Lifecycle of Semantic Web Processes

Jorge Cardoso
Chistoph Bussler
Amit P. Sheth
Wright State University - Main Campus, amit.sheth@wright.edu

Follow this and additional works at: http://corescholar.libraries.wright.edu/knoesis

Part of the Bioinformatics Commons, Communication Technology and New Media Commons, Databases and Information Systems Commons, OS and Networks Commons, and the Science and Technology Studies Commons

Repository Citation
http://corescholar.libraries.wright.edu/knoesis/21

This Presentation is brought to you for free and open access by the The Ohio Center of Excellence in Knowledge-Enabled Computing (Kno.e.sis) at CORE Scholar. It has been accepted for inclusion in Kno.e.sis Publications by an authorized administrator of CORE Scholar. For more information, please contact corescholar@www.libraries.wright.edu.
LIFECYCLE OF SEMANTIC WEB PROCESSES

Jorge Cardoso
Department of Mathematics and Engineering
University of Madeira
9000-390 Funchal – Portugal

Amit Sheth
Department of Computer Science
University of Georgia
Athens, GA – USA
Our Focus

Semantics

Web Processes
- Web Process Composition
- Web Process QoS

Web Services
- Web Service Annotation
- Web Service Discovery
- Web Service QoS
Web Process
An Example

Organization A

\[ t_1 \]
- Setup

\[ t_2 \]
- Prepare Sample

\[ t_3 \]
- Prepare Clones and Sequence

\[ t_4 \]
- Assembly

\[ t_5 \]
- Test Quality

\[ t_6 \]
- Get Sequences

Organization B

\[ t_7 \]
- Sequence Processing

Organization C

\[ t_8 \]
- Process Report

Web Processes
What are Web Processes?

- The next generation workflow technology
- Facilitate the interaction of organizations with markets, competitors, suppliers, customers etc.
- Support enterprise-level and core business activities
  - Encompass the ideas of both intra and inter organizational workflow.
  - Created from the composition of Web services
  - Can use BPEL4WS to represent composition
Globalization of Processes

- Workflows
- Distributed Workflows
- B2B
- E-Services
- Enterprise
- Inter-Enterprise
- Web Processes
- Global

Processes driving the Networked Economy
BIG Challenges

- **Heterogeneity and Autonomy**
  - Syntactic, semantic and pragmatic
  - Complex rules/regulations related to B2B and e-commerce interactions
  - **Solution**: Machine processable descriptions

- **Dynamic** nature of business interactions
  - **Demands**: Efficient Discovery, Composition, etc.

- **Scalability** (Enterprises → Web)
  - **Needs**: Automated service discovery/selection and composition

**Proposition**: Semantics is the most important enabler to address these challenges.
When Web services and Web processes are semantically described, we may call such processes: **Semantic Web Processes**

An ontology provides semantic grounding.
- It includes a **vocabulary of terms**, and some **specification of their meaning**.
- The goal is to create an **agreed-upon vocabulary** and semantic structure for exchanging information about a domain.
Semantic Web Services (OWL-S)

- OWL-S
  - Formerly DAML-S
  - Set of markup language constructs
  - Describe the properties and capabilities of Web services
  - Unambiguous and computer-interpretable
OWL-S

Introduction

OWL-S provides support for the following elements:

- Process description.
- Advertisement and discovery of services.
- Selection, composition & interoperation.
- Invocation.
- Execution and monitoring.
OWL-S defines ontologies for the construction of service models:

- Service Profile
- Process Model
- Service Grounding
The Service Profile provides details about a service.

**Inputs.** Inputs that should be provided to invoke the service.

**Outputs.** Outputs expected after the interaction with the service.

**Preconditions.** Set of conditions that should hold prior to the service being invoked.

**Effects.** Set of statements that should hold true if the service is invoked successfully.
Service Profile
An example of Inputs and Outputs

...<!ENTITY temporal "http://ovid.cs.uga.edu:8080/scube/daml/Temporal.daml">
<!ENTITY address "http://ovid.cs.uga.edu:8080/scube/daml/Address.daml">
...
<input>
	<profile:ParameterDescription rdf:ID="Addr">
		<profile:parameterName> Addr </profile:parameterName>
		<profile:restrictedTo rdf:resource="&address;#Address"/>
		<profile:refersTo rdf:resource="&congo;#congoBuyReceipt"/>
	</profile:ParameterDescription>
</input>
...

