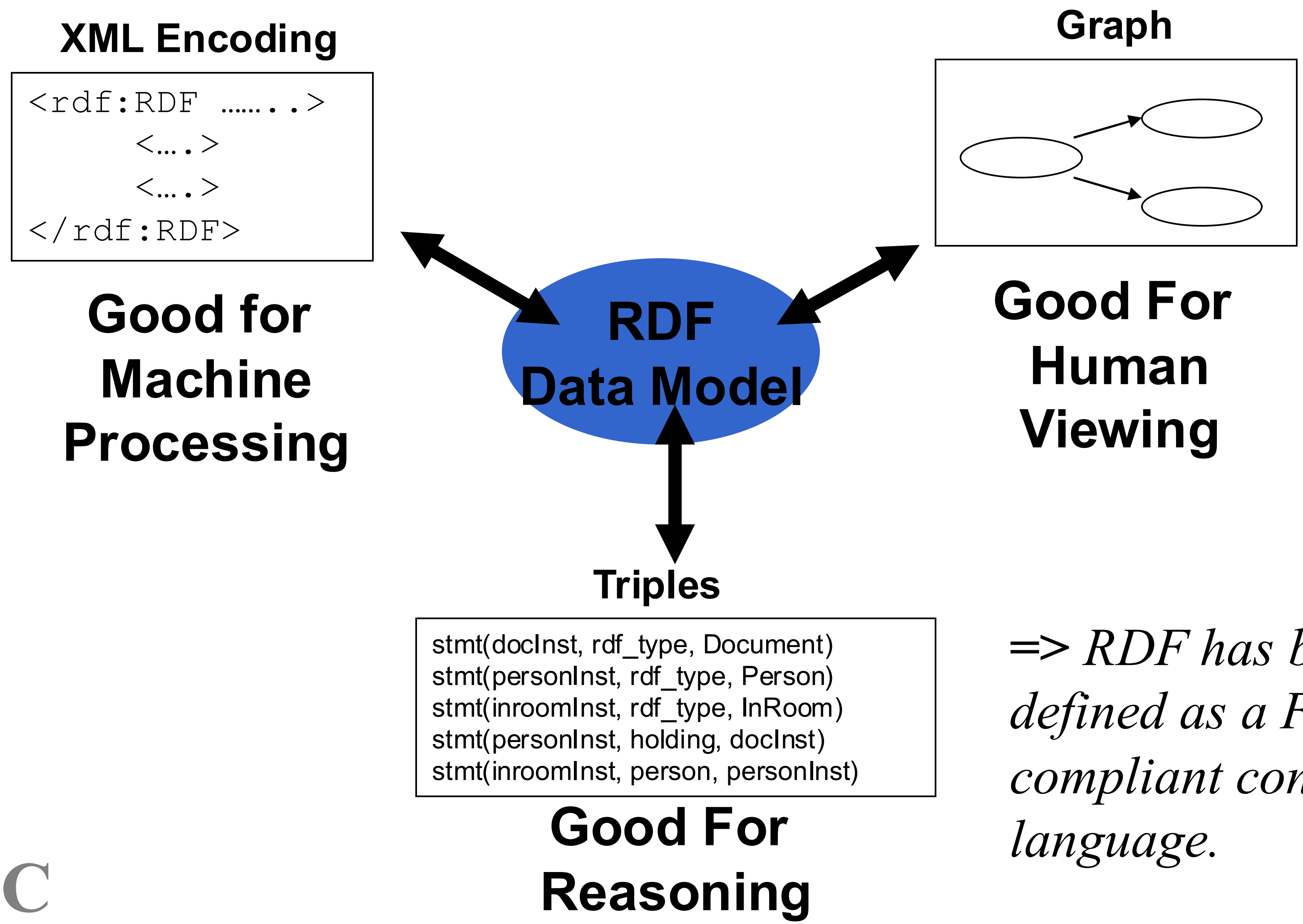
Semantic Web

Semantic Web Languages

Semantic web languages today

- Today there are, IOHO, three semantic web languages
  - **RDF** – Resource Description Framework http://www.w3.org/RDF/
  - **DAML+OIL** – Darpa Agent Markup Language http://www.daml.org/
  - **OWL** – Ontology Web Language http://www.w3.org/2001/sw/
- **Topic maps** (http://topicmaps.org) are another breed
- with more to come? ….

RDF is the first SW language



XML Encoding

```
<rdf:RDF ……..>
     <….>
     <….>
</rdf:RDF>
```

Good for Machine Processing

Graph

Good For Human Viewing

RDF Data Model

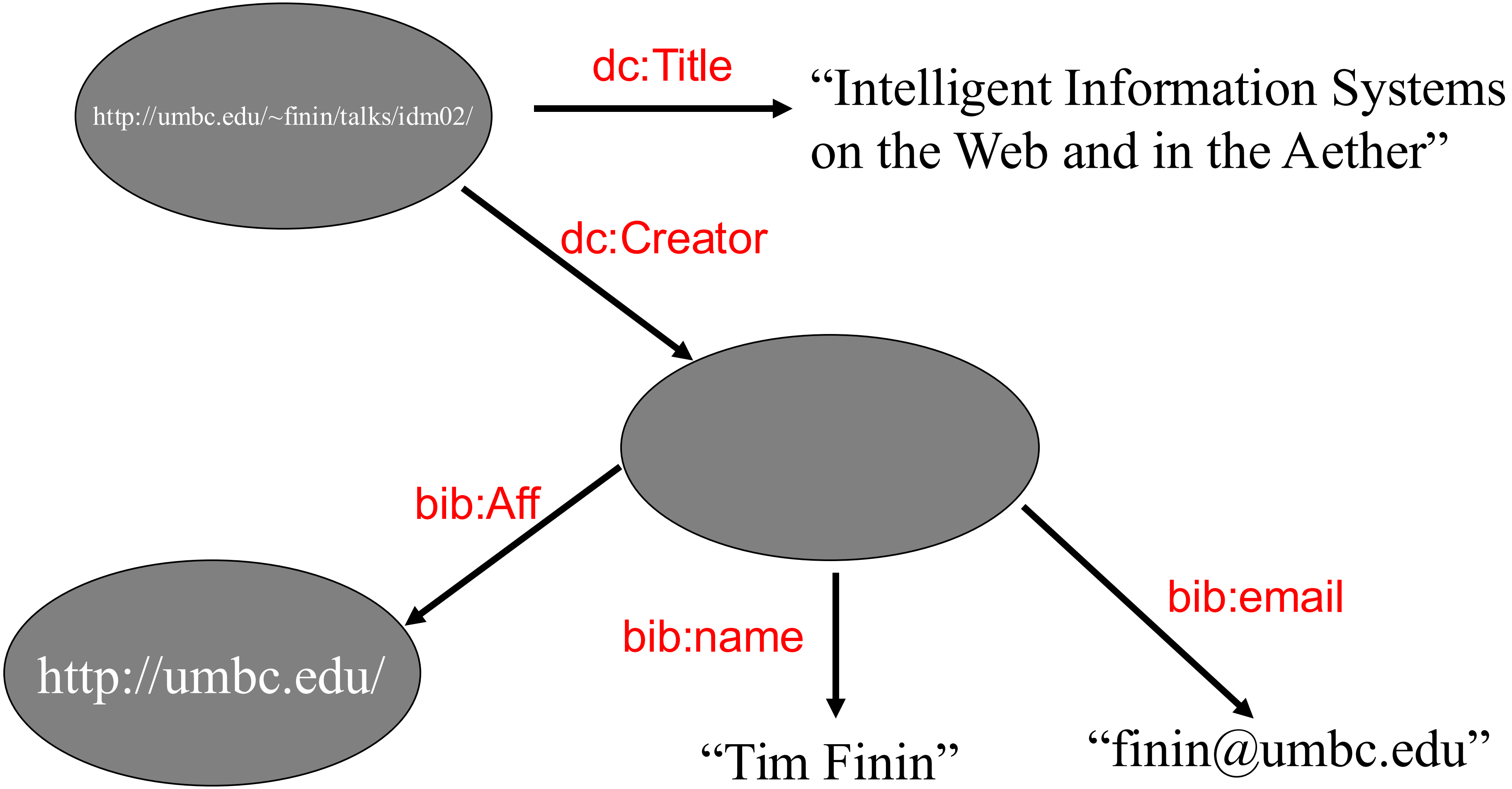
Triples

```
stmt(docInst, rdf_type, Document)
stmt(personInst, rdf_type, Person)
stmt(inroomInst, rdf_type, InRoom)
stmt(personInst, holding, docInst)
stmt(inroomInst, person, personInst)
```

Good For Reasoning

=> *RDF has been defined as a FIPA compliant content language.*

Simple RDF Example


http://umbc.edu/~finin/talks/idm02/
dc:Title
“Intelligent Information Systems on the Web and in the Aether”
dc:Creator
bib:Aff
bib:email
bib:name
http://umbc.edu/
“Tim Finin”
“finin@umbc.edu”

XML encoding for RDF

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:dc="http://purl.org/dc/elements/1.1/"
    xmlns:bib="http://daml.umbc.edu/ontologies/bib/">
<description about="http://umbc.edu/~finin/talks/idm02/">
  <dc:title>Intelligent Information Systems on the Web and in the
Aether</dc:Title>
  <dc:creator>
    <description>
      <bib:Name>Tim Finin</bib:Name>
      <bib:Email>finin@umbc.edu</bib:Email>
      <bib:Aff resource="http://umbc.edu/" />
    </description>
  </dc:Creator>
</description>
</rdf:RDF>
```

N triple representation

- RDF can be encoded as a set of triples.
- *<subject> <predicate> <object>* .

```
<http://umbc.edu/~finin/talks/idm02/> <http://purl.org/dc/elements/1.1/Title>
"Intelligent Information Systems on the Web and in the Aether" .
_:j10949 <http://daml.umbc.edu/ontologies/bib/Name> "Tim Finin" .
_:j10949 <http://daml.umbc.edu/ontologies/bib/Email> "finin@umbc.edu" .
_:j10949 <http://daml.umbc.edu/ontologies/bib/Aff> <http://umbc.edu/> .
_:j10949 <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <Description> .
<http://umbc.edu/~finin/talks/idm02/> <http://purl.org/dc/elements/1.1/Creator> _:j10949 .
<http://umbc.edu/~finin/talks/idm02/> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
    <Description> .
```

- Note the gensym for the anonymous node (_:j10949)

Triple Notes

- **RDF triples have one of two forms:**
  - <URI> <URI> <URI>
  - <URI> <URI> <quoted string>
- **Triples are also easily mapped into logic**
  - <subject> <predicate> <object>
  - <predicate>(<subject>,<object>)
  - With type(<S>,<O>) becoming <O>(<S>)
  - Example:
    - subclass(man,person)
    - sex(man,male)
    - domain(sex,animal)
    - man(adam)
    - age(adam,100)

; Note: we're not
; showing the actual
; URIs for clarity

- **Triples are easily stored and managed in a DBMS**

N3 notation for RDF

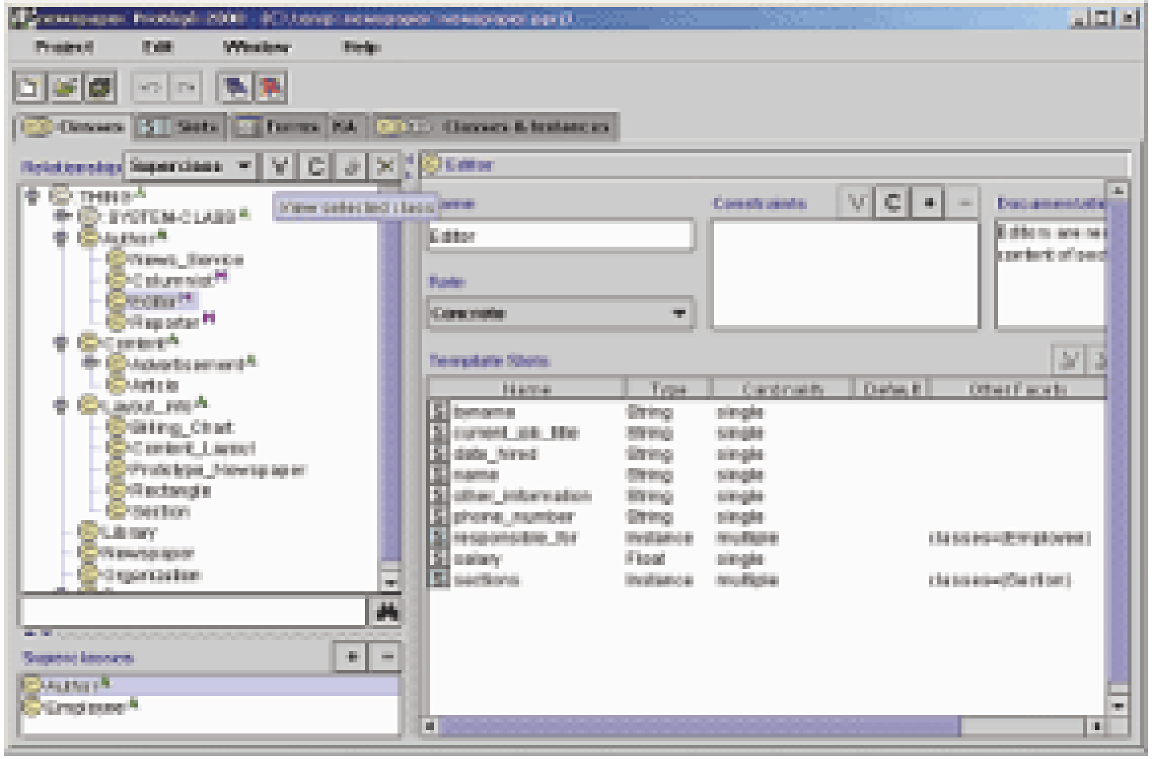
N3 is a compact notation for triples which is easier for people to read and edit

Example:

```
@prefix log: <http://www.w3.org/2000/10/swap/log#> .
:Person a rdfs:Class.
:Woman a rdfs:Class; rdfs:subClassOf :Person .
:Eve a :Woman; age “100”.
:sister a rdf:Property.
:sister rdfs:domain :Person; rdfs:range :Woman.
```

RDF Schema (RDFS)

- **RDF Schema adds taxonomie for classes & properties**
  - subClass and subProperty
- **and some metadata.**
  - domain and range constraints on properties
- **Several widely used KB tools can import and export in RDFS**



Name: Editor

Role: Concrete

Name	Type	Cardinality	Default	Other Facets
[illegible]	String	single		
current_job_title	String	single		
date_hired	String	single		
name	String	single		
other_information	String	single		
phone_number	String	single		
responsible_for	Instance	multiple		classes={Employee}
salary	Float	single		
sections	Instance	multiple		classes={Section}

Stanford Protégé KB editor

- Java, open sourced
- extensible, lots of plug-ins
- provides reasoning & server capabilities

UMBC
an Honors University in Maryland

RDFS supports simple inferences

New and Improved!
100% Better than XML!!

- An RDF ontology plus some RDF statements may imply additional RDF statements.
- This is not true of XML.
- Example:

ontology
domain(parent,person)
range(parent,person)
subproperty(mother,parent)
range(mother,woman)

instance
mother(eve,cain)

Implies:
subclass(woman,person)
parent(eve,cain)
person(eve)
person(cain)
woman(eve)

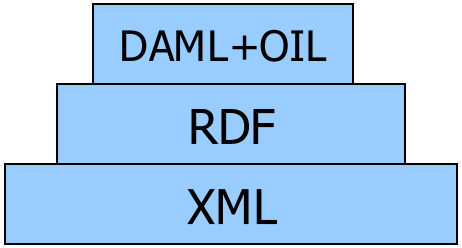
- This is **part of the data model** and not of the accessing/processing code

RDF is being already in use

- **RDF has a solid specification**
  - See the RDF model theory spec - http://www.w3.org/TR/rdf-mt/
- **RDF is being used in a number of W3C specifications**
  - CC/PP (Composite Capabilities/Preference Profiles) http://www.w3.org/Mobile/CCPP/
  - P3P (Platform for Privacy Preferences Project) http://www.w3.org/P3P/
- **And in other web standards**
  - RSS 1.0 (RDF Site Summary), RDF calendar (~ iCalendar in RDF), Dublin Core, …
- **And in other systems**
  - Netscape's Mozilla web browser
  - Open directory (http://dmoz.org/)
  - Adobe products via XMP (eXtensible Metadata Platform)

RDF isn’t enough, but is a good foundation

- RDF lacks expressive adequacy for many tasks
  - No properties of properties (transitive, inverse etc.)
  - No equivalence, disjointness, coverings, etc.
  - No necessary and sufficient conditions
  - No rules, axioms, logical constraints
- DAML+OIL extends RDF
  - Layering makes partial knowledge available to apps which only understand RDF
  - Layering has disadvantages



DAML+OIL

RDF

XML

We’re going down a familiar road

KR trends

- 55-65: arbitrary data structures
- 65-75: semantic networks
- 75-85: simple frame systems
- 85-95: description logics
- *95-??: logic?, rules?*

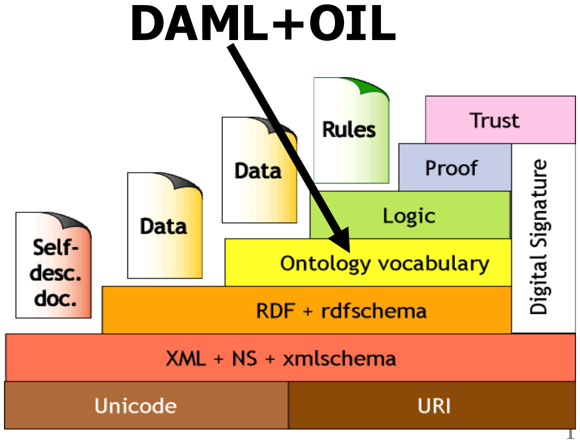
Web trends

- 95-97: XML as arbitrary structures
- 97-98: RDF
- 98-99: RDFS (schema) as a frame-like system
- 00-01: DAML+OIL
- 02-*??:* OWL…*???*…

Only much faster!

DAML+OIL as a Semantic Web Language

- DAML = Darpa Agent Markup Language
  - DARPA program with 17 projects & an integrator developing language spec, tools, applications for SW.
- OIL = Ontology Inference Layer
  - An EU effort aimed at developing a layered approach to representing knowledge on the web.
- Process
  - Joint Committee: US DAML and EU Semantic Web Technologies participants
  - DAML+OIL specs released 01/01 & 03/01
  - See http://www.daml.org/
  - Includes model theoretic and axiomatic semantics



A Simple DAML Example

