Acquiring Evolving Technologies: Web Services Standards

Harry L. Levinson
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

October 26, 2006
1. REPORT DATE  
26 OCT 2006

2. REPORT TYPE

3. DATES COVERED  
00-00-2006 to 00-00-2006

4. TITLE AND SUBTITLE  
Acquiring Evolving Technologies: Web Services Standards

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

6. AUTHOR(S)

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)  
Carnegie Mellon University, Software Engineering Institute (SEI), Pittsburgh, PA, 15213

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR’S ACRONYM(S)

11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT  
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:
   a. REPORT  
      unclassified
   b. ABSTRACT  
      unclassified
   c. THIS PAGE  
      unclassified

17. LIMITATION OF ABSTRACT  
Same as Report (SAR)

18. NUMBER OF PAGES  
22

19a. NAME OF RESPONSIBLE PERSON

Standard Form 298 (Rev. 8-98)  
Prescribed by ANSI Std Z39-18
Acquiring Evolving Technologies

Purpose: combine ideas from different systems engineering areas into a repeatable process for managing technology assessments

This presentation discusses

- challenges of acquiring Web services
- why assess technology?
- assessing technology appropriateness
- applicability to net-centricity

Although not detailed, this presentation borrows from

- system and software architecture
- business principles
- process improvement
- technology solutions
- system of systems techniques
Symbols Used in This Presentation

- **Concept**?

- **Example**?

- **Technology**?
Acquisition Challenges

Program Requirements
- Technology Solutions

Schedule and Cost

Program and Cost

Constraints

New Program
- Architectural and Design Decisions
- Program A
- Program B
A *notional program*, Language Translation Services (LTS), helps us explore this topic within a specific context.

**LTS Version 1 (2005)**

- **Purpose**: translate a paragraph of text from one language to another

**Features**

- anyone in the world can create and/or use a translation service
- customization of features (such as accuracy, speed, and dialect) is supported
LTS Upgrade


- Goal: improve accuracy

New Features

- Link up to 10 paragraphs; changes to previous translation responses may be returned
- request translations with additional features including domain, linking, and alternate choices when the accuracy of translation is less than 98%
- the service must report state changes within 10 seconds (for example, degraded performance)
A service-oriented architecture (SOA) was selected as the architecture for LTS Version 1.

SOAs have been described as

- “SOA is about separation” —CBDI
- “supports integrating your business as linked, repeatable, business tasks” —IBM
- “a lifestyle” and “something you do, not something you buy” —Burton Group

Issues with SOAs that we will not discuss today

- organizational and cultural change
- governance
- infrastructure
- adoption techniques
- implementation techniques
Using an SOA approach impacts the quality attributes in different ways.

<table>
<thead>
<tr>
<th>Positive Impact</th>
<th>Neutral Impact</th>
<th>Negative Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interoperability</td>
<td>Reliability</td>
<td>Security</td>
</tr>
<tr>
<td>Extensibility</td>
<td>Availability</td>
<td>Performance</td>
</tr>
<tr>
<td>Adaptability</td>
<td>Scalability</td>
<td>Testability</td>
</tr>
<tr>
<td>Modifiability</td>
<td>Usability</td>
<td>Auditability</td>
</tr>
<tr>
<td>Operability and Deployability</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

[O’Brien 05] Quality Attributes and Service-Oriented Architectures (CMU/SEI-2005-TN-014)
Implementing an SOA Using Web Services Standards

Think of Web services standards (WS-*) as a tool for SOA technology (Burton Group) or standards-based SOA (Sonic).

Launched in the year 2000, arguably

- six years old; today’s hot topic
  - adolescent or mature?

From 50 to 240 specifications

- open framework with a large number of commercial solutions
  - options or confusion?

Three organizations manage the open standards

- many companies large and small participating
  - cooperating or competing?
Why Should We Assess Technologies?

Risks related to acquiring technology

- complexity of implementation
- testing challenges
- managing change
  - neither technology nor programs stand still

DoD policy requires for Major Defense Acquisition Programs (MDAPs) and Major Acquisition Information Systems (MAIS) programs

- Technology Readiness Assessment (TRA) per DoD 5000.2 usually via Technology Readiness Levels (TRLs)
- TRLs assign a single number, which especially for software, does not address the many dimensions of readiness assessment.
Beyond Technology Readiness Levels

Simple, yet meaningful method to assessment

- prototypes or models are meaningful, but difficult and time-consuming to create
- white paper research is not deep enough
- Is there something in between?