<output>
	<profile:ParameterDescription rdf:ID="When">
		<profile:parameterName> When </profile:parameterName>
		<profile:restrictedTo rdf:resource="&temporal;#Date"/>
		<profile:refersTo rdf:resource="&congo;#congoBuyReceipt"/>
	</profile:ParameterDescription>
</output>
...
Lifecycle of semantic Web processes
Semantics for Web Processes

- **Data/Information Semantics**
  - **What**: (Semi-)Formal definition of data in input and output messages of a web service
  - **Why**: for discovery and interoperability
  - **How**: by annotating input/output data of web services using ontologies

- **Functional Semantics**
  - (Semi-) Formally representing capabilities of web service
  - for discovery and composition of Web Services
  - by annotating operations of Web Services as well as provide preconditions and effects
Semantics for Web Processes

- **Execution Semantics**
  - (Semi-) Formally representing the execution or flow of a service in a process or operations in a service
  - for analysis (verification), validation (simulation) and execution (exception handling) of the process models
  - using State Machines, Petri nets, activity diagrams etc.

- **QoS Semantics**
  - (Semi-) Formally describing operational metrics of a web service/process (incl. SLA)
  - To select the most suitable service to carry out an activity in a process
  - using QoS model [Cardoso and Sheth, 2002] and QoS ontology for web services
Lifecycle of semantic Web processes
Web service specifications (e.g. WSDL) only define syntactic characteristics

- Insufficient
- Interoperation of Web services cannot be successfully achieved
- Solution: add meaning to methods and data

Annotation

- Use an ontology to annotate the data involved in Web service operations
- Use an ontology to annotate the Web service’s operations
Semantic annotation of a Web service specified with WSDL
After the service is annotated, it has to be advertised.

The UDDI registry:
- Open doors for the success of service oriented computing
- Should scale to the magnitude of the Web by efficiently discovering relevant services among tens and thousands

Limitations:
- Low precision (many services you do not want)
- Low recall (missed the services you really need to consider)

Challenges:
- Semantic search engines
- Automated discovery
Discovery

- The search of Web services differs from the search of tasks to model workflows
  - The number of Web services available to the composition process
  - In the Web potentially thousands of Web services are available
- Several issues need to be considered:
  - The search has to be based, not only on syntactic information, but also on data, functional, and QoS semantics
  - Enable the automatic determination of the degree of integration of the discovered Web services and a Web process host
- Fundamental steps
  1. Construct a cluster of Web services that match initial requirements
  2. Selected from the cluster the Web service that more closely matches our requirements
  3. The cluster which contains the list of other services, which also match the requirements, is maintained.
     - A service may be chosen later in case of failure or breach of contract.
Selection

- Selection is a need that is almost as important as service discovery.

- Each service can have different Quality of Service aspects:
  - Selection involves locating the service that provides the best quality criteria match.

- **Domain Independent** QoS metrics:
  - There can be some QoS criteria that can be applied to services in all domains irrespective of their functionality or specialty.

- **Domain Specific** QoS metrics:
  - Web services in different domains can have different quality aspects.

- A solution:
  - Use ontologies to define the domain specific and domain independent QoS metrics.
Composition

- The power of Web services can be realized only when they are efficiently composed into Web process
- This stage involves creating a representation of Web processes
  - BPEL4WS
  - BPML
  - WSCI
  - ...
- Facts
  - Web services are highly distributed, autonomous, and heterogeneous
- Requirements
  - High degree of Interoperability among Web services
  - Integration of heterogeneous systems from multiple companies
  - Automating inter-organizational processes across supply chains
- Four kinds of semantics need to be taken into account
  - Functional semantics - functionality of the participating services
  - Data semantics - data that is passed between services
  - QoS semantics - quality of services and the quality of the process as a whole
  - Execution semantics - execution pattern of these services
Execution

- **Execution semantics of a Web service encompasses**
  - The ideas of message sequence
    - request-response
  - Conversation pattern of Web service execution
    - peer-to-peer pattern
    - global controller pattern
  - Flow of actions
    - Sequence
    - Parallel
    - Loops
  - Preconditions and effects of Web service invocation, etc.