```
<rdfs:Class about="#Animal"/>
<rdfs:Class about="#Plant">
    <daml:disjointFrom
          resource="#Animal"/>
</rdfs:Class>
```

- Note the mixture of rdf (*plant and animal are classes)* and DAML *(plant and animal are disjoint)*

DAML+OIL ↔ RDF

- DAML+OIL ontology is a set of RDF statements
  - DAML+OIL defines semantics for certain statements
  - Does **NOT** restrict what can be said

    Ontology can include arbitrary RDF
  - But no semantics for non-DAML+OIL statements
- Adds capabilities common to description logics:
  - cardinality constraints, defined classes (=> classification), equivalence, local restrictions, disjoint classes, etc.
- More support for ontologies
  - Ontology imports ontology
- But not (yet) variables, quantification, and general rules

DAML in One Slide

DAML is built on top of XML and RDF

It allows the definition, sharing, composition and use of ontologies

DAML is ~= a frame based knowledge representation language

It can be used to add metadata about anything which has a URI.

URIs are a W3C standard generalizing URLs

everything has URI

```
<rdf:RDF xmlns:rdf ="http://w3.org/22-rdf-syntax-ns#"
    xmlns:rdfs="http://w3.org/rdf-schema#"
    xmlns:daml="http://daml.org/daml+oil#“>
<daml:Ontology rdf:about="">
     <daml:imports rdf:resource="http://daml.org/daml+oil"/>
</daml:Ontology>
<rdfs:Class rdf:ID="Person">
  <rdfs:subClassOf rdf:resource="#Animal"/>
  <rdfs:subClassOf>
    <daml:Restriction>
      <daml:onProperty rdf:resource="#hasParent"/>
      <daml:toClass rdf:resource="#Person"/>
    </daml:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <daml:Restriction daml:cardinality="1">
      <daml:onProperty rdf:resource="#hasFather"/>
    </daml:Restriction>  </rdfs:subClassOf> </rdfs:Class>
<Person rdf:about=“http://umbc.edu/~finin/">
<rdfs:comment>Finin is a person.</rdfs:comment>
</Person>
```

DAML+OIL Extends RDF

```
<rdfs:Class about="#Animal"/>
<rdfs:Class about="#Plant">
  <daml:disjointFrom
    resource="#Animal"/>
</rdfs:Class>
```

Note mix of RDF (*plant & animal are classes)* & DAML *(plant & animal are disjoint)*

DAML adds

cardinality constraints
defined classes
equivalence
local restrictions
disjoint classes
imports
...

But not yet

rules
variables

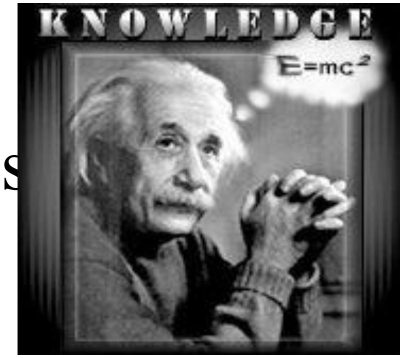
DAML+OIL Ontologies

- Research efforts are developing some interesting ontologies for services, trust, rules, time, geospatial relations, e.g., …
- DAML-S is an ontology for describing properties and capabilities of services
  - http://www.daml.org/services/
- DAML-time covers temporal concepts and properties common to any formalization of time
  - http://www.cs.rochester.edu/~ferguson/daml/
- DAML-spatial covers spatial concepts and properties
  - http://www.daml.org/listarchive/daml-spatial/date.html#29

W3C Web Ontology Working Group

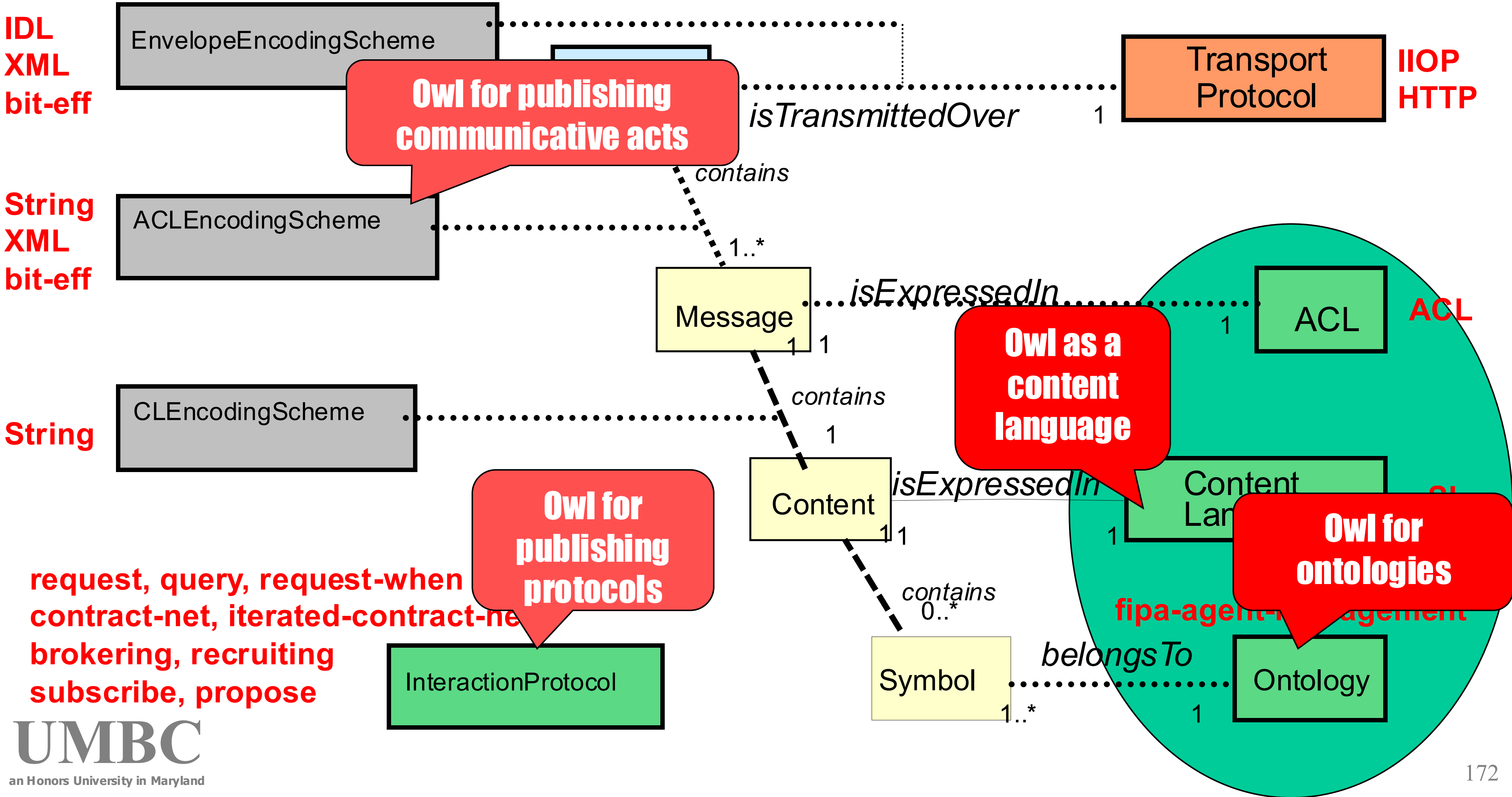
- The WOWG is developing a "Web Ontology Language": **OWL**
- WOWG has ~56 members, co-chaired by Jim Hendler (U. Maryland) & Guus Schreiber, U. Amsterdam)
- OWL =~ DAML+OIL with some renaming of properties
- Current plan is to have three compliance levels: OWL lite, OWL, OWL plus
- See http://www.w3.org/2001/sw/WebOnt/
- About to go to “last call” after which the language (v1) formally becomes a proposed standard.

KR meets the Web (and MAS)



- One way to think about the semantic web is that we are creating a knowledge repres language for the Web.
- And for multi-agent systems?
- This is more than just selecting an appropriate KR language and selecting an (XML) encoding.
- The Web (and MAS) as information systems has many significant properties.
  - Highly distributed
  - Many independent content providers
  - Dynamic and evolving
  - Inconsistent
  - Potential for malicious agents
  - …

Where OWL fits into FIPA Standards



IDL
XML
bit-eff

EnvelopeEncodingScheme

Owl for publishing communicative acts

isTransmittedOver

1

Transport Protocol

IIOP
HTTP

contains

String
XML
bit-eff

ACLEncodingScheme

1..*

Message

isExpressedIn

1

ACL

ACL

1 1

contains

Owl as a content language

String

CLEncodingScheme

1

Content

isExpressedIn

Content Lan

SL

1 1

1

Owl for ontologies

Owl for publishing protocols

request, query, request-when
contract-net, iterated-contract-ne
brokering, recruiting
subscribe, propose

contains
0..*

fipa-agent-m agement

InteractionProtocol

Symbol

belongsTo

Ontology

1..*

1

Lots of Open Issues

- How expressive should the KR language be?
- What kind of KR/reasoning system
  - F.O. logic, logic programming, fuzzy, …
- On Web Ontologies
  - One (e.g. CYC) or many (DAML)
  - If many, composable (IEEE IFF) or monolithic (IEEE SUMO)
  - Will general "upper ontologies" (e.g., IEEE SUO) be useful?
- Will industry buy in?
  - Or continue to explore ad hoc XML based solutions
- How will it be used?
  - As markup? As alternative content? Just both machines and people?
- Is it good as a content language for agents?

=> Only experimentation will yield answers.

8
Applications

Two kinds of systems

- RDF is being used to support many practical, useful applications
  - We'll touch on RSS, CCPP, P3P, …
- RDF, DAML+OIL and OWL are being experimented with in many research prototypes
  - We'll describe some research at UMBC

RSS

- Rich Site Summary or RDF Site Summary
- A lightweight multipurpose extensible metadata description and syndication format for the web
  - news and other headline syndication
  - weblog syndication
  - propagation of software update lists.