Change: a key challenge of assessment

- wait until stable > nothing gets done
- blindly go ahead > everything gets confused
- keep changing the decision > everyone gets confused

Dimensions of the assessment

- ability to meet the requirements
- environmental appropriateness and constraints
- importance to the solution
- lifecycle match [Smith 04]

Processes within the acquisition life cycle must allow decisions to be reevaluated on a regular basis.

Assessing Web Services Standards

**WS-* standards**

- How effective is each standard?
- Where is each standard in the process?
- How much effort is being put into developing the standard?
- conflicting and/or competing standards?
- compatibility and certification?

**Standards process, W3C, OASIS, WS-I**

- Which companies are participating?
- What impact are they having on the process?

**Products available**

- companies implementing and advertising WS-*
- tools to develop and manage WS-* solutions?
- market acceptance, availability?
- opinions of external research organizations?
Assessment Dimensions

Assessing a standard’s maturity

• rate of change
• number of features
• number of features not available
• number of implementations available

Assessing a standard’s impact

• enable, inhibit, or add confusion to system implementation
• trade-off decisions to be made
• potential changes to standards, how it affects architectural decisions

Proposed Analysis Method

• compare the needed system capabilities to SOA quality attributes
• match them with the appropriate Web service standards and
• assess the WS-* maturity and impact on the system
# Initial Analysis of LTS Version 1

<table>
<thead>
<tr>
<th>LTS Capabilities</th>
<th>SOA Quality Attributes</th>
<th>Web Services Maturity and Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add New Services</td>
<td>Interoperability(+)</td>
<td>WS-Discovery(-)</td>
</tr>
<tr>
<td></td>
<td>Availability(.)</td>
<td>WS-BPEL(.)</td>
</tr>
<tr>
<td>World-Wide, Multiple Services</td>
<td>Extensibility(+)</td>
<td>WSDL(+)</td>
</tr>
<tr>
<td></td>
<td>Scalability(.)</td>
<td>ASAP(-)</td>
</tr>
<tr>
<td></td>
<td>Performance(-)</td>
<td>WS-Transfer(.)</td>
</tr>
<tr>
<td>Assorted Functionality</td>
<td>Adaptability(+)</td>
<td>WS-Coordination(-)</td>
</tr>
<tr>
<td></td>
<td>Modifiability(+)</td>
<td>WS-Context(-)</td>
</tr>
<tr>
<td>World-Wide, Multiple Users</td>
<td>Interoperability(+)</td>
<td>WS-Trust(-)</td>
</tr>
<tr>
<td></td>
<td>Availability(.)</td>
<td>UDDI(+)</td>
</tr>
</tbody>
</table>

Combining maturity & impact blurs meaning of each dimension.

Quality attributes and Web service standards are associated with multiple capabilities.

Items that have positive, negative, minimal, plus varied maturity and impact are associated with a single capability.
### Improved Analysis for LTS Version 1

**WS Standard:** Web Services Security (WS-Security)  
**Organization:** OASIS, Ver: 1.0 3/04

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Impact</th>
<th>Maturity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adaptability</td>
<td>Minimal</td>
<td>Mature</td>
</tr>
<tr>
<td>Auditability</td>
<td>Negative</td>
<td>Adolescent</td>
</tr>
<tr>
<td>Availability</td>
<td>Minimal</td>
<td>Mature</td>
</tr>
<tr>
<td>Extensibility</td>
<td>Positive</td>
<td>Mature</td>
</tr>
<tr>
<td>Interoperability</td>
<td>Positive</td>
<td>Mature</td>
</tr>
<tr>
<td>Modifiability</td>
<td>Positive</td>
<td>Mature</td>
</tr>
<tr>
<td>Operability and Deployability</td>
<td>Minimal</td>
<td>Mature</td>
</tr>
<tr>
<td>Performance</td>
<td>Negative</td>
<td>Adolescent</td>
</tr>
<tr>
<td>Reliability</td>
<td>Positive</td>
<td>Mature</td>
</tr>
<tr>
<td>Scalability</td>
<td>Minimal</td>
<td>Mature</td>
</tr>
<tr>
<td>Security</td>
<td>Positive</td>
<td>Adolescent</td>
</tr>
<tr>
<td>Testability</td>
<td>Negative</td>
<td>Adolescent</td>
</tr>
<tr>
<td></td>
<td>Minimal</td>
<td>Mature</td>
</tr>
</tbody>
</table>