- **Formal mathematical models to represent execution semantics**
  - Process Algebra
  - Concurrency formalisms (Petri Nets, state machines)
  - Simulation techniques
  - Etc...
Semantics for Web Process Life-Cycle

Execution (Orchestration?)
- BPWS4J
- Commercial BPEL Execution Engines
- Intalio n3, HP eFlow

Composition (Choreography?)
- BPEL, BPML, WSCI, WSCL, OWL-S, METEOR-S (MWSCF)

Development / Description / Annotation
- WSDL, WSEL
- OWL-S
- METEOR-S (MWSAF)

Publication / Discovery
- UDDI
- WSIL, OWL-S METEOR-S (MWSDI)

Data / Information Semantics
- BPWS4J, BPML, WSCI, WSCL, OWL-S, METEOR-S (MWSCF)
Semantics for Web Process Life-Cycle

Execution (Orchestration?)
- BPWS4J, Commercial BPEL Execution Engines, Intalio n3, HP eFlow

Development / Description / Annotation
- WSDL, WSEL, OWL-S, WSDL-S, METEOR-S (MWSAF)

Composition (Choreography?)
- BPEL, BPML, WSCI, WSCL, OWL-S, METEOR-S (MWSCF)

Publication / Discovery
- UDDI, WSIL, OWL-S, METEOR-S (MWSDI)

Functional / Operational Semantics
Semantics for Web Process Life-Cycle

**QoS Semantics**

**Development / Description / Annotation**
- WSDL, WSEL
- OWL-S
- METEOR-S
- (MWSAF)

**Publication / Discovery**
- UDDI
- WSIL, OWL-S
- METEOR-S
- (MWSDI)

**Composition (Choreography?)**
- BPEL, BPML, WSCI, WSCL, OWL-S, METEOR-S
- (MWSCF)

**Execution (Orchestration?)**
- BPWS4J, Commercial BPEL
- Execution Engines, Intalio n3, HP
- eFlow

**BPEL, BPML, WSCI, WSCL, OWL-S, METEOR-S (MWSCF)**
Semantics for Web Process Life-Cycle

Execution Semantics

Execution (Orchestration?)
- BPWS4J, Commercial BPEL Execution Engines, Intalio, n3, HP eFlow

Composition (Choreography?)
- BPEL, BPML, WSCI, WSCL, OWL-S, METEOR-S (MWSCF)

Development / Description / Annotation
- WSDL, WSEL, OWL-S, WSDL-S, METEOR-S (MWSAF)

Publication / Discovery
- UDDI, WSIL, OWL-S, METEOR-S (MWSDI)
Data and Functional Ontology
An example
QoS Ontology in METEOR-S

An example
Using semantic Web services for E-Tourism: A case study
Introduction - Tourism Industry

- Highly competitive business
- Competitive advantage is driven by
  - Science
  - Information technology
  - Innovation
- Statistics
  - By 2020, tourist travel will increase over 200%
  - 95% of customers use the Internet to gather travel information
  - The number of people using the Internet for travel planning has increased more than 300% over the past 5 years
Dynamic Packaging

Old technology
- Travelers must visit manually multiple independent Web sites to plan their trip
- Register their personal information multiple times
- Spend hours or days waiting for response or confirmation
- Make multiple payments by credit card

Dynamic packaging technology
- Consumers or travel agents can bundle trip components
- Build customized trips
- Combine preferences with flights, car rentals, hotel, and leisure activities in a single price
Current Applications - Expedia

- Expedia pioneered dynamic packaging in 2002 and now gets almost 30% of revenue from package buyers.
- One-stop shopping
- Consumers can book airline tickets and hotel rooms, and also book a shuttle to pick them up at the airport and set up prepaid restaurant meals using.
- Strategy focus on the total journey of consumers
- Dynamic packaging solution is one of the best among the competition
Current Applications - Orbitz