```
<channelrdf:about="http://agents.umbc.edu/awchannel.rdf">
  <title>UMBC AgentWeb</title>
  <link>http://agents.umbc.edu/</link>
  <description>UMBC AgentWeb...</description>
  <language>en-us</language>
  <copyright>copyright...</copyright>
  <managingEditor>Tim...</managingEditor>
  <pubDate> Mon ... </pubDate>
  <lastBuildDate> Mon ... </lastBuildDate>
  <webMaster>finin@umbc.edu</webmaster>
  <rating>(PICS-1.1 ... </rating>
  <image rdf:resource="http:..." />
<items><rdf:Seq>...</rdf:Seq></items>
</channel>
<image>
  <title>UMBC AgentWeb</title>
  <url>http://agents...</url>
  <link>http://agents.umbc.edu/</link>
  <width>46</width>  <height>66</height>
</image>
<item rdf:about="http:...">
 <title>UDDIe</title>
 <link>http://...</link>
 <description>the ...</item>
</rdf:RDF>
```

CCPP

- Composite Capabilities/Preference Profiles
- RDF-based W3C recommended standard for customizing web content for devices and users
- It is a Client profile data format
    - For describing device capabilities and user preferences
    - Enables adaptation of content presented to that device
- It is not a standard explaining how the profile is transferred, or what attributes must be generated or recognized

P3P

- Platform for Privacy Preferences Project
- P3P is developed by the World Wide Web Consortium.
- It allows web-sites to communicate about their privacy policies in a standard computer-readable format.
- It enables the development of tools that are built into browsers or separate applications that summarize privacy policies, compare privacy policies with user preferences, alert and advise users.
- It does not require web-sites to change their server software.
- P3P support is in IE, Netscape

Some UMBC research systems

(1) Semantic web and agents (ITTalks)

(2) Ontology mapping

(3) Cooperating personal agents

(4) Learning markup

(5) Information retrieval on the SW

(6) Modeling trust policies

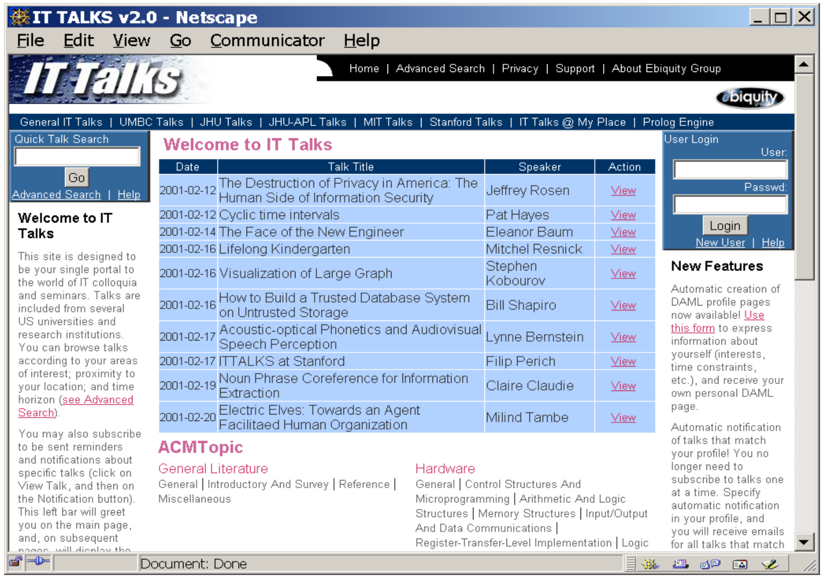
(7) Semantic web & pervasive computing

(8) Trading agent Game in Agentcities

Joint work with Anupam Joshi, Yun Peng, Scott Cost, Tim Oates, Jim Mayfield, Benjamin Grosof, Yelena Yesha, and many students.

(1) ITTALKS

- **ITTALKS** is a database driven web site of IT related talks at UMBC and other institutions. The database contains information on
  - Seminar events
  - People (speakers, hosts, users, …)
  - Places (rooms, institutions, …)
- Web pages with DAML markup are generated
- The DAML markup supports agent-based services relating to these talks.
- Users get talk announcements based on the interests, locations and schedules.


IT TALKS v2.0 - Netscape
File Edit View Go Communicator Help
IT Talks
Home | Advanced Search | Privacy | Support | About Ebiquity Group
ebiquity
General IT Talks | UMBC Talks | JHU Talks | JHU-APL Talks | MIT Talks | Stanford Talks | IT Talks @ My Place | Prolog Engine
Quick Talk Search
Go
Advanced Search | Help
User Login
User:
Passwd:
Login
New User | Help
Document: Done

Welcome to IT Talks

This site is designed to be your single portal to the world of IT colloquia and seminars. Talks are included from several US universities and research institutions. You can browse talks according to your areas of interest; proximity to your location; and time horizon (see Advanced Search).

You may also subscribe to be sent reminders and notifications about specific talks (click on View Talk, and then on the Notification button). This left bar will greet you on the main page, and, on subsequent

Welcome to IT Talks

Date	Talk Title	Speaker	Action
2001-02-12	The Destruction of Privacy in America: The Human Side of Information Security	Jeffrey Rosen	View
2001-02-12	Cyclic time intervals	Pat Hayes	View
2001-02-14	The Face of the New Engineer	Eleanor Baum	View
2001-02-16	Lifelong Kindergarten	Mitchel Resnick	View
2001-02-16	Visualization of Large Graph	Stephen Kobourov	View
2001-02-16	How to Build a Trusted Database System on Untrusted Storage	Bill Shapiro	View
2001-02-17	Acoustic-optical Phonetics and Audiovisual Speech Perception	Lynne Bernstein	View
2001-02-17	ITTALKS at Stanford	Filip Perich	View
2001-02-19	Noun Phrase Coreference for Information Extraction	Claire Claudie	View
2001-02-20	Electric Elves: Towards an Agent Facilitaed Human Organization	Milind Tambe	View

ACMTopic

General Literature
General | Introductory And Survey | Reference | Miscellaneous

Hardware
General | Control Structures And Microprogramming | Arithmetic And Logic Structures | Memory Structures | Input/Output And Data Communications | Register-Transfer-Level Implementation | Logic

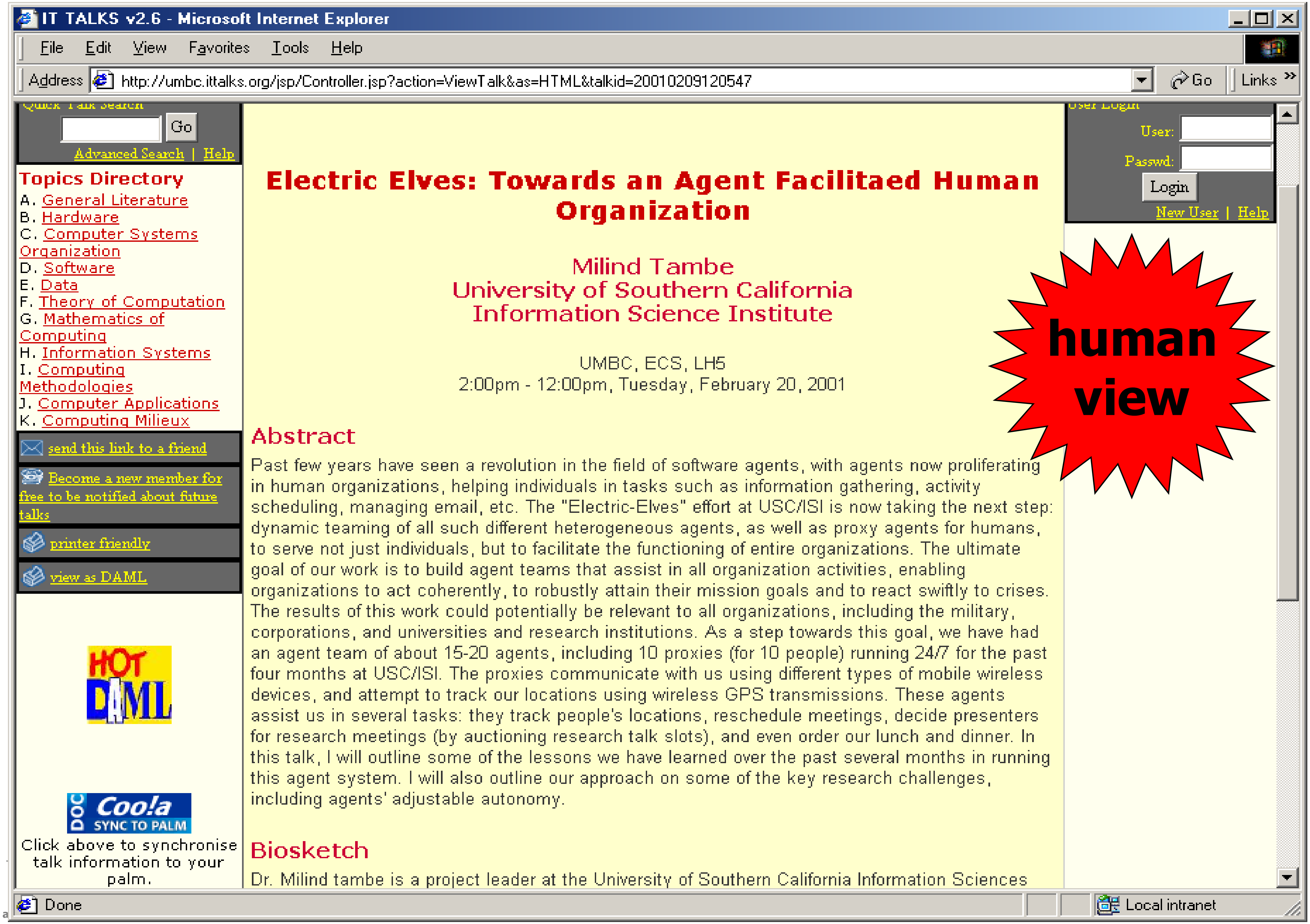
New Features

Automatic creation of DAML profile pages now available! Use this form to express information about yourself (interests, time constraints, etc.), and receive your own personal DAML page.

Automatic notification of talks that match your profile! You no longer need to subscribe to talks one at a time. Specify automatic notification in your profile, and you will receive emails for all talks that match

http://ittalks.org/

UMBC
an Honors University in Maryland



IT TALKS v2.6 - Microsoft Internet Explorer
File
Edit
View
Favorites
Tools
Help
Address
http://umbc.ittalks.org/jsp/Controller.jsp?action=ViewTalk&as=HTML&talkid=20010209120547
Go
Links
Quick Talk Search
Go
Advanced Search | Help
User Login
User:
Passwd:
Login
New User | Help
send this link to a friend
Become a new member for free to be notified about future talks
printer friendly
view as DAML
HOT DAML
Coo!a SYNC TO PALM
Done
Local intranet

Topics Directory

A. General Literature
B. Hardware
C. Computer Systems Organization
D. Software
E. Data
F. Theory of Computation
G. Mathematics of Computing
H. Information Systems
I. Computing Methodologies
J. Computer Applications
K. Computing Milieux

Click above to synchronise talk information to your palm.

Electric Elves: Towards an Agent Facilitaed Human Organization

Milind Tambe
University of Southern California
Information Science Institute

UMBC, ECS, LH5
2:00pm - 12:00pm, Tuesday, February 20, 2001

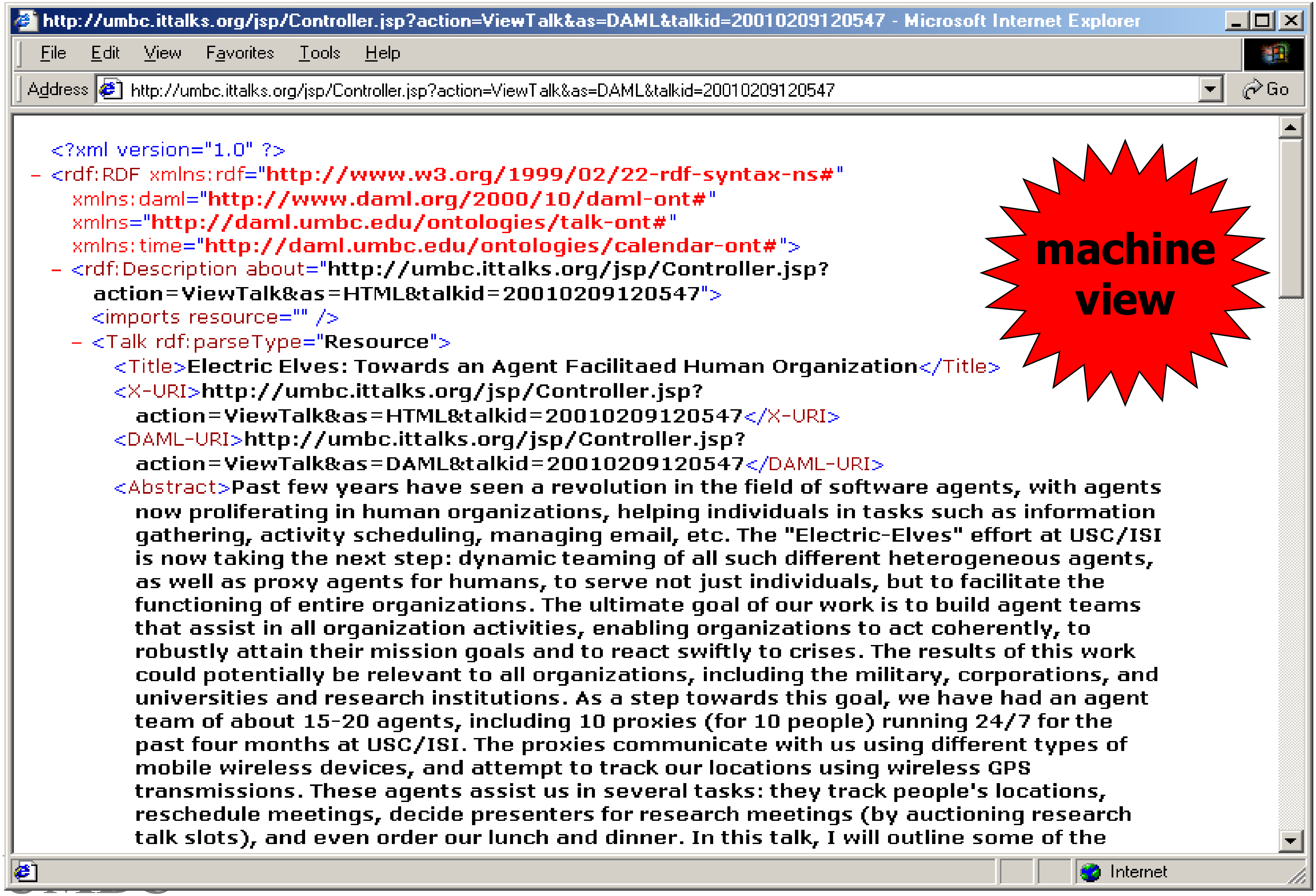
Abstract

Past few years have seen a revolution in the field of software agents, with agents now proliferating in human organizations, helping individuals in tasks such as information gathering, activity scheduling, managing email, etc. The "Electric-Elves" effort at USC/ISI is now taking the next step: dynamic teaming of all such different heterogeneous agents, as well as proxy agents for humans, to serve not just individuals, but to facilitate the functioning of entire organizations. The ultimate goal of our work is to build agent teams that assist in all organization activities, enabling organizations to act coherently, to robustly attain their mission goals and to react swiftly to crises. The results of this work could potentially be relevant to all organizations, including the military, corporations, and universities and research institutions. As a step towards this goal, we have had an agent team of about 15-20 agents, including 10 proxies (for 10 people) running 24/7 for the past four months at USC/ISI. The proxies communicate with us using different types of mobile wireless devices, and attempt to track our locations using wireless GPS transmissions. These agents assist us in several tasks: they track people's locations, reschedule meetings, decide presenters for research meetings (by auctioning research talk slots), and even order our lunch and dinner. In this talk, I will outline some of the lessons we have learned over the past several months in running this agent system. I will also outline our approach on some of the key research challenges, including agents' adjustable autonomy.

Biosketch

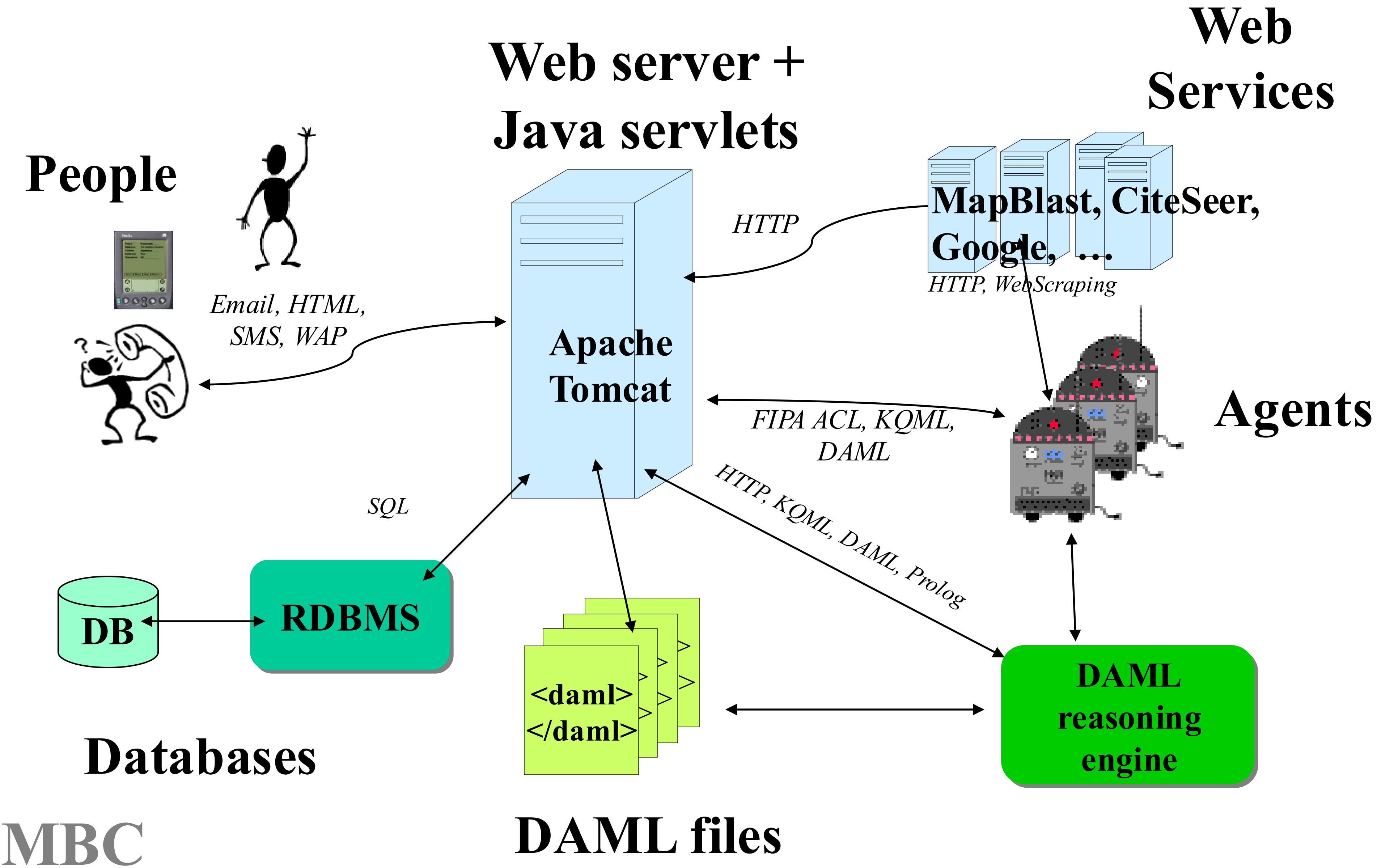
Dr. Milind tambe is a project leader at the University of Southern California Information Sciences

human view

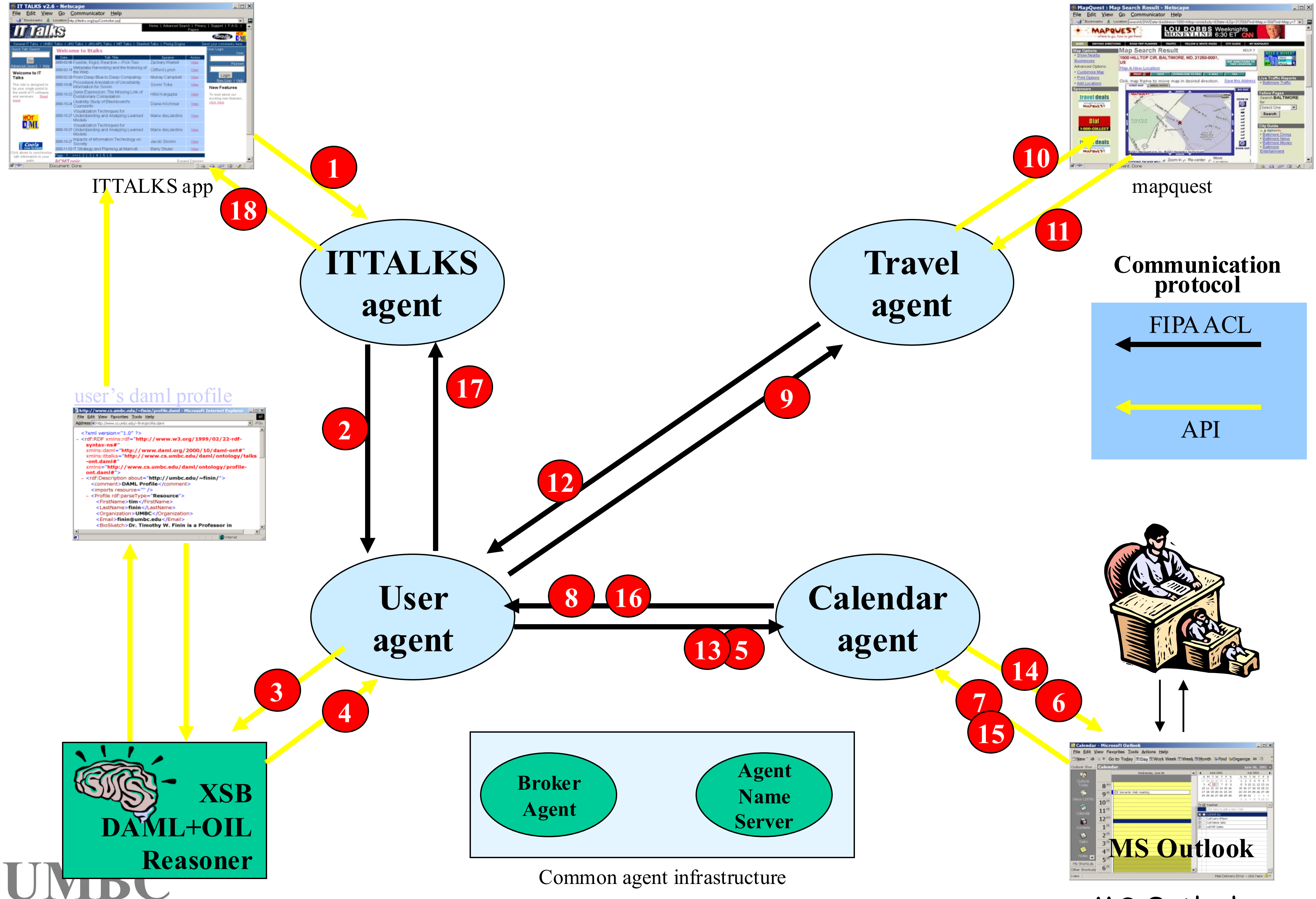

http://umbc.ittalks.org/jsp/Controller.jsp?action=ViewTalk&as=DAML&talkid=20010209120547 - Microsoft Internet Explorer
File
Edit
View
Favorites
Tools
Help
Address
http://umbc.ittalks.org/jsp/Controller.jsp?action=ViewTalk&as=DAML&talkid=20010209120547
Go
Internet
machine
view

