Using Facets of Security within a Knowledge-based Framework to Broker and Manage Semantic Web Services

Randy Howard, Larry Kerschberg
George Mason University; Fairfax, VA USA
choward@gmu.edu, kersch@gmu.edu

More Publications at:
http://eceb.gmu.edu/publications.htm
Research Goals

- Provide a framework & methodology to create Virtual Organizations (VO) via Semantic Web Services
- Support end-to-end requirements & life-cycle tasks to create VO on the fly
- Address layers that correspond to Specification, Design and Implementation
- Focus here is on Intelligent Middle-ware Services for Secure Knowledge Management
Where is the VO Knowledge?

- Humans as part of the VO
- Intellectual Property wrapped in Semantic Web Services
- Policies that govern the VO
  - Service-level agreements
  - QoS agreements
- Security Policies and Protocols
- Access Control, Authentication Services for VO
- Virtual Security for GRID Services
Problem Space

- Automate Web Services
  - Apply Semantic Web Technologies (Semantic Web Services)
  - Deal w/ Plethora of Standards and Protocols
- Issues of a Virtual Organization
  - Rapid configuration needed due to temporal nature of requirements;
  - Enterprise Issues of Resource Management, Quality of Service and Negotiation, and
  - Security issues run through every facet of the VO
Solution Space

- Knowledge-based Dynamic Semantic Web Services (KDSWS) Framework
  - Meta-Model for Semantic Web Services
  - Meta-Process (Methodology)
  - Specification Languages based on KDM/KDL

- Specifies:
  - End-to-end tasks of the life-cycle for context,
  - Threads to deal with Management, Workflow, Transaction Control, Interoperation, Security, Transportation and Feedback
  - Enterprise and Local Perspectives
  - Functional Architecture Components
Brokering and Management

- Brokering, or matchmaking, involves [Paolucci, 2004]:
  - Services advertising themselves to a broker
  - Broker handling queries about the available services
  - Mediating the results for the requestor

- Management Levels [Nayak, 2001]:
  - Strategic
  - Asset
  - Value-Chain
KDSWS Framework - Processes

Life-Cycle Tasks

Prepare for Publish → Publish

Prepare for Request → Request

Request → Discover

Discover → Select

Select → Configure

Configure → Deploy

Deploy → Deliver

Deliver → Retire

Retire → Feedback and/or Fulfilled Request

Feedback and/or Fulfilled Request → Master Request

Master Request → Candidate Services

Candidate Services → Select

Select → Configure

Configure → Certified Services

Certified Services → Deploy

Deploy → Confirmed Services

Confirmed Services → Retire

Retire → Feedback and/or Fulfilled Request

Feedback and/or Fulfilled Request → Available Capabilities

Available Capabilities → Requestor Profile

Requestor Profile → Request Profile

Request Profile → Discover

Discover → Interface

Provider Profile → Provider

Provider → Prepare for Publish

Prepare for Publish → Available Capabilities

Available Capabilities → Requestor Profile

Requestor Profile → Request

Request → Request Profile

Request Profile → Master Request

Master Request → Candidate Services

Candidate Services → Select

Select → Configure

Configure → Certified Services

Certified Services → Deploy

Deploy → Confirmed Services

Confirmed Services → Retire

Retire → Feedback and/or Fulfilled Request

Feedback and/or Fulfilled Request → Available Capabilities

Available Capabilities → Requestor Profile

Requestor Profile → Request

Request → Request Profile

Request Profile → Master Request

Master Request → Candidate Services

Candidate Services → Select

Select → Configure

Configure → Certified Services

Certified Services → Deploy

Deploy → Confirmed Services

Confirmed Services → Retire

Retire → Feedback and/or Fulfilled Request

Feedback and/or Fulfilled Request → Available Capabilities

Available Capabilities → Requestor Profile

Requestor Profile → Request
## KDSWS Functional Architecture

<table>
<thead>
<tr>
<th>Layer</th>
<th>Functional Agent Services Architecture</th>
<th>Virtual Agents</th>
<th>Line Agents</th>
<th>Support Agents</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Services</td>
<td>User Agency</td>
<td>Request Preparation</td>
<td>Process</td>
<td>User Profile Administration</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Publication Preparation</td>
<td>Planning</td>
<td></td>
</tr>
<tr>
<td>Intelligent Middleware Services</td>
<td>Functional Services Agency</td>
<td>Broker</td>
<td>Discovery</td>
<td>Ontology</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Federation</td>
<td>Negotiation</td>
<td>Curation</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Feedback</td>
<td>Contracting</td>
<td>QoS Monitoring</td>
</tr>
<tr>
<td>Web Services</td>
<td>Services Coordination Agency</td>
<td>Fulfillment</td>
<td>Service Mediation</td>
<td>Security</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Publication</td>
<td>Workflow Coordination</td>
<td>Registration</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Requesting</td>
<td>Transaction Management</td>
<td>Certification</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Deployment</td>
<td>Configuration</td>
<td>Testing</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Delivery</td>
<td>Classification</td>
<td>Metrics</td>
</tr>
</tbody>
</table>