- Started in June 2001
- The third largest online travel site in the world
- Founded by five major airlines, American, Continental, Delta, Northwest and United.
- The main objective was to compete with Expedia and online ticketing sales, hoping to take advantage of increase in ticket sales online.
- Orbitz’s Web site has already completed the full implementation of its dynamic packaging engine
- Customer relationship doesn’t end when a customer buys a travel product
  - Monitors nationwide travel conditions for travelers
  - Provides the latest information on flight delays, weather conditions, gate changes, airport congestion or any other event that might impact travel via mobile phone, pager, PDA or e-mail.
Current Applications - Travelocity

- Owned by Sabre, the world’s largest GDS
- Provides information for more than 700 airlines, more than 55,000 hotels and more than 50 car rental companies
- Strategic acquisition of Site59.com
- Dynamic packaging technology allows Travelocity to respond to the growing popularity of Expedia’s dynamic packages.
- Its dynamic vacation technology will be the first to allow users to book specific airline seats and hotel rooms themselves
The Integration Problem

Dynamic Package

- HDS Sheraton Hotel
- HDS Marriott Hotel
- GDS Amadeus
- GDS Sabre
- CRS Delta Airlines
- CRS Lufthansa
- Airline, Hotel, Car rental, etc website
- Portals and third party websites
- GDS Web site
- On Line Travel Agent
- CRS Avis Car Rental
- CRS Budget Car Rental
- CRS Hertz Car Rental
Data Sources

- **Computerized Reservation System**
  - A Computerized Reservation System (CRS) is a travel supplier’s own central reservation system.

- **Global Distribution System**
  - A GDS is a super switch connecting several CRSs. A GDS integrates travel information about airlines, hotels, car rentals, cruises and other travel products.

- **Hotel Distribution System**
  - Hotel Distribution System (HDS) work closely with GDSs to provide the hotel industry with automated sales and booking services.

- **Direct distribution using supplier Web sites**
  - The Internet is revolutionizing the distribution of tourism information and sales. Small and large companies can have Web sites with an “equal Internet access” to international tourism markets.
Integrating Data Sources

Diagram showing the integration of data sources including:
- Dynamic Package
- Car Rental
- Hotels
- Airlines
- Etc...

Sub-areas include:
- HDS
- CRS
  - Airline/Car/etc
- GDS
  - GDS Web site
  - On Line Travel Agent
  - Ports and third party websites
  - Airline, Hotel, Car rental, etc call center
  - Airline, Hotel, Car rental, etc website
Lack of standards

- The price of tourism products is expressed in many different currencies
  - Euros, dollars, British pounds, etc.
- Time units do not follow a standard
  - Some Web sites state time in hours, others in minutes, others in hours and minutes…etc.
  - For example, 1 hour and 30 minutes, 1h and 30 min, 1:30 h, 90 min, one hour and thirty minutes, ninety minutes, 1:30 pm, etc.
Lack of standards

- Keywords used to express a date are not normalized
  - Some express a day of the week using the words Monday, Tuesday, ..., Sunday, while other use the keywords M, T, ..., Su
- The temperature unit scale is not standard
  - It can be expressed in degrees centigrade as well as in degrees Celsius.
- Numerical values are not expressed in a normalized way
  - 1, 2, and 3 or
  - one, two, and three.
Enabling technologies for Dynamic Packaging

- Semantic Web
- Ontologies
- Web services
- Web processes
Objective

- Integrate e-tourism data sources
- Find a solution to surpass the lack of standards in e-tourism
- Automatically understand the different ways of expressing tourism products
- Create dynamic packages
Our architecture to develop a dynamic packaging infrastructure has four major phases:

- Integration of e-Tourism information sources
- Semantic mediator generation
- Dynamic packaging process generation
- Dynamic packaging final products
Overall Architecture
Integration of e-Tourism information sources

- **Challenges**
  - Develop dynamic packaging applications to integrate the non-standard way of defining e-tourism products
  - No standards to express transportation vehicles, leisure activities, weather conditions, etc.