```
  <?xml version="1.0" ?>
- <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:daml="http://www.daml.org/2000/10/daml-ont#"
    xmlns="http://daml.umbc.edu/ontologies/talk-ont#"
    xmlns:time="http://daml.umbc.edu/ontologies/calendar-ont#">
  - <rdf:Description about="http://umbc.ittalks.org/jsp/Controller.jsp?
      action=ViewTalk&as=HTML&talkid=20010209120547">
      <imports resource="" />
    - <Talk rdf:parseType="Resource">
        <Title>Electric Elves: Towards an Agent Facilitaed Human Organization</Title>
        <X-URI>http://umbc.ittalks.org/jsp/Controller.jsp?
          action=ViewTalk&as=HTML&talkid=20010209120547</X-URI>
        <DAML-URI>http://umbc.ittalks.org/jsp/Controller.jsp?
          action=ViewTalk&as=DAML&talkid=20010209120547</DAML-URI>
        <Abstract>Past few years have seen a revolution in the field of software agents, with agents
          now proliferating in human organizations, helping individuals in tasks such as information
          gathering, activity scheduling, managing email, etc. The "Electric-Elves" effort at USC/ISI
          is now taking the next step: dynamic teaming of all such different heterogeneous agents,
          as well as proxy agents for humans, to serve not just individuals, but to facilitate the
          functioning of entire organizations. The ultimate goal of our work is to build agent teams
          that assist in all organization activities, enabling organizations to act coherently, to
          robustly attain their mission goals and to react swiftly to crises. The results of this work
          could potentially be relevant to all organizations, including the military, corporations, and
          universities and research institutions. As a step towards this goal, we have had an agent
          team of about 15-20 agents, including 10 proxies (for 10 people) running 24/7 for the
          past four months at USC/ISI. The proxies communicate with us using different types of
          mobile wireless devices, and attempt to track our locations using wireless GPS
          transmissions. These agents assist us in several tasks: they track people's locations,
          reschedule meetings, decide presenters for research meetings (by auctioning research
          talk slots), and even order our lunch and dinner. In this talk, I will outline some of the
```

ITTALKS Architecture



UMBC
an Honors University in Maryland



IT TALKS v2.6 - Netscape

http://ittalks.org/jsp/Controller.jsp

Welcome to Ittalks

Date	Talk Title	Speaker	Action
2000-02-08	Fusible, Rigid, Reactive -- Pick Two	Zachary Wartell	View
2000-02-14	Metadata Harvesting and the Indexing of the Web	Clifford Lynch	View
2000-02-28	From Deep Blue to Deep Computing	Murray Campbell	View
2000-10-06	Procedural Annotation of Uncertainty Information for Sovrin	Sovrin Tolia	View
2000-10-23	Gene Expression: The Missing Link of Evolutionary Computation	Hillol Kargupta	View
2000-10-24	Usability Study of Blackboard's Courseinfo	Diane Krichmar	View
2000-10-27	Visualization Techniques for Understanding and Analyzing Learned Models	Marie desJardins	View
2000-10-27	Visualization Techniques for Understanding and Analyzing Learned Models	Marie desJardins	View
2000-10-27	Impacts of Information Technology on Society	Jacob Slonim	View
2000-11-03	IT Strategy and Planning at Marriott	Barry Shuler	View

Welcome to IT Talks

This site is designed to be your single portal to the world of IT colloquia and seminars. .... Read more

New Features

To read about our exciting new features, click here.

http://www.cs.umbc.edu/~finin/profile.daml - Microsoft Internet Explorer