- **Web Services Protocols**: UDDI, WSDL, SOAP, BEPLWS, OWL-S
- **Grid Interface**

---

**September 25, 2004**

**SKM 2004**
KDL Specification Example

kdsdBlanketsSecurityConstraint

:DESCRIPTION Provider-side security constraints

:SUPERTYPES kdsdSecurity
kdsdConstraint
kdsdProvider

:SUBTYPES kdsdPrivacy

:ATTRIBUTES
kdsdDescription :TYPE Object
kdsdAccessLevel :TYPE Integer
kdsdAuthorityLevel :TYPE Integer
kdsdEncryptMethod :TYPE String :CONSTRAINT In ("x509?", "Kerberos")
kdsdSignatureSwitch :TYPE Boolean
kdsdVisibility :TYPE String :CONSTRAINT In ("Public", "Partner", "Internal")
kdsdIdentity :TYPE Object
kdsdAuthorityLevel :TYPE Integer

:CONSTRAINTS
:CONSTRAINT-ID C-02-1
:CONSTRAINT-CATEGORIES Supply, Security
Allow only partners to access

:PREFERENCES
:PREFERENCE-ID P-02-1
:PREFEHRENCE-CATEGORIES Supply, Security
Prefer medium security for assurance of fund transfer

:HEURISTICS
:HEURISTIC-ID H-02-1
:HEURISTIC-CATEGORIES Supply, Security
Don't let security impede acquisition

:METHODS
:METHOD-ID M-02-1
Check for partner and access level
## Knowledge-based Dynamic Services/Process Language Specification Example

**kdspSearchForProviders**

| :DESCRIPTION | Core Broker activities |
| :GOALS | ProviderSearchGoal (Find services from providers that meet the goals of the request) |
| :TASK | kdspDiscover |
| :THREAD | kdspManagement |
| :OWNER | kdspSearchAgent |
| :STEWARD | kdspKnowledgeSifter |
| :PREDECESSORS | kdspClassifyRequest |
| :SUCCESSORS | kdspCompileSearchResults |
| :STEPS |  |
| :STEPNAME | kdspSearchUDDI |
| :SEQUENCE-NUMBER | 1 |
| :STEP-DESCRIPTION | Search the UDDI registry for acceptable providers and services |
| :DELEGATE | kdspKnowledgeSifter |
| :DELEGATE-TYPE | AGENT |
| :DELEGATE-ROLE | LINE |
| :OPERATION | searchUDDI |
| :METHOD-NAME | kdspKnowledgeSifter.Search |
| :STEP-SUCCESSORS |  |
| :STEP-SUCCESSOR-MODE | Decision |
| :STEP-SUCCESSOR-BRANCH | kdspAdjustSearchParameters |
| :STEP-CONTROL-CONDITION | Insufficient Results |
| :STEP-SUCCESSORS |  |
| :STEP-SUCCESSOR-MODE | Sequential |
| :STEP-SUCCESSOR-BRANCH | kdspRankResults |
| :STEP-CONTROL-CONDITION | Sufficient Results |

### Constraints

| :CONSTRAINT-ID | C-13-1 |
| :CONSTRAINT-CATEGORIES | Search |
| Constraint: kdspSearchReturnLimit (Return only the top 25) |

| :CONSTRAINT-ID | C-13-2 |
| :CONSTRAINT-CATEGORIES | Security |
| Constraint: Select only partners that support PKI |

### Heuristics

| :HEURISTIC-ID | H-13-1 |
| :HEURISTIC-CATEGORIES | Search |
| Constraint: Partners who are in bankruptcy are a bad risk; therefore, do not use services from providers who are in bankruptcy. |
KDSWS Contributions

- Three-tiered framework for specification, design and implementation of Virtual Organizations using Semantic Web Services.
- Languages for enhanced specification of Semantic Web Service requirements for the VO.
- Security issues are addressed in specification, design and implementation phases of VO lifecycle.
- Agency-based functional architecture allows for agent specialization of functional capabilities including security.
- Workflow management of VO “transactions” with end-to-end security.
Conclusions

- Web Services and Semantic Web Services are still in their infancy so new tools and techniques are needed for Secure Knowledge Management within the Virtual Organization.

- The KDSWS Framework is one approach to meeting the above goal.
  - Meta-models capture the data organization,
  - Methodology helps to integrate the plethora of standards
  - Languages embody the meta-model & methodology to allow for “security semantics” specification
  - Integrated specification, design and implementation environment.
Questions and Answers