- **One possible solution**
  - The semantic Web can considerably improve e-Tourism
  - Use of ontologies
  - Use semantic annotation
Data Integration

- A dynamic packaging platform must include provisions for supporting and integrating:
  - Structured data
  - Semi-structured data
  - Unstructured data

- Use a common data representation!!
  - XML
Common data representation

Web service

XML

Airline, Hotel, Car rental, etc.

website

Airline, Hotel, Car rental, etc.
call center

Portals and third party websites

GDS Web site

On Line Travel Agent

WebL

XML

Traditional distribution Model

Unstructured to Semi-Structured data
(Web page->WebL->XML)
XML does not solve the Integration Problem!!!

- XML is a published standard
  - Defines the standard to define tags and relationships
  - But it does not specify a predefined set of tags
  - Leaves the design of a specific set of tags to individual businesses and industries

Different tourism businesses and industries use different XML schema!!!
E-Tourism Ontology

- The e-Tourism ontology provides a way of viewing the world of tourism
  - Achieving interoperability through the use of a shared vocabulary and meanings for terms

- The e-Tourism ontology was created using Protégé and the OWL language
E-Tourism Ontology

- What
- Where
- How
- When

Diagram showing a user engaging in a touristic activity with a map and timeline.
E-Tourism Ontology

- A working group at DERI is also constructing an ontology for the tourism industry
- Our approach differs – it is objective-oriented
  - The ontology is able to answers four types of questions that can be asked when developing a dynamic package.
- These questions involve the predicates *What*, *Where*, *When*, and *How*.
  - *What* can a tourist see and visit
  - *Where* are located the interesting places to see and visit.
  - *When* can the tourist visit a particular place?
    - Day of the week and the hours of the day
    - Atmospheric conditions of the weather
  - *How* can the tourist get to its destination to see or do an activity?
    - Which transportation can he use and which routes to follow.
Semantic registration

- A dynamic packaging infrastructure requires integrating data from XML sources
  - It requires querying in a uniform way and across multiple heterogeneous XML sources containing tourism related information

- Semantics can be used to resolve the differences among the data present in distinct e-Tourism XML sources

- Semantic Registration
  - Maintain a mapping table that maps tags in XML documents with ontological concepts
  - The purpose is to assign semantics to the text between the opening and closing tags
**Semantic Web service Registration**

- **E-Tourism Information Manager**
  - Maintain a mapping table
  - Map Web services with ontological concepts
  - Describe which information a Web service generates
XML Tag Semantic Registration

- **E-Tourism Information Manager**
  - Maintain a tag mapping table
  - Each tag of XML source is mapped with ontological concepts
  - Semantically describe the data in XML sources
Instance Creation

E-Tourism information source manager

CRS Hotel

Create Instance

<Nome>
<Registo>
<Preço>
<NomeHotel>
<Tennis>

XML

Hiking Web site

WebL

<Ident>
<Caminhadas>

XML

Automatic Instance Generation

Flight

Dynamic Package

Reservation

Accommodation

Hotel

Bed&Breakfast

Pestana
Street 234
9000-019

MadB&B
Cural Freiras
9050

Hiking

Madeira Shopping
St. António
9390-200

Levadas
Av. Mar
9000-5

Sport

Activity

Shopping

Web service
Abstract semantic Web process

- An abstract Web process specifies the control-flow and data-flow of an application
  - Does not define which Web services will be executed at runtime
- Abstracting away the resource descriptions allows:
  - Web processes modeling dynamic packages to be portable
  - Reuse processes to generate different process instance at runtime
A dynamic package that includes a fishing experience in the morning, takes the tourist for shopping, schedules a golf game or a movie in the afternoon, and a dinner at night.
Concrete dynamic package Web processes are automatically created using a suitable generator.

The generator may optimize the concrete process based on the availability of Web services.