```
  <?xml version="1.0" ?>
- <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
    syntax-ns#"
    xmlns:daml="http://www.daml.org/2000/10/daml-ont#"
    xmlns:ittalks="http://www.cs.umbc.edu/daml/ontology/talks
    -ont.daml#"
    xmlns="http://www.cs.umbc.edu/daml/ontology/profile-
    ont.daml#">
  - <rdf:Description about="http://umbc.edu/~finin/">
      <comment>DAML Profile</comment>
      <imports resource="" />
    - <Profile rdf:parseType="Resource">
        <FirstName>tim</FirstName>
        <LastName>finin</LastName>
        <Organization>UMBC</Organization>
        <Email>finin@umbc.edu</Email>
        <BioSketch>Dr. Timothy W. Finin is a Professor in
```

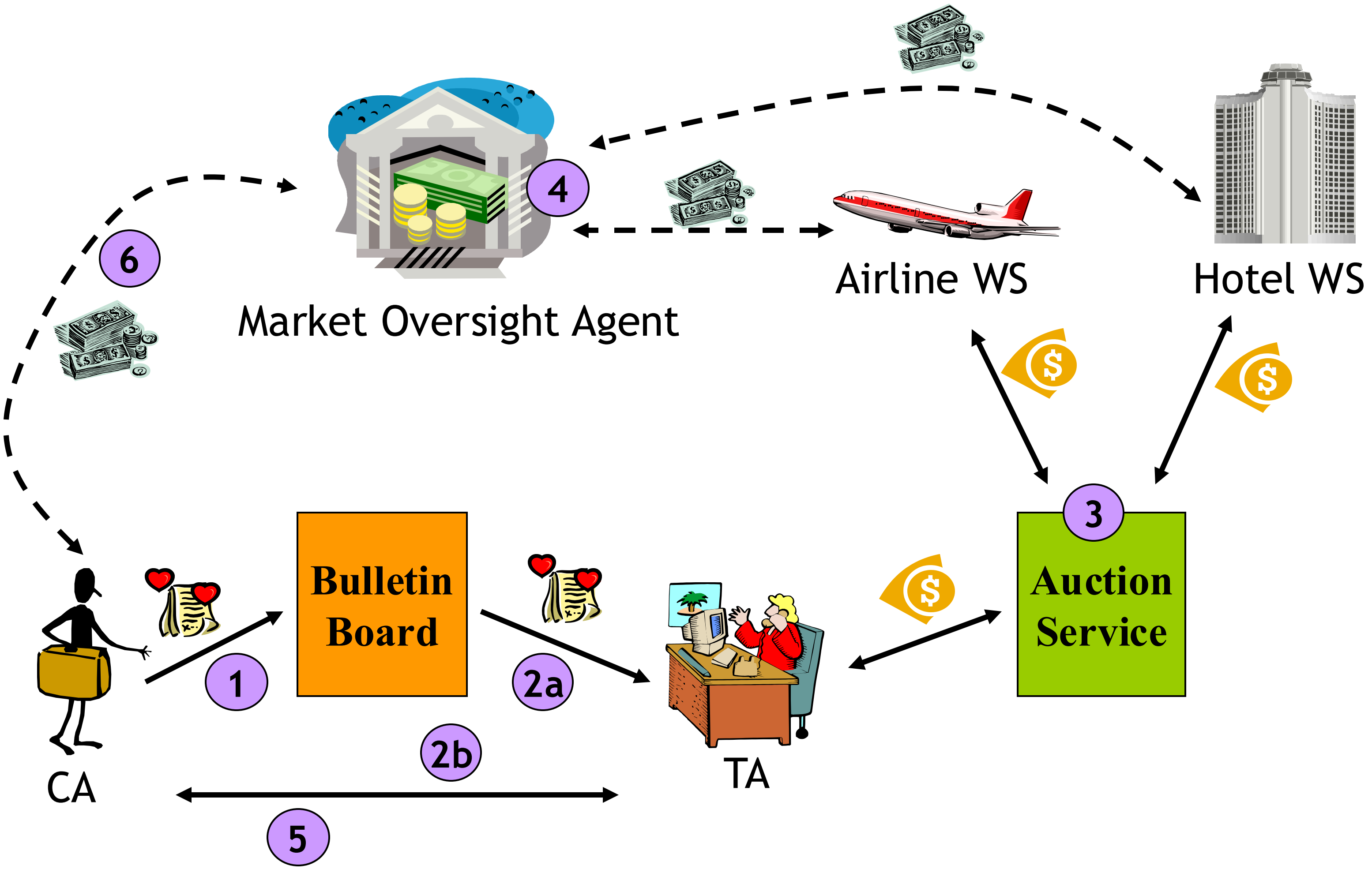
MS Outlook

UMBC
an Honors University in Maryland

(8) Trading Agents

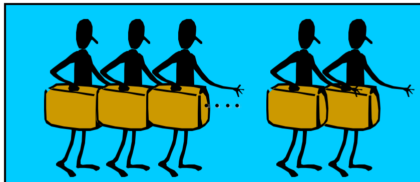
- We’ve built an agent-based environment inspired by TAC, the Trading agent Competition
  - TAC is a forum promoting research on the trading agent problem with games run in 00, 01, 02, 03
  - TAC agents operate within a travel shopping scenario, buying and selling goods to best serve their clients and are scored based on client's preferences for trips assembled, and minimizing cost.
  - TAC in organized around a central auction server
- Our goal is to open up the system, allowing peer-to-peer communication among agents as well various kinds of mediator, auction, discovery, service provider agents.

A Typical Scenario



UMBC
an Honors University in Maryland

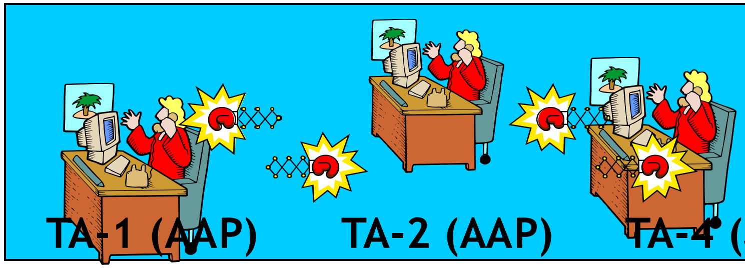
TAGA Agents (1)



Customer Agents

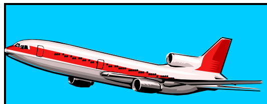
One CA joins the Game every 30 Sec.

- Find travel arrangements
- Save $$

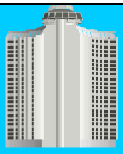


TA-1 (AAP) TA-2 (AAP) TA-4 (JADE)

- Organize travel
- Maximize profits



Airline Web Service

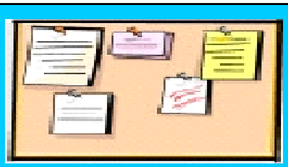


Hotel Web Service

Entertainment Web Service

- sell "goods"
- Maximize profits

TAGA Agents (2)

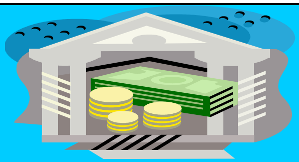


Bulletin Board Agent

- Helps CA find and engage one or more TA

Auction Service Agent

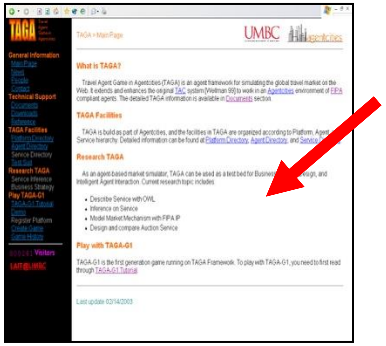
- Operates the auctions markets: English, Dutch,
  Priceline and Hotwire.



Market Oversight Agent

- Manage the financial records
- Announces the winning TA

TAGA in Action



TAGA > Main Page

UMBC agentcities

General Information
Main Page
News
People
Contact
Technical Support
Documents
Downloads
Reference
TAGA Facilities
Platform Directory
Agent Directory
Service Directory
Test Suit
Research TAGA
Service Inference
Business Strategy
Play TAGA-G1
TAGA-G1 Tutorial
Demo
Register Platform
Create Game
Game History

000241 Visitors

LAIT@UMBC

What is TAGA?

Travel Agent Game in Agentcities (TAGA) is an agent framework for simulating the global travel market on the Web. It extends and enhances the original TAC system [Wellman 99] to work in an Agentcities environment of FIPA compliant agents. The detailed TAGA information is available in Documents section.

TAGA Facilities

TAGA is build as part of Agentcities, and the facilities in TAGA are organized according to Platform, Agent and Service hierarchy. Detailed information can be found at Platform Directory, Agent Directory, and Service Directory.

Research TAGA

As an agent based market simulator, TAGA can be used as a test bed for Business Strategy Design, and Intelligent Agent Interaction. Current research topic includes

- Describe Service with OWL
- Inference on Service
- Model Market Mechanism with FIPA IP
- Design and compare Auction Service

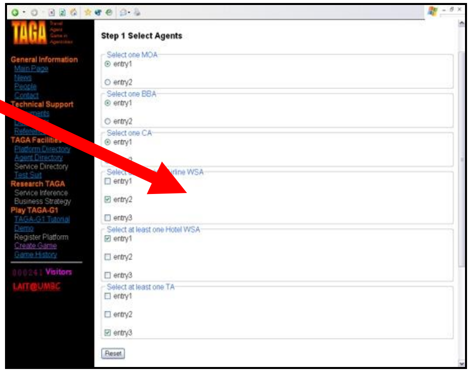
Play with TAGA-G1

TAGA-G1 is the first generation game running on TAGA Framework. To play with TAGA-G1, you need to first read through TAGA-G1 Tutorial

Last update 03/14/2003

TAGA Home Page
http://taga.umbc.edu/

Create a TAGA
game online



Step 1 Select Agents

Select one MOA
● entry1
○ entry2

Select one BBA
● entry1
○ entry2

Select one CA
● entry1

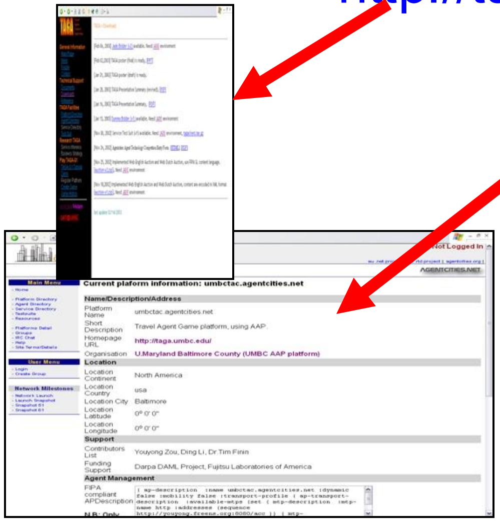
Select at least one Airline WSA
□ entry1
☑ entry2
□ entry3

Select at least one Hotel WSA
☑ entry1
□ entry2
□ entry3

Select at least one TA
□ entry1
□ entry2
☑ entry3

Reset

Download latest TAGA pkg and docs
http://taga.umbc.edu/taga/download/



Not Logged In

AGENTCITIES.NET

Main Menu
- Home
- Platform Directory
- Agent Directory
- Service Directory
- Testsuite
- Resources
- Platforms Detail
- Groups
- IRC Chat
- Help
- Site Terms/Details

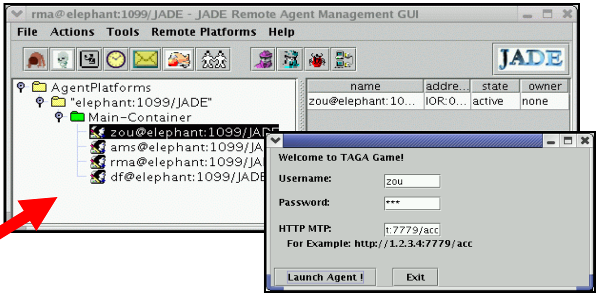
User Menu
- Login
- Create Group

Network Milestones
- Network Launch
- Launch Snapshot
- Snapshot 51
- Snapshot 61

Current plaform information: umbctac.agentcities.net

Name/Description/Address	
Platform Name	umbctac.agentcities.net
Short Description	Travel Agent Game platform, using AAP.
Homepage URL	http://taga.umbc.edu/
Organisation	U.Maryland Baltimore County (UMBC AAP platform)
Location	
Location Continent	North America
Location Country	usa
Location City	Baltimore
Location Latitude	0º 0' 0"
Location Longitude	0º 0' 0"
Support	
Contributors List	Youyong Zou, Ding Li, Dr.Tim Finin
Funding Support	Darpa DAML Project, Fujitsu Laboratories of America
Agent Management	
FIPA compliant APDescription	(ap-description :name umbctac.agentcities.net :dynamic false :mobility false :transport-profile (ap-transport-description :available-mtps (set (mtp-description :mtp-name http :addresses (sequence http://youyong.freens.org:8080/acc)) (mtp-

TAGA on Agentcities net
UMBCTac.agentcities.net/
http://www.agentcities.net/



AgentPlatforms
"elephant:1099/JADE"
Main-Container
zou@elephant:1099/JADE
ams@elephant:1099/JA
rma@elephant:1099/JA
df@elephant:1099/JADE

name	addre...	state	owner
zou@elephant:10...	IOR:0...	active	none

Welcome to TAGA Game!

Username: zou

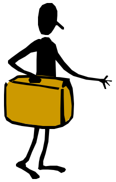
Password: ***

HTTP MTP: t:7779/acc

For Example: http://1.2.3.4:7779/acc

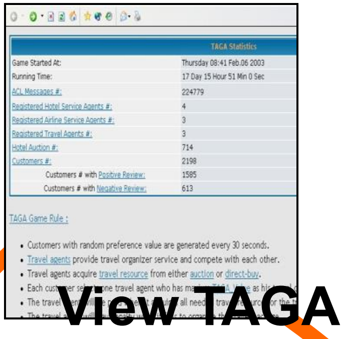
Launch Agent ! | Exit

TAGA supports heterogeneous agent
platform. A FIPA-JADE agent can
interact with a FIPA-AAP agent



The TAGA Game Server

http://taga.umbc.edu/taga/play/demo.htm

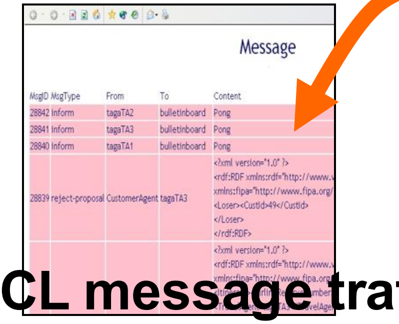


TAGA Statistics	
Game Started At:	Thursday 08:41 Feb.06 2003
Running Time:	17 Day 15 Hour 51 Min 0 Sec
ACL Messages #:	224779
Registered Hotel Service Agents #:	4
Registered Airline Service Agents #:	3
Registered Travel Agents #:	3
Hotel Auction #:	714
Customers #:	2198
Customers # with Positive Review:	1585
Customers # with Negative Review:	613

TAGA Game Rule :

- Customers with random preference value are generated every 30 seconds.
- Travel agents provide travel organizer service and compete with each other.
- Travel agents acquire travel resource from either auction or direct-buy.
- Each customer selects one travel agent who has maximum TAGA Value as his travel
- The travel agent will be paid when it acquires all needed travel resource for the tr
- The travel agent will [illegible]

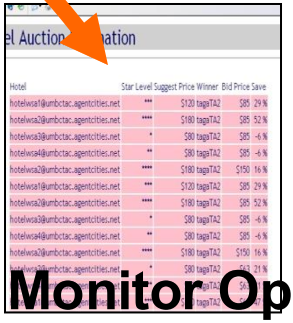
View TAGA game status



Message

MsgID	MsgType	From	To	Content
28842	inform	tagaTA2	bulletinboard	Pong
28841	inform	tagaTA3	bulletinboard	Pong
28840	inform	tagaTA1	bulletinboard	Pong
28839	reject-proposal	CustomerAgent	tagaTA3	<?xml version="1.0" ?> <rdf:RDF xmlns:rdf="http://www.w xmlns:fipa="http://www.fipa.org/ <Loser><CustId>49</CustId> </Loser> </rdf:RDF>
				<?xml version="1.0" ?> <rdf:RDF xmlns:rdf="http://www.w xmlns:fipa="http://www.fipa.org <Itinerary><AirlineReserveNumber <TravelAgent>tagaTA3</TravelAgen

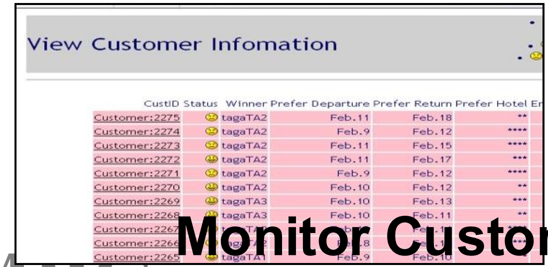
View ACL message traffic



el Auction Information

Hotel	Star Level	Suggest Price	Winner	Bid Price	Save
hotelwsa1@umbctac.agentcities.net	***	$120	tagaTA2	$85	29 %
hotelwsa2@umbctac.agentcities.net	****	$180	tagaTA2	$85	52 %
hotelwsa3@umbctac.agentcities.net	*	$80	tagaTA2	$85	-6 %
hotelwsa4@umbctac.agentcities.net	**	$80	tagaTA2	$85	-6 %
hotelwsa2@umbctac.agentcities.net	****	$180	tagaTA2	$150	16 %
hotelwsa1@umbctac.agentcities.net	***	$120	tagaTA2	$85	29 %
hotelwsa2@umbctac.agentcities.net	****	$180	tagaTA2	$85	52 %
hotelwsa3@umbctac.agentcities.net	*	$80	tagaTA2	$85	-6 %
hotelwsa4@umbctac.agentcities.net	**	$80	tagaTA2	$85	-6 %
hotelwsa2@umbctac.agentcities.net	****	$180	tagaTA2	$150	16 %
hotelwsa3@umbctac.agentcities.net	*	$80	tagaTA2	$63	21 %
[illegible]	**	[illegible]	tagaTA2	$63	[illegible]
[illegible]	***	[illegible]	tagaTA2	[illegible]	[illegible]

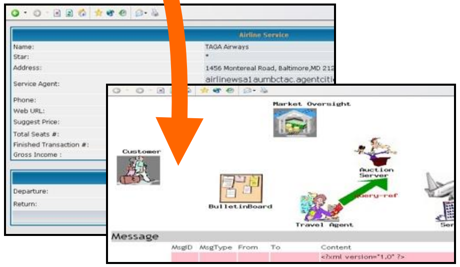
Monitor Open Market Auction



View Customer Infomation

CustID	Status	Winner	Prefer Departure	Prefer Return	Prefer Hotel	Er
Customer:2275	☹	tagaTA2	Feb.11	Feb.18	**	
Customer:2274	☹	tagaTA2	Feb.9	Feb.12	****	
Customer:2273	☹	tagaTA2	Feb.11	Feb.15	****	
Customer:2272	☺	tagaTA2	Feb.11	Feb.17	***	
Customer:2271	☹	tagaTA2	Feb.9	Feb.12	****	
Customer:2270	☺	tagaTA2	Feb.10	Feb.12	**	
Customer:2269	☺	tagaTA3	Feb.10	Feb.13	***	
Customer:2268	☺	tagaTA3	Feb.10	Feb.11	**	
Customer:2267	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	
Customer:2266	[illegible]	tagaTA2	Feb.8	[illegible]	[illegible]	
Customer:2265	☺	tagaTA1	Feb.9	Feb.10	[illegible]	

Monitor Customer Agents



Airline Service	
Name:	TAGA Airways
Star:	*
Address:	1456 Montereal Road, Baltimore,MD 212
Service Agent:	airlinewsa1aumbctac.agentciti
Phone:	
Web URL:	
Suggest Price:	
Total Seats #:	
Finished Transaction #:	
Gross Income :	

Departure:

Return:

Message

MsgID	MsgType	From	To	Content
				<?xml version="1.0" ?>

(and more … new agent, user login, create game, game history)

UMBC

an Honors University in Maryland

TAGA goal and features

- **Goal:** an open test bed for research on agents, FIPA, and SW in an ecommerce environment
- Features:
  - Part of the Agentcities network
  - Everything is a FIPA-compliant agent
  - Supported by OWL ontologies
  - Agents use RDF and OWL as for their content language
  - DAML-S used for service description and discovery
  - New FIPA compliant protocols for various kinds of auctions

TAGA Travel Agent Game in Agentcities

Motivation

- Market dynamics
- Auction theory (TAC)
- Semantic web
- Agent collaboration (FIPA & Agentcities)

Features

- Open Market Framework
- Auction Services
- OWL message content
- OWL Ontologies
- Global Agent Community

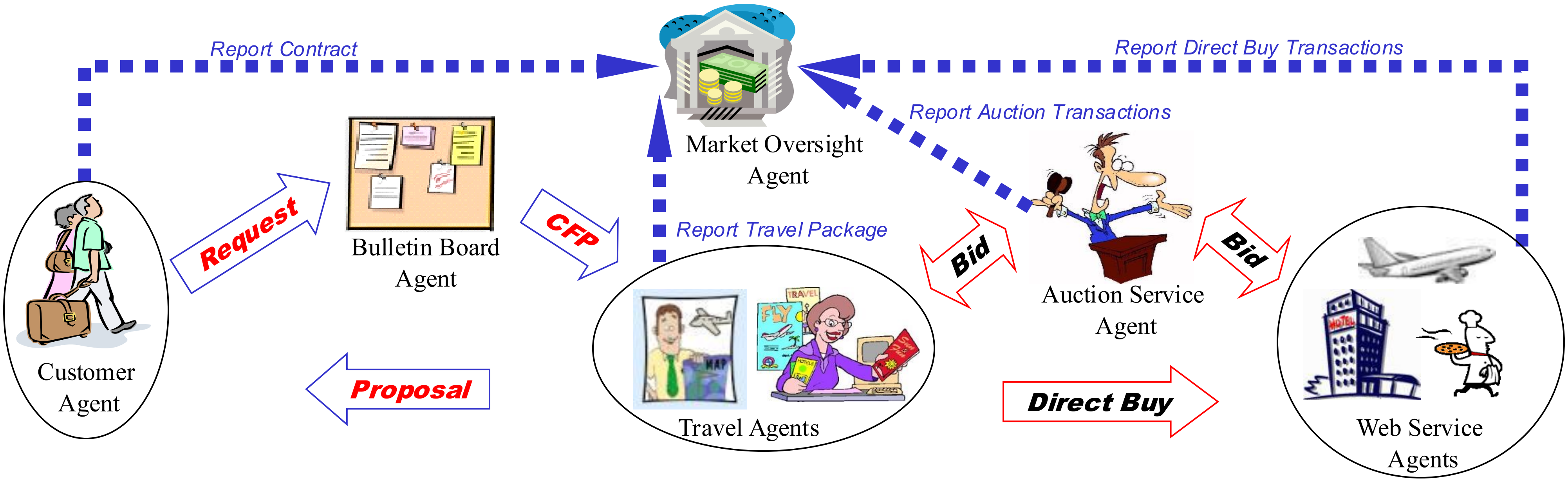
Technologies

- FIPA (JADE, April Agent Platform)
- Semantic Web (RDF, OWL)
- Web (SOAP,WSDL,DAML-S)
- Internet (Java Web Start)

Ontologies

http://taga.umbc.edu/ontologies/

- travel.owl - travel concepts
- fipaowl.owl - FIPA content lang.
- auction.owl - auction services
- tagaql.owl - query language



FIPA platform infrastructure services, including directory facilitators enhanced to use DAML-S for service discovery

http://taga.umbc.edu/

Acknowledgements: DARPA contract F30602-00-2-0591 and Fujitsu Laboratories of America.
Students: Y. Zou, L. Ding, H. Chen, R. Pan. Faculty: T. Finin, Y. Peng, A. Joshi, R. Cost. 4/03

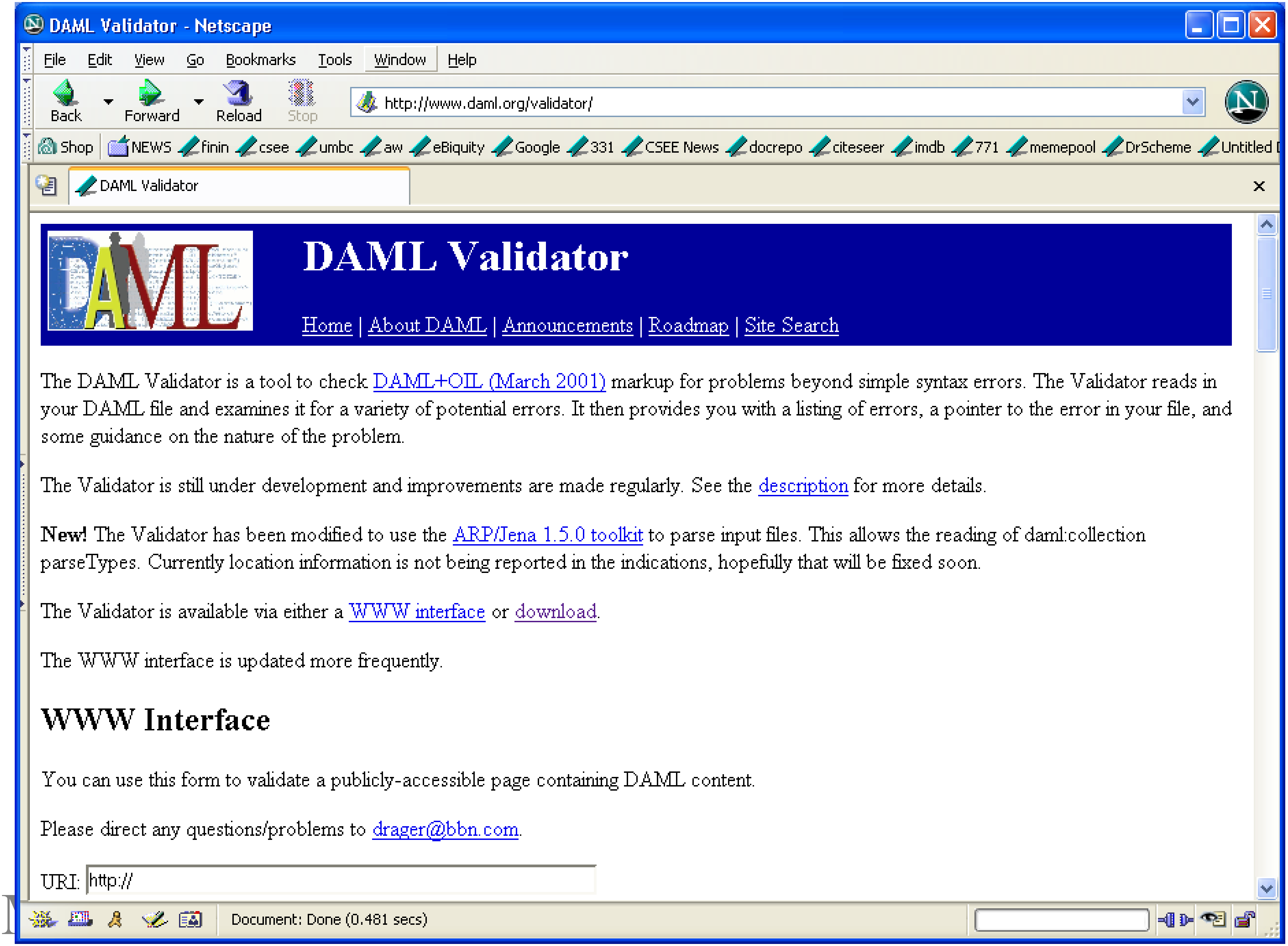
9 Software Tools & Research Issues

Software Tools

- There are a number of “research grade” tools for the semantic web
  - Many are open source
- Some commercial tools and systems are beginning to appear
- Good places to watch:
  - http://w3c.org/
  - http://www.daml.org/
  - http://semanticweb.org/

Kinds of Tools

- **Parsers** that read RDF, DAML, or OWL and produce appropriate data structures.
- **Validators** that read an RDF/DAML/OWL file and identify potential problems
- **Reasoners** that draw inferences licensed by SW languages
- **SW Editors** which allow one to browse and edit SW ontologies while enforcing constraints and/or identifying contradictions.
- **Markup tools and editors** which help authors add SW annotations to other documents.
- **Query languages** designed for SW languages
- **Database systems** for storing SW content and answering queries about it


DAML Validator - Netscape
File Edit View Go Bookmarks Tools Window Help
Back Forward Reload Stop
http://www.daml.org/validator/
Shop NEWS finin csee umbc aw eBiquity Google 331 CSEE News docrepo citeseer imdb 771 memepool DrScheme Untitled
DAML Validator
DAML
Document: Done (0.481 secs)

DAML Validator

Home | About DAML | Announcements | Roadmap | Site Search

The DAML Validator is a tool to check DAML+OIL (March 2001) markup for problems beyond simple syntax errors. The Validator reads in your DAML file and examines it for a variety of potential errors. It then provides you with a listing of errors, a pointer to the error in your file, and some guidance on the nature of the problem.

The Validator is still under development and improvements are made regularly. See the description for more details.

New! The Validator has been modified to use the ARP/Jena 1.5.0 toolkit to parse input files. This allows the reading of daml:collection parseTypes. Currently location information is not being reported in the indications, hopefully that will be fixed soon.

The Validator is available via either a WWW interface or download.

The WWW interface is updated more frequently.

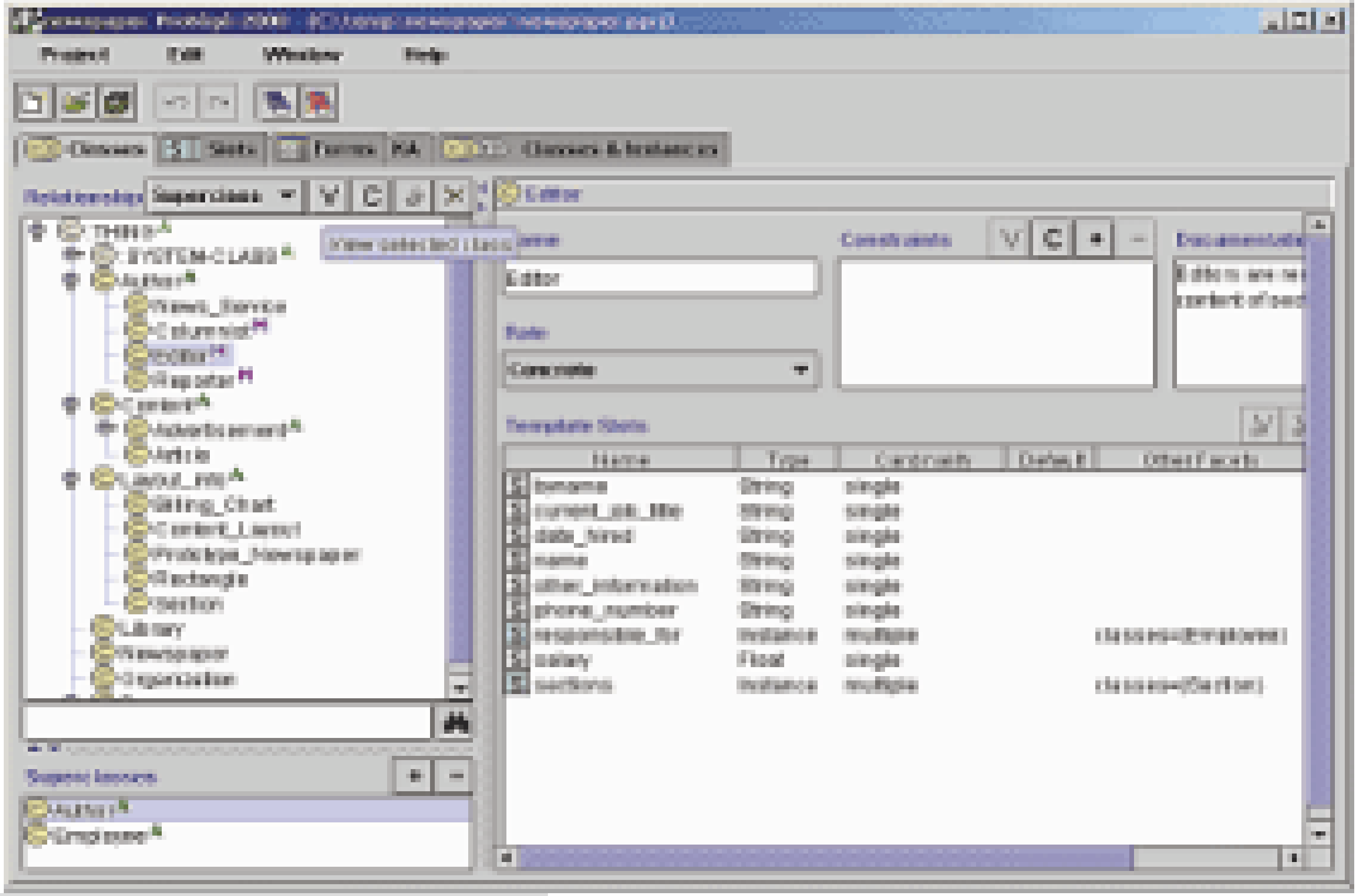
WWW Interface

You can use this form to validate a publicly-accessible page containing DAML content.

Please direct any questions/problems to drager@bbn.com.

URI: http://

Stanford Protégé KB editor



Name: Editor

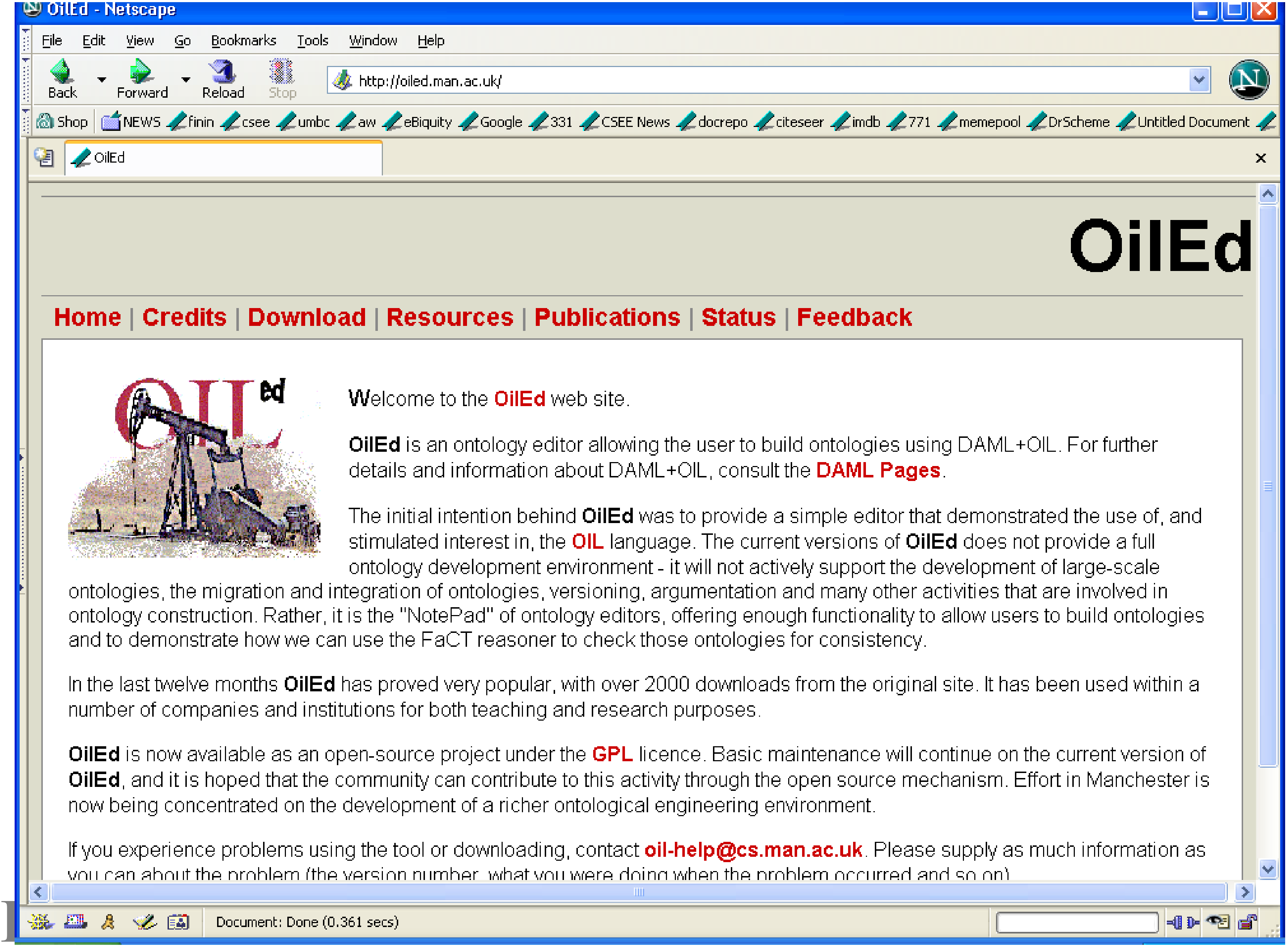
Role: Concrete

Documentation: Editors are re... content of sec...

Template Slots

Name	Type	Cardinality	Default	Other Facets
bioname	String	single		
current_job_title	String	single		
date_hired	String	single		
name	String	single		
other_information	String	single		
phone_number	String	single		
responsible_for	Instance	multiple		classes={Employee}
salary	Float	single		
sections	Instance	multiple		classes={Section}

- Java, open sourced
- Extensible, lots of plug-ins
- Provides reasoning & server capabilities
- Imports and exports RDF
- Alpha version of a DAML+OIL plugin


OilEd - Netscape
File Edit View Go Bookmarks Tools Window Help
Back Forward Reload Stop
http://oiled.man.ac.uk/
Shop NEWS finin csee umbc aw eBiquity Google 331 CSEE News docrepo citeseer imdb 771 memepool DrScheme Untitled Document
OilEd
Document: Done (0.361 secs)

OilEd