- Each service is turned into an executable service by specifying the locations of the Web service implementation.
Concrete Dynamic Packaging
Web Process

- An abstract Web process typically originates several concrete processes.
  - Each Web process invokes different Web services
- The processes are valid from a functional point-of-view, but they may not generate valid dynamic package
  - Need to follow time or cost constraints
Conditional Planning

- Select a schedule that is consistent with the overall dynamic package
- Conditional planning
- The main objective of the planning is to schedule an appropriate timeframe during which the tourist will realize a particular activity referenced by a dynamic package
Dynamic Packages and QoS

- At this stage
  - All the dynamic packages are valid
  - Some packages may take more time to execute than others or be more expensive for the tourist
  - They have a distinct QoS (Quality of Service)
- Compute the QoS
  - Use the SWR algorithm
- Ranking and selecting
  - Rank and select the packages which have a set of characteristics that is more similar with the tourist QoS requirements
Examples of Ontologies
Examples of Real Ontologies

MGED Ontology

- The MGED Ontology
  - Provide standard terms for the annotation of microarray experiments.
  - Terms will enable unambiguous descriptions of how the experiment was performed.
  - 212 classes, 101 properties.

- The MGED Ontology is being developed within the microarray community to provide consistent terminology for experiments.

- This community effort has resulted in a list of multiple resources for many species.
  - Approximately 50 other ontologies for different species

- The concepts are structured in DAML+OIL and available in other formats (rdfs)
The MGED Ontology is Structured in DAML+OIL using OILed 3.4

Source: "The MGED Ontology is an Experimental Ontology," 5th Annual Bio-Ontologies meeting (Edmonton, Canada Aug. 2002)
MGED Ontology consists of classes, properties, and individuals (instances)

Source: "OntologyEntry in MAGE," MGED 6 (Aix-en-Provence, France Sept., 2003)
MGED Ontology: Biomaterial Description: Biosource Property: Age

Source: "The MGED Ontology is an Experimental Ontology," 5th Annual Bio-Ontologies meeting (Edmonton, Canada Aug. 2002)
Examples of Real Ontologies

OBO

OBO (Open Biological Ontologies)

- Is an umbrella organization for structured shared controlled vocabularies and ontologies for use within the genomics and proteomics domains.

The ontologies must be open and can be used by all without any constraint other than that their origin must be acknowledged and they can not be altered and redistributed under the same name.

The ontologies are in, or can be instantiated in, a common shared syntax. This may be either the GO syntax, extensions of this syntax, or OWL.

The ontologies share an unique identifier space.

The ontologies include textual definitions of their terms.

The ontologies are orthogonal to other ontologies already lodged with OBO.
Examples of Real Ontologies

GO Ontology

Gene Ontology (GO)

- Describes gene products in terms of their
  - Associated biological processes,
  - cellular components and
  - Molecular functions in a species-independent manner.

GO format - flat files, XML, MySQL

Component ontology
1379 terms
212 KB

Process ontology
8151 terms
4.82 MB

Function ontology
7278 terms
1.16 MB
Tools

- Gene Ontology Editors
  - DAG-Edit, COBrA
- Gene Ontology Browsers
  - AmiGO, MGI GO, QuickGO, EP GO, etc…
- Other tools
  - Aprox. 30 tools
Examples of Toy Ontologies
DAML library

- DAML Ontology Library
  - 282 ontologies
- A few examples
  - http://cicho0.tripod.com/cs_Courses_ont
  - http://daml.umbc.edu/ontologies/calendar-ont.daml
  - http://mnemosyne.umd.edu/~aelkiss/weather-ont.daml
  - http://www.ai.sri.com/daml/ontologies/sri-basic/1-0/Person.daml
  - http://www.kestrel.edu/DAML/2000/12/TIME.daml
  - http://www.daml.ecs.soton.ac.uk/ont/currency.daml
  - ...

Examples of Toy Ontologies

wine.daml

- Classes
  - ALSATIAN-WINE, AMERICAN-WINE, ANJOU, AUSTRALIAN-REGION, BEAUJOLAIS, BLAND-FISH, BORDEAUX, BORDEAUX-REGION, BOURGOGNE-REGION, BURGUNDY, CABERNET-FRANC, CALIFORNIA-WINE, ...