Home | **Credits** | **Download** | **Resources** | **Publications** | **Status** | **Feedback**

Welcome to the **OilEd** web site.

OilEd is an ontology editor allowing the user to build ontologies using DAML+OIL. For further details and information about DAML+OIL, consult the **DAML Pages**.

The initial intention behind **OilEd** was to provide a simple editor that demonstrated the use of, and stimulated interest in, the **OIL** language. The current versions of **OilEd** does not provide a full ontology development environment - it will not actively support the development of large-scale ontologies, the migration and integration of ontologies, versioning, argumentation and many other activities that are involved in ontology construction. Rather, it is the "NotePad" of ontology editors, offering enough functionality to allow users to build ontologies and to demonstrate how we can use the FaCT reasoner to check those ontologies for consistency.

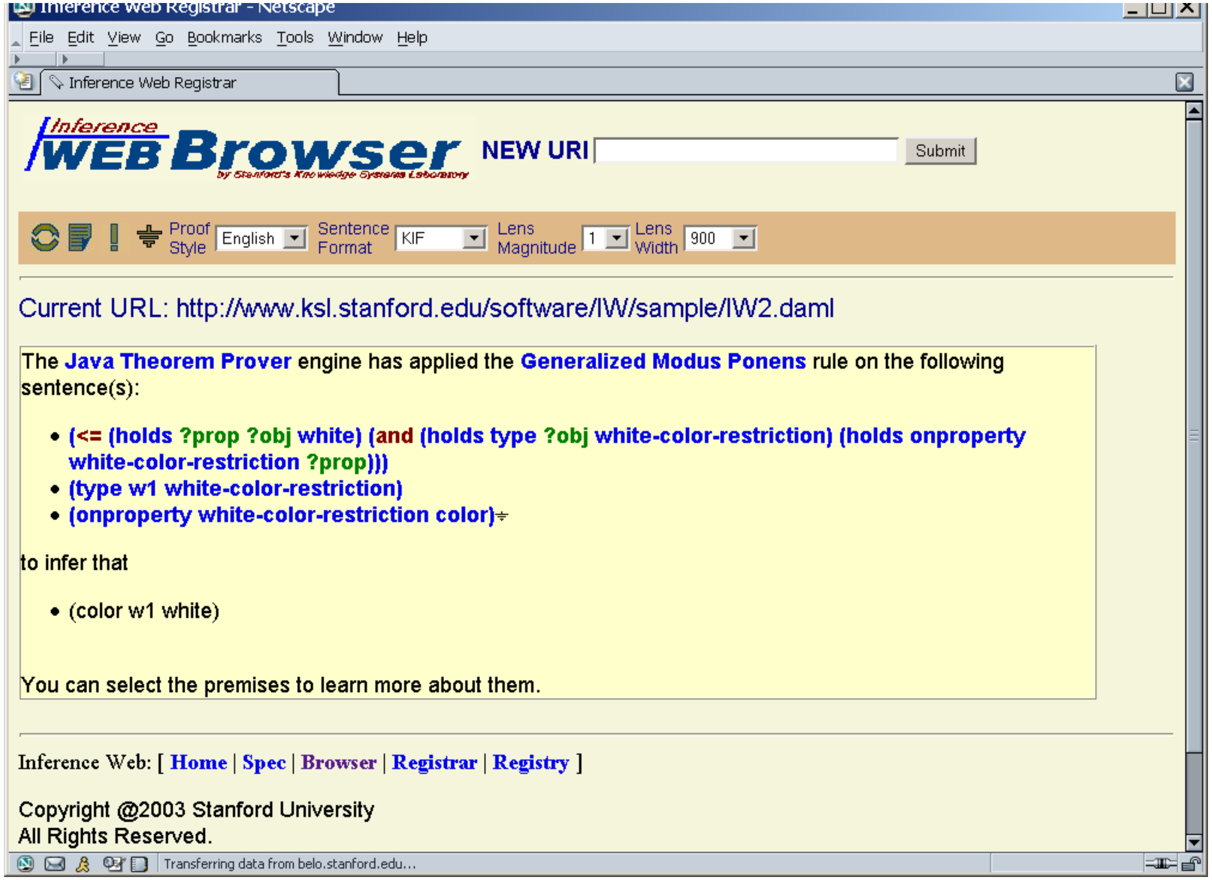
In the last twelve months **OilEd** has proved very popular, with over 2000 downloads from the original site. It has been used within a number of companies and institutions for both teaching and research purposes.

OilEd is now available as an open-source project under the **GPL** licence. Basic maintenance will continue on the current version of **OilEd**, and it is hoped that the community can contribute to this activity through the open source mechanism. Effort in Manchester is now being concentrated on the development of a richer ontological engineering environment.

If you experience problems using the tool or downloading, contact **oil-help@cs.man.ac.uk**. Please supply as much information as you can about the problem (the version number, what you were doing when the problem occurred and so on).

JTP – Java Theorem Prover

- http://www.ksl.stanford.edu/software/JTP/
- JTP has a hybrid reasoning architecture
  - First-order logic model elimination theorem prover
  - Suite of special purpose reasoners
  - Dispatchers and APIs for reasoners
- Works with knowledge in KIF, RDF, DAML+OIL
- Demo: Inference Web Browser


Inference Web Registrar - Netscape
File Edit View Go Bookmarks Tools Window Help
Inference Web Registrar
Inference WEB Browser
by Stanford's Knowledge Systems Laboratory
NEW URI
Submit
Proof Style
Sentence Format
Lens Magnitude
Lens Width
Transferring data from belo.stanford.edu...

Proof Style: English
Sentence Format: KIF
Lens Magnitude: 1
Lens Width: 900

Current URL: http://www.ksl.stanford.edu/software/IW/sample/IW2.daml

The Java Theorem Prover engine has applied the Generalized Modus Ponens rule on the following sentence(s):

- (<= (holds ?prop ?obj white) (and (holds type ?obj white-color-restriction) (holds onproperty white-color-restriction ?prop)))
- (type w1 white-color-restriction)
- (onproperty white-color-restriction color)

to infer that

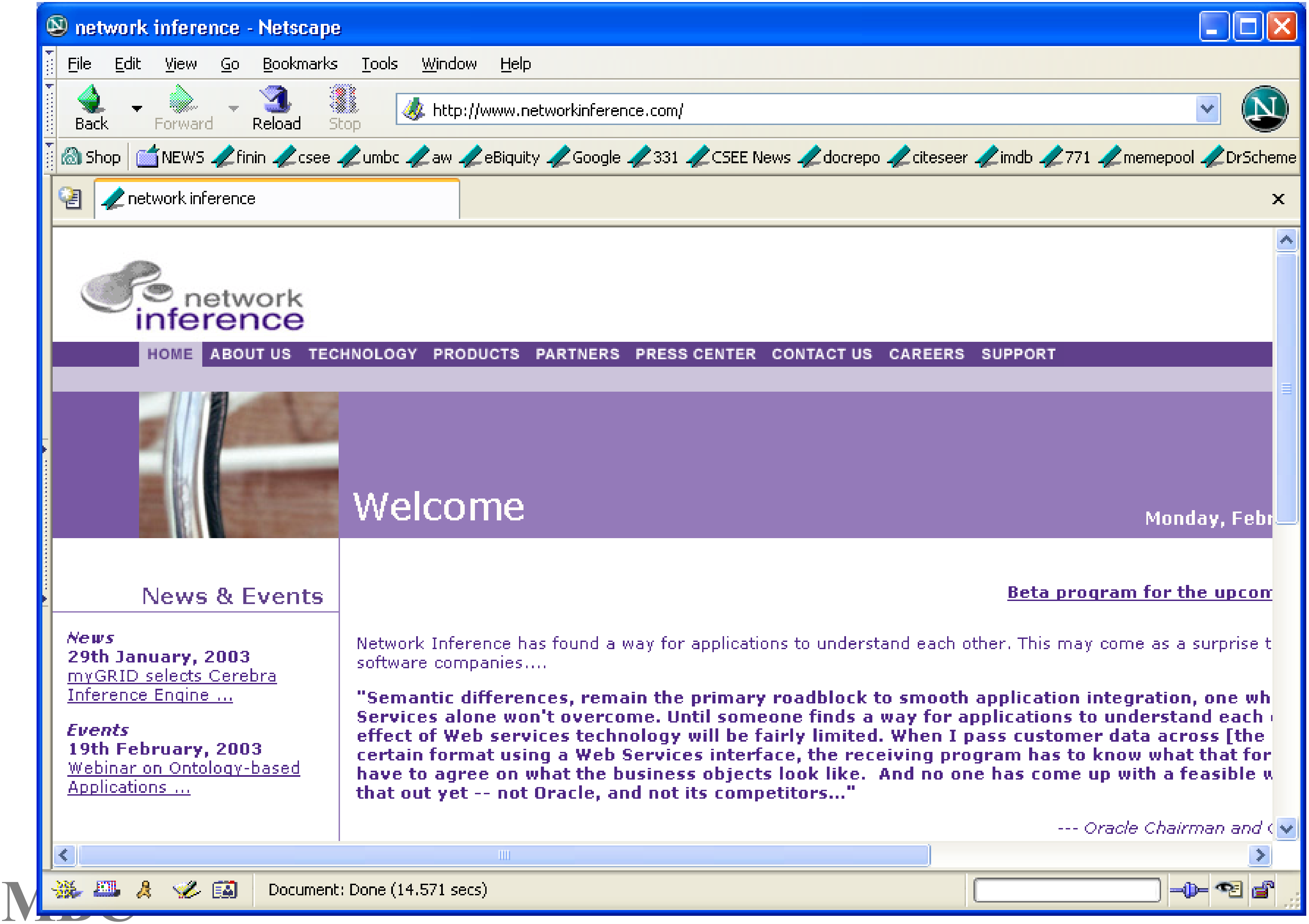
- (color w1 white)

You can select the premises to learn more about them.

Inference Web: [Home | Spec | Browser | Registrar | Registry]

Copyright @2003 Stanford University
All Rights Reserved.

http://belo.stanford.edu:8080/iwbrowser/index.jsp



Welcome

Monday, Febr

News & Events

News
29th January, 2003
mYGRID selects Cerebra Inference Engine ...

Events
19th February, 2003
Webinar on Ontology-based Applications ...

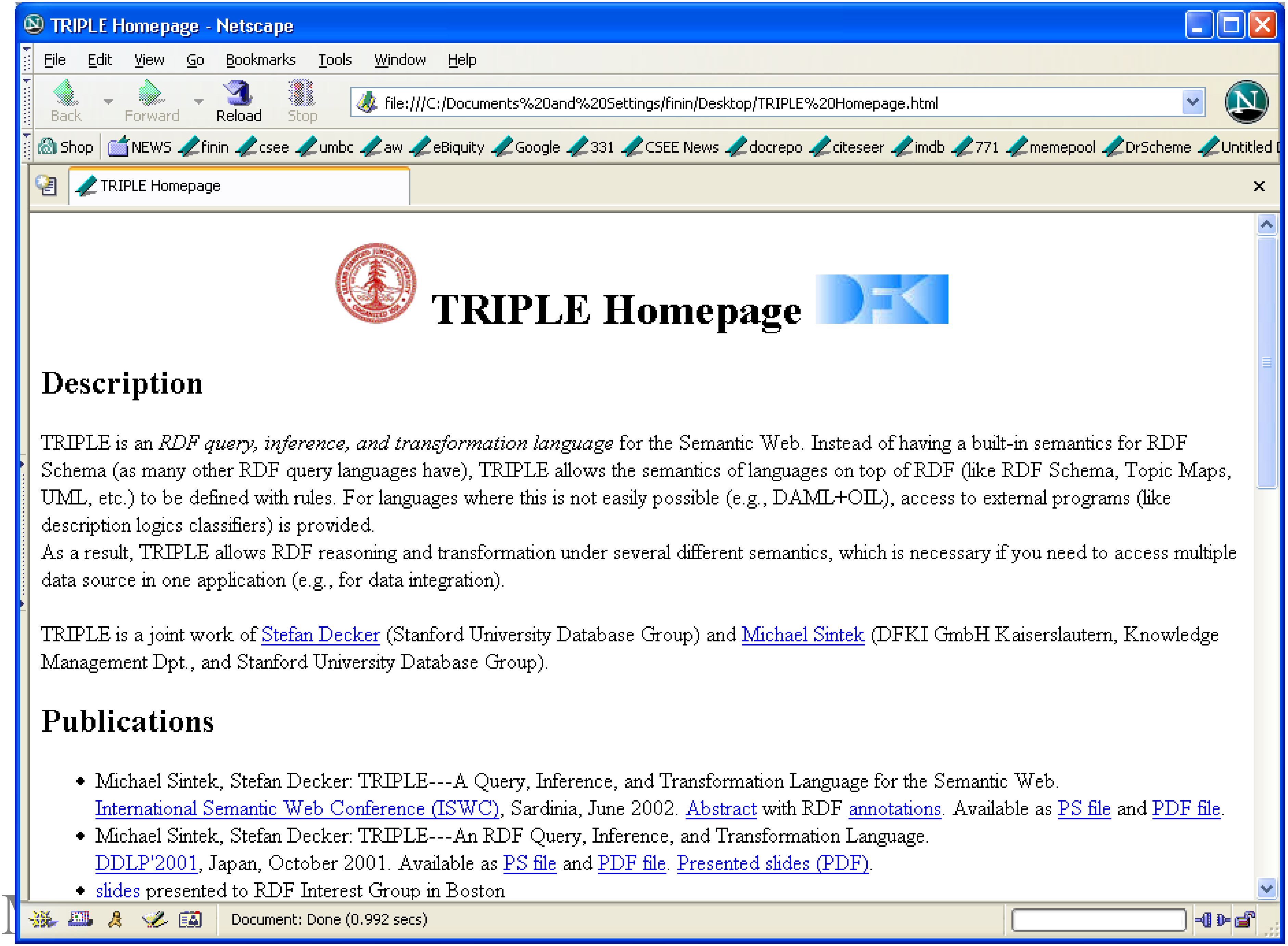
Beta program for the upcon

Network Inference has found a way for applications to understand each other. This may come as a surprise t
software companies....

"Semantic differences, remain the primary roadblock to smooth application integration, one wh
Services alone won't overcome. Until someone finds a way for applications to understand each
effect of Web services technology will be fairly limited. When I pass customer data across [the
certain format using a Web Services interface, the receiving program has to know what that for
have to agree on what the business objects look like. And no one has come up with a feasible w
that out yet -- not Oracle, and not its competitors..."

--- Oracle Chairman and C

Document: Done (14.571 secs)


TRIPLE Homepage - Netscape
File Edit View Go Bookmarks Tools Window Help
Back Forward Reload Stop
file:///C:/Documents%20and%20Settings/finin/Desktop/TRIPLE%20Homepage.html
Shop NEWS finin csee umbc aw eBiquity Google 331 CSEE News docrepo citeseer imdb 771 memepool DrScheme Untitled
TRIPLE Homepage
Document: Done (0.992 secs)

TRIPLE Homepage

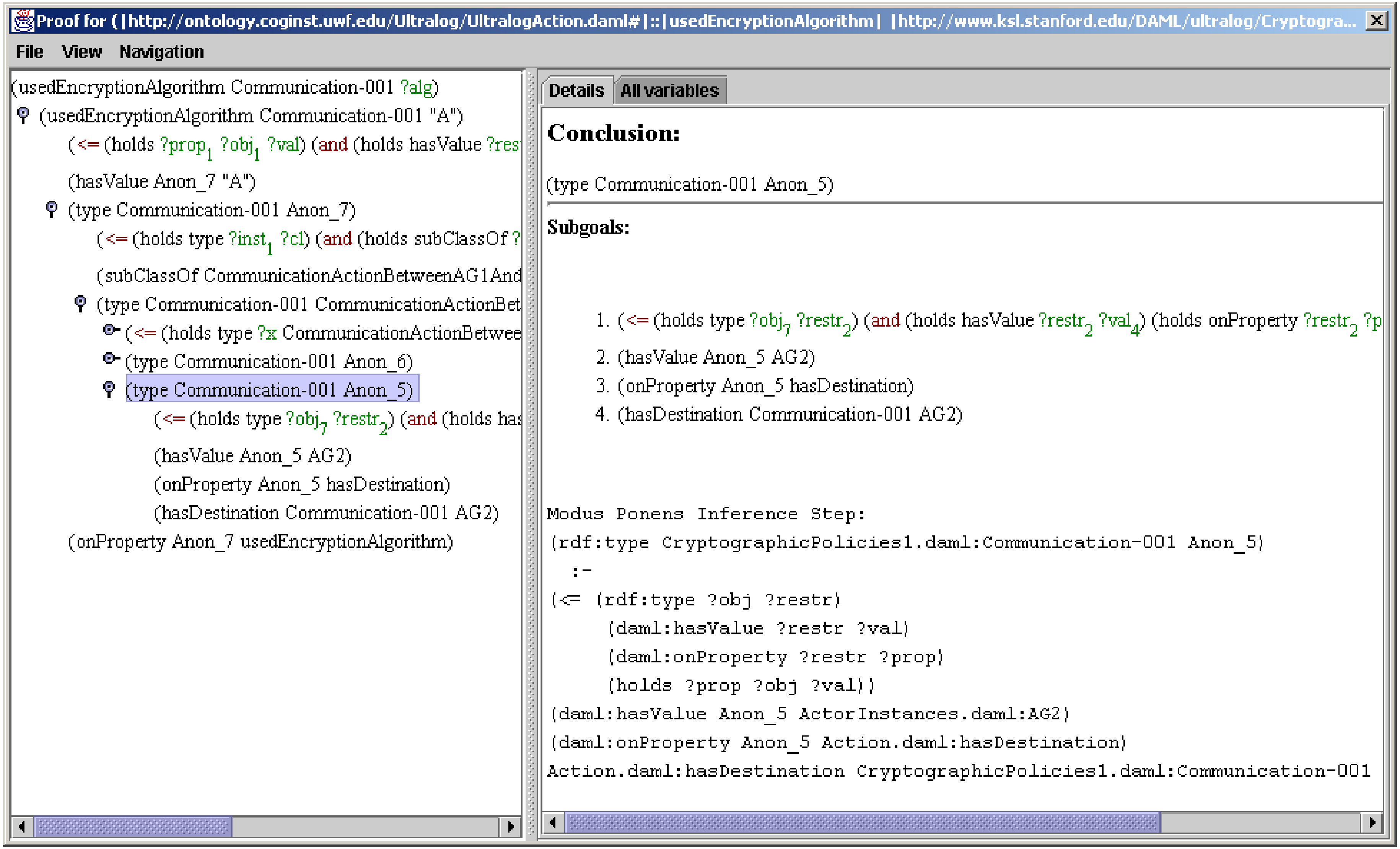
Description

TRIPLE is an *RDF query, inference, and transformation language* for the Semantic Web. Instead of having a built-in semantics for RDF Schema (as many other RDF query languages have), TRIPLE allows the semantics of languages on top of RDF (like RDF Schema, Topic Maps, UML, etc.) to be defined with rules. For languages where this is not easily possible (e.g., DAML+OIL), access to external programs (like description logics classifiers) is provided.
As a result, TRIPLE allows RDF reasoning and transformation under several different semantics, which is necessary if you need to access multiple data source in one application (e.g., for data integration).

TRIPLE is a joint work of Stefan Decker (Stanford University Database Group) and Michael Sintek (DFKI GmbH Kaiserslautern, Knowledge Management Dpt., and Stanford University Database Group).

Publications

- Michael Sintek, Stefan Decker: TRIPLE---A Query, Inference, and Transformation Language for the Semantic Web. International Semantic Web Conference (ISWC), Sardinia, June 2002. Abstract with RDF annotations. Available as PS file and PDF file.
- Michael Sintek, Stefan Decker: TRIPLE---An RDF Query, Inference, and Transformation Language. DDLP'2001, Japan, October 2001. Available as PS file and PDF file. Presented slides (PDF).
- slides presented to RDF Interest Group in Boston



Proof for (|http://ontology.coginst.uwf.edu/Ultralog/UltralogAction.daml#|::|usedEncryptionAlgorithm| |http://www.ksl.stanford.edu/DAML/ultralog/Cryptogra...

(usedEncryptionAlgorithm Communication-001 ?alg)
 (usedEncryptionAlgorithm Communication-001 "A")
  (<= (holds ?prop$_1$?obj$_1$?val) (and (holds hasValue ?res
  (hasValue Anon_7 "A")
  (type Communication-001 Anon_7)
   (<= (holds type ?inst$_1$?cl) (and (holds subClassOf ?
   (subClassOf CommunicationActionBetweenAG1And
   (type Communication-001 CommunicationActionBet
    (<= (holds type ?x CommunicationActionBetwee
    (type Communication-001 Anon_6)
    (type Communication-001 Anon_5)
     (<= (holds type ?obj$_7$?restr$_2$) (and (holds has
     (hasValue Anon_5 AG2)
     (onProperty Anon_5 hasDestination)
     (hasDestination Communication-001 AG2)
  (onProperty Anon_7 usedEncryptionAlgorithm)

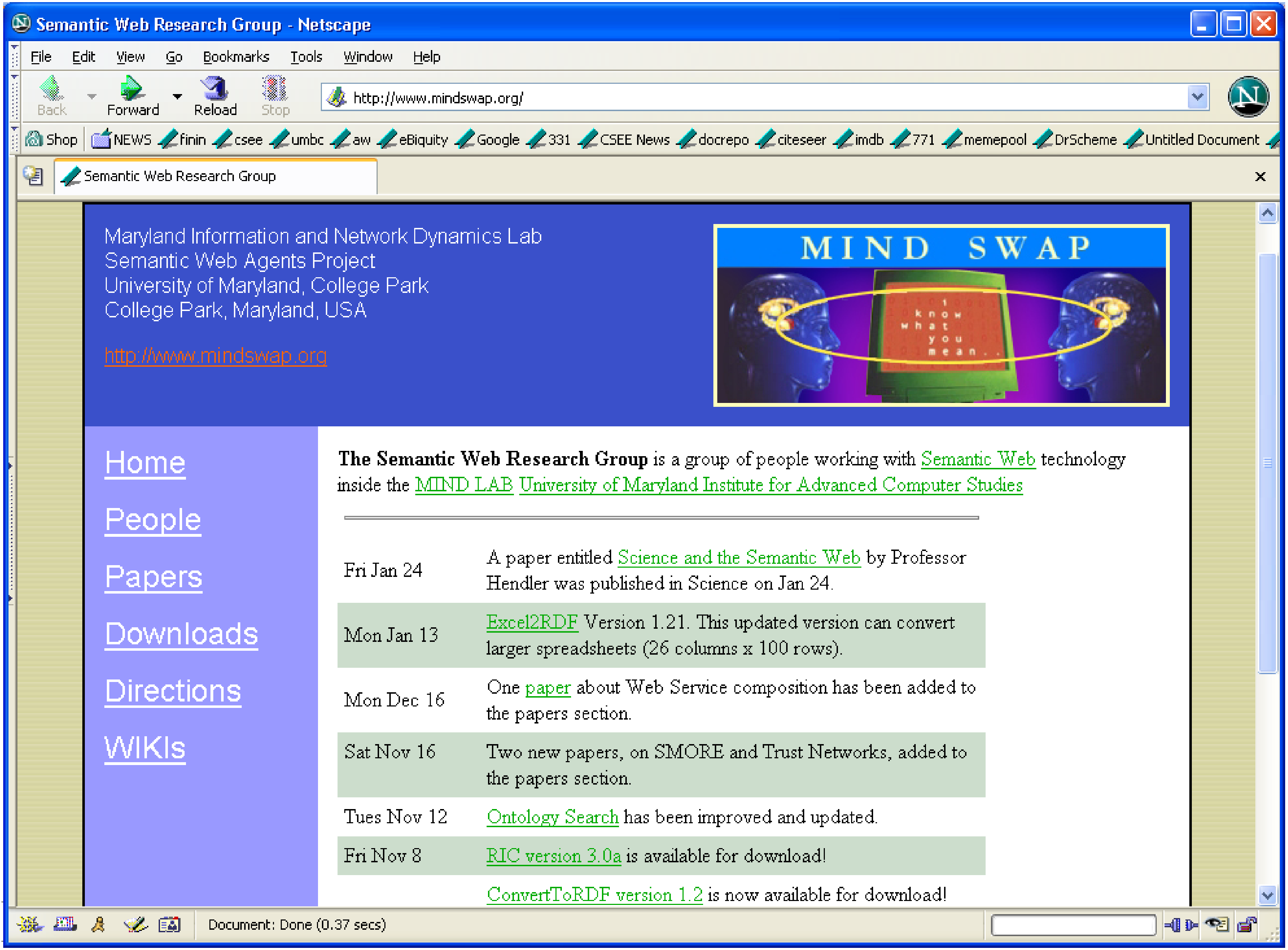
Conclusion:

(type Communication-001 Anon_5)

Subgoals:

1. (<= (holds type ?obj$_7$?restr$_2$) (and (holds hasValue ?restr$_2$?val$_4$) (holds onProperty ?restr$_2$?p
2. (hasValue Anon_5 AG2)
3. (onProperty Anon_5 hasDestination)
4. (hasDestination Communication-001 AG2)

```
Modus Ponens Inference Step:
(rdf:type CryptographicPolicies1.daml:Communication-001 Anon_5)
  :-
(<= (rdf:type ?obj ?restr)
     (daml:hasValue ?restr ?val)
     (daml:onProperty ?restr ?prop)
     (holds ?prop ?obj ?val))
(daml:hasValue Anon_5 ActorInstances.daml:AG2)
(daml:onProperty Anon_5 Action.daml:hasDestination)
Action.daml:hasDestination CryptographicPolicies1.daml:Communication-001
```


Semantic Web Research Group - Netscape
File Edit View Go Bookmarks Tools Window Help
Back Forward Reload Stop
http://www.mindswap.org/
Shop NEWS finin csee umbc aw eBiquity Google 331 CSEE News docrepo citeseer imdb 771 memepool DrScheme Untitled Document
Semantic Web Research Group
MIND SWAP
I know what you mean..
Document: Done (0.37 secs)

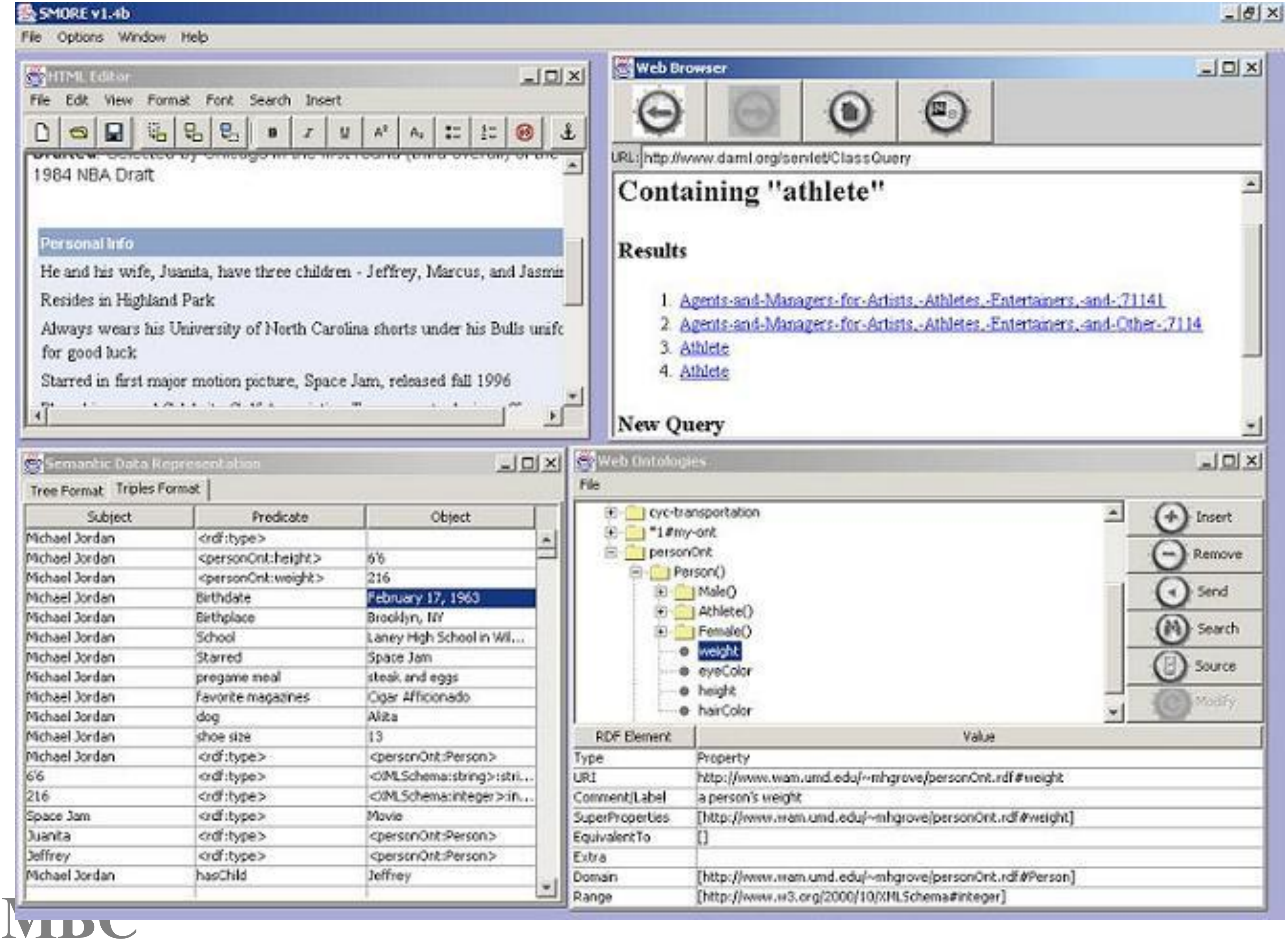
Maryland Information and Network Dynamics Lab
Semantic Web Agents Project
University of Maryland, College Park
College Park, Maryland, USA

http://www.mindswap.org

- Home
- People
- Papers
- Downloads
- Directions
- WIKIs

The Semantic Web Research Group is a group of people working with Semantic Web technology inside the MIND LAB University of Maryland Institute for Advanced Computer Studies

Fri Jan 24	A paper entitled Science and the Semantic Web by Professor Hendler was published in Science on Jan 24.
Mon Jan 13	Excel2RDF Version 1.21. This updated version can convert larger spreadsheets (26 columns x 100 rows).
Mon Dec 16	One paper about Web Service composition has been added to the papers section.
Sat Nov 16	Two new papers, on SMORE and Trust Networks, added to the papers section.
Tues Nov 12	Ontology Search has been improved and updated.
Fri Nov 8	RIC version 3.0a is available for download!
	ConvertToRDF version 1.2 is now available for download!



1984 NBA Draft

Personal Info

He and his wife, Juanita, have three children - Jeffrey, Marcus, and Jasmir

Resides in Highland Park

Always wears his University of North Carolina shorts under his Bulls unifc for good luck

Starred in first major motion picture, Space Jam, released fall 1996

URL: http://www.daml.org/servlet/ClassQuery

Containing "athlete"

Results

1. Agents-and-Managers-for-Artists,-Athletes,-Entertainers,-and-;71141
2. Agents-and-Managers-for-Artists,-Athletes,-Entertainers,-and-Other-;7114
3. Athlete
4. Athlete

New Query

Subject	Predicate	Object
Michael Jordan	<rdf:type>	
Michael Jordan	<personOnt:height>	6'6
Michael Jordan	<personOnt:weight>	216
Michael Jordan	Birthdate	February 17, 1963
Michael Jordan	Birthplace	Brooklyn, NY
Michael Jordan	School	Laney High School in Wil...
Michael Jordan	Starred	Space Jam
Michael Jordan	pregame meal	steak and eggs
Michael Jordan	favorite magazines	Cigar Afficionado
Michael Jordan	dog	Alita
Michael Jordan	shoe size	13
Michael Jordan	<rdf:type>	<personOnt:Person>
6'6	<rdf:type>	<XMLSchema:string>:stri...
216	<rdf:type>	<XMLSchema:integer>:in...
Space Jam	<rdf:type>	Movie
Juanita	<rdf:type>	<personOnt:Person>
Jeffrey	<rdf:type>	<personOnt:Person>
Michael Jordan	hasChild	Jeffrey

RDF Element	Value
Type	Property
URI	http://www.wam.umd.edu/~mhgrove/personOnt.rdf#weight
Comment/Label	a person's weight
SuperProperties	[http://www.wam.umd.edu/~mhgrove/personOnt.rdf#weight]
EquivalentTo	[]
Extra	
Domain	[http://www.wam.umd.edu/~mhgrove/personOnt.rdf#Person]
Range	[http://www.w3.org/2000/10/XMLSchema#integer]

Research Issues

- How will the semantic web be used?
- What kind of KR is needed for the SW?
- How expressive should SW languages be?
- One ontology or many?
- How does industry view the SW?
- How do the web and agents paradigms fit?
- What are good query languages for the SW?

Conclusions

Some key ideas

- Software agents offer a new paradigm for very large scale distributed heterogeneous applications.
- The paradigm focuses on the interactions of autonomous, cooperating processes which can adapt to humans and other agents.
- Agent Communication Languages are a key enabling technology
  - Mobility is an orthogonal characteristic which many, but not all, consider central.
  - Intelligence is always a desirable characteristic but is not strictly required by the paradigm.
- The paradigm is still forming and ACLs will continue to evolve.

Agent Communication

- Agent-agent communication is a key to realizing the potential of the agent paradigm.
- Since interoperability is a defining characteristic of agents, standards are important!
- Candidates for standardization include
  - Agent architecture
  - Agent communication language
  - Agent interaction protocols
  - Agent knowledge
  - Agent programming languages
- Standards will most develop through natural selection, *“nature red in tooth and claw”*

Speculations

- SW might be a chance for us to get intelligent agents out of the lab
  - –Solving the symbol grounding problem
  - –Rethinking agent communication
- How do we get there?

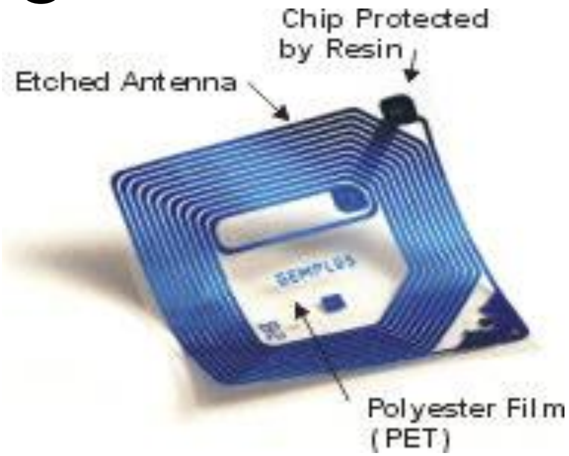
The symbol grounding problem

- An argument against human-like AI is that it's impossible unless machines share our perception of the world.
- A solution to this "symbol grounding problem" is to give agents (soft or hard) human inspired senses.

- But the world we experience is determined by our senses, and human and machine bodies may lead to different conceptions of the world (*cf*. Nagel's *What Is It Like To Be a Bat?*)
- Maybe the Semantic Web is a way out of this problem?

Solving the symbol grounding problem

- The web may become a common world that both humans and agents can understand.
- Confession: the web is more familiar and real to me than much of the real world.
- Physical objects can be tagged with low cost (e.g., $0.05) transponders or RFIDs encoding their URIs
  - – See HP's Cooltown project http://cooltown.com/



Rethinking the agent communication

- Much multi-agent systems work is grounded in Agent Communication Languages (e.g., KQML, FIPA) and associated software infrastructure.
  - This paradigm was articulated ~1990, about the same time as the WWW was developed.
  - Our MAS approach has not yet left the laboratory yet the Web has changed the world.
- Maybe we should try something different?
  - The communication MAS paradigm has been peer-to-peer message oriented communication mediated by brokers and facilitators -- an approach inherited from client-server systems.

Rethinking the agent communication

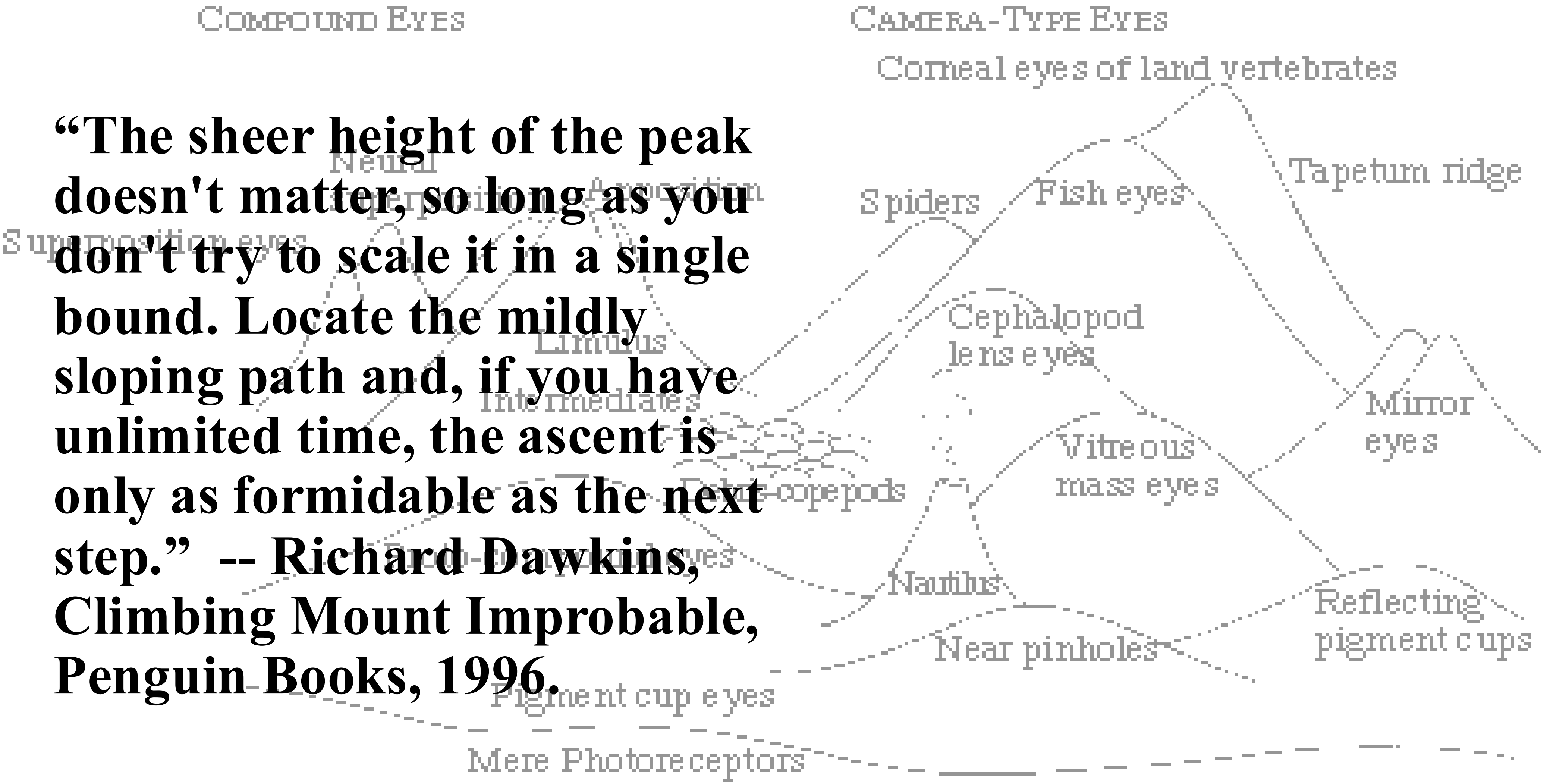
A possible new paradigm?

- Agents “publish” beliefs, requests, and other “speech acts” on web pages.
- Brokers “search” for and “index” published content
- Agents “discover” what peers have published on the web and browse for more details
- Agents “speak for” content on web pages by
  - Answering queries about them
  - Accepting comments and assertions about them

How do we get there from here?

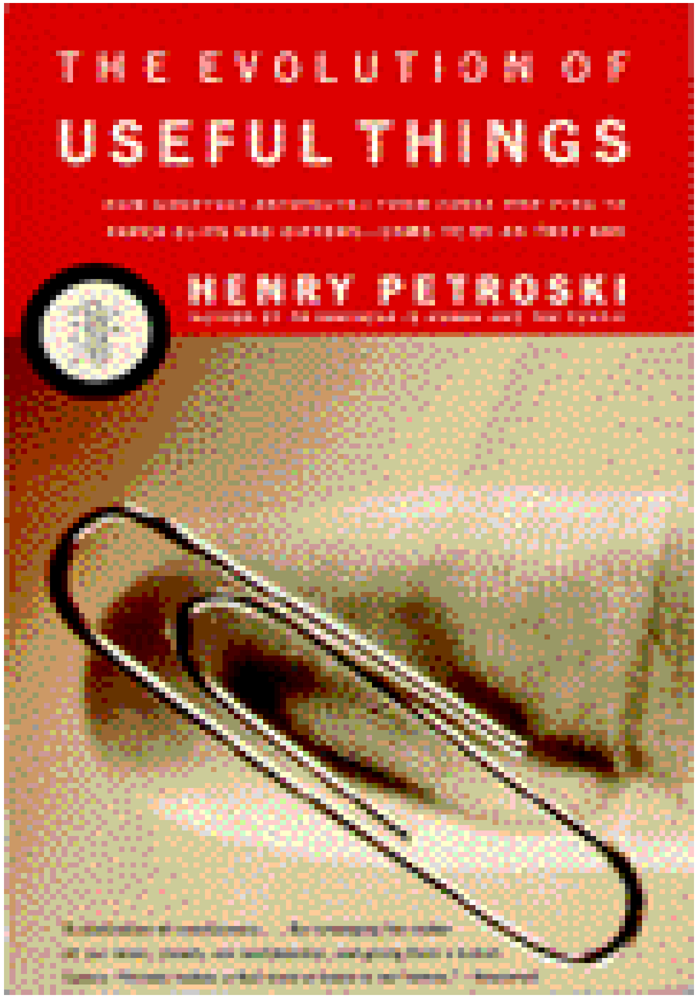
- This semantic web emphasizes ontologies – their development, use, mediation, evolution, etc.
- It will take some time to really deliver on the agent paradigm, either on the Internet or in a pervasive computing environment.
- The development of complex systems is basically an evolutionary process.
- Random search carried out by tens of thousands of researchers, developers and graduate students.

“The sheer height of the peak doesn't matter, so long as you don't try to scale it in a single bound. Locate the mildly sloping path and, if you have unlimited time, the ascent is only as formidable as the next step.” -- Richard Dawkins, Climbing Mount Improbable, Penguin Books, 1996.



Climbing Mount Improbable

The Evolution of Useful Things



- *The Evolution of Useful Things*, Henry Petroski, 1994.
- Prior to the 1890’s, papers were held together with straight pens.
- The development of “spring steel” allowed the invention of the paper clip in 1899.
- It took about 25 years (!) for the evolution of the modern “gem paperclip”, considered to be optimal for general use.

So, we should …

- **Start with the simple and move toward the complex**
  - E.g., from vocabularies to FOL theories
- **Allow many ontologies to bloom**
  - Let natural evolutionary processes select the most useful as common consensus ontologies.
- **Support diversity in ontologies**
  - Monocultures are unstable
  - There should be no THE ONTOLOGY FOR X.
- **The evolution of powerful, machine readable ontologies will take many years, maybe generations**
  - Incremental benefits will more than pay for effort

For More Information

- **Agents:** http://agents.umbc.edu/
- **FIPA:** http://www.fipa.org/
- **KSE:**
  - http://agents.umbc.edu/kqml/
  - http://agents.cs.umbc.edu/kif/
  - http://agents.umbc.edu/ontology/
- **RDF:** http://www.w3.org/RDF/
- **DAML+OIL:** http://www.daml.org/
- **OWL:** http://www.w3.org/2001/sw/
- **Semantic web:** http://semanticweb.org/
- **These slides:** http://agents.umbc.edu/aamas03/

Presenters

Tim Finin

http://umbc.edu/~finin
finin@cs.umbc.edu

Tim Finin is a Professor in the Department of Computer Science and Electrical Engineering at the University of Maryland Baltimore County (UMBC). He has over 30 years of experience in the applications of Artificial Intelligence to problems in information systems, intelligent interfaces and robotics. He holds degrees from MIT and the University of Illinois. Prior to joining the UMBC, he held positions at Unisys, the University of Pennsylvania, and the MIT AI Laboratory. Finin is the author of over 180 refereed publications and has received research grants and contracts from a variety of sources. He has been the past program chair or general chair of several major conferences, is a former AAAI councilor and is AAAI's representative on the board of directors of the Computing Research Association.

Yannis Labrou

http://www.cs.umbc.edu/~jklabrou
yannis@fla.fujitsu.com

Yannis Labrou is a member of Research Staff with Fujitsu Labs of America. Prior to joining Fujitsu he served as the Director of Technology at PowerMarket, Inc. and as a Visiting Assistant Professor at the UMBC Computer Science and Electrical Engineering Department and at the Institute for Global Electronic Commerce. He holds a PhD in Computer Science from UMBC (1996) and a Diploma in Physics from the University of Athens, Greece. Dr. Labrou's research focuses on software agents, Electronic Commerce and the semantic web. Dr. Labrou is a founding member of the FIPA Academy and has been an active participant in the development of the FIPA specifications for software agents standards. He has served on a number of conference organizing committees, program committees, and panels, and has delivered invited tutorials and talks to conferences, research labs and universities. He is the author of more than 30 publications in research journals, books, and conferences. Prior to joining UMBC, Dr. Labrou worked as an intern at the Intelligent Network Technology group of the IBM T.J. Watson Research Center.