- Properties
  - BODY, COLOR, COURSE, DRINK, FLAVOR, FOOD, GRAPE-SLOT, MAKER, REGION, SUGAR
Ontologies Needed

Extract from U.S. 2002 North American Industry Classification System (NAICS) Industry Ontology

Enterprise

Agriculture
- Crop Production
  - Cotton 11192
  - Wheat 11114
- Animal Production
  - Cattle 112111

Construction
- Coal 2121
- Metal 2122

Mining
- Ore
- Extraction
  - Petroleum 2111
  - Gas 2111

Manufacturing
- Fabricated Metal
  - Sheet 332322
- Machinery
- Capacitors
  - Farm 333111
  - Construction 33312
  - Electronic 334414
The Universal Data Element Framework (UDEF)

- cross-industry metadata identification
- designed to facilitate convergence and interoperability among e-business and other standards.
- provide a means of real-time identification for semantic equivalency
- seeks only be an attribute in the data element

<table>
<thead>
<tr>
<th>UDEF ID = ebXML UID</th>
<th>EIA-836</th>
<th>X12 (EDI)</th>
<th>Vendor A</th>
</tr>
</thead>
<tbody>
<tr>
<td>9_5.8</td>
<td>Product Part Identifier</td>
<td>Product/Service ID</td>
<td>Part No</td>
</tr>
<tr>
<td>9_9</td>
<td>Product Name</td>
<td>Product/Service Name</td>
<td>Supplier</td>
</tr>
<tr>
<td>y.3_9</td>
<td></td>
<td>Entity (Supplier) Name</td>
<td></td>
</tr>
<tr>
<td>e.2_8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>f.g.9</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2_33</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **<ProductPartIdentifier PRD:GUID="9_5.8">123-456-789</ProductPartIdentifier>**
- **<ProductServiceID PRD:GUID="9_5.8">123-456-789</ProductServiceID>**
- **<PartNo PRD:GUID="9_5.8">123-456-789</PartNo>**

Ron Schuldt, Co-Chair, AIA Electronic Enterprise Working Group, XML Standards Relevant to the Aerospace Industry
Ontology Domains

- Aerospace and defense,
- Automotive,
- Consumer products,
- Travel,
- Telecommunications
- Engineering and construction,
- Banking
- Health care
- …
Ontology Editors
Tools: Ontology Editors

- More than 50 applications. A few examples,
  - Protégé 2000
  - OILed
  - WebOnto
  - GKB-Editor
  - Chimaera
  - …
Protégé 2000

Supports OWL

http://protege.stanford.edu/
Chimaera

Analysis: 15 active commands
Class: 2 active commands

Decomposition: One active command
File: 10 active commands
Add to decomposition [Ctrl-Sh-D]

Taxonomy: No active commands

View: 16 active commands

Name: Transportation

Economy-Sector
- Basic Material
- Financial Sector {from Cmu-web-Ontology}
- Services Sector {from Cmu-web-Ontology}
- Utilities Sector {from Cmu-web-Ontology}

- Agricultural-Sector {from World-Fact-Book}
- Industrial-Sector {from World-Fact-Book}
- Service-Industry {from World-Fact-Book}

- Capital Goods Sector {from Cmu-web-Ontology}
- Conglomerates Industry {from Cmu-web-Ontology}
- Consumer Cyclic Sector {from Cmu-web-Ontology}
- Consumer Non-cyclic Sector {from Cmu-web-Ontology}
- Energy Sector {from Cmu-web-Ontology}
- Healthcare Sector {from Cmu-web-Ontology}
- Technology Sector {from Cmu-web-Ontology}

Transportation Sector [Go] {from Cmu-web-Ontology}

http://www.ksl.stanford.edu/software/chimaera/
GKB-Editor
(Generic Knowledge Base Editor)

http://www.ai.sri.com/~gkb/
WebOnto Project

Ontology browsing and editing tool

List of all ontology objects
Toolbar
Graphical Display Area

Browsing sis1-as-gen-design: classes

Java Applet Window
Thank you...

Questions.