**Impact Average:** 0.15  
**Maturity Average:** 0.69

**Use SOA quality attributes to help tradeoff decisions**  
**Separate dimensions for more accurate analysis**  
**Roll up analysis into a single number for quick comparisons**

**Color coding for quick analysis. Include comments to capture reasoning**
## Comparison of Select Standards

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>SOAP</td>
<td>0.15</td>
<td>0.15</td>
<td>0.77</td>
<td>0.85</td>
</tr>
<tr>
<td>WSDL</td>
<td>0.23</td>
<td>0.38</td>
<td>0.69</td>
<td>0.31</td>
</tr>
<tr>
<td>UDDI</td>
<td>0.38</td>
<td>0.38</td>
<td>0.62</td>
<td>0.62</td>
</tr>
<tr>
<td>WS-Security</td>
<td>0.15</td>
<td>0.15</td>
<td>0.69</td>
<td>0.54</td>
</tr>
<tr>
<td>WS-BPEL</td>
<td>0.08</td>
<td>0.23</td>
<td>-0.31</td>
<td>-0.62</td>
</tr>
<tr>
<td>WS-Transfer</td>
<td>0.00</td>
<td>0.00</td>
<td>-0.15</td>
<td>0.08</td>
</tr>
<tr>
<td>WS-Trust</td>
<td>0.00</td>
<td>0.00</td>
<td>-0.54</td>
<td>-0.54</td>
</tr>
<tr>
<td>WS-Coordination</td>
<td>0.23</td>
<td>0.23</td>
<td>0.69</td>
<td>-0.54</td>
</tr>
<tr>
<td>WS-Context</td>
<td>0.15</td>
<td>0.31</td>
<td>-1.00</td>
<td>-0.15</td>
</tr>
<tr>
<td>WS-Discovery</td>
<td>0.15</td>
<td>0.15</td>
<td>-1.00</td>
<td>-1.00</td>
</tr>
</tbody>
</table>

Maturity:
- Immature
- Adolescent
- Mature

Impact:
- Negative
- Minimal
- Positive

**Note:** The values represent the maturity and impact of each standard in the years 2005 and 2006.
Net-Centric Acquisition Challenges

Operational
  • implement capability using varied and distributed systems

Interoperable
  • address system-of-system issues, such as emergent properties

Evolution
  • handle changes in technology while keeping the program operational and interoperable

SOAs and Web services standards are a natural fit for net-centric solutions because of their positive quality attributes. However, they bring with them negative attributes that complicate implementation.
## LTS Assessment, Including Net-Centric Objectives

<table>
<thead>
<tr>
<th>LTS Capabilities</th>
<th>Version</th>
<th>SOA Quality Attributes (SEI Technical Note)</th>
<th>Web Services Maturity and Impact (Authors’ Analysis)</th>
<th>NESI Enterprise Technology Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add New Services</td>
<td>Version 1</td>
<td>Interoperability(+)</td>
<td>WS-Discovery(-)</td>
<td>Capability On Demand</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Availability(.)</td>
<td>WS-BPEL(.)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>UDDI(+)</td>
<td></td>
</tr>
<tr>
<td>World-Wide, Multiple</td>
<td>Version 1</td>
<td>Extensibility(+)</td>
<td>WSDL(+)</td>
<td>Distributed Operations</td>
</tr>
<tr>
<td>Services</td>
<td></td>
<td>Scalability(.)</td>
<td>ASAP(-)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Performance(-)</td>
<td>WS-Transfer(.)</td>
<td></td>
</tr>
<tr>
<td>Assorted Functionality</td>
<td>Version 1</td>
<td>Adaptability(+)</td>
<td>WS-Coordination(-)</td>
<td>Customized Applications</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Modifiability(+)</td>
<td>WS-Context(-)</td>
<td></td>
</tr>
<tr>
<td>World-Wide, Multiple</td>
<td>Version 1</td>
<td>Interoperability(+)</td>
<td>WS-Trust(-)</td>
<td>Multi-user Access</td>
</tr>
<tr>
<td>Users</td>
<td></td>
<td>Availability(.)</td>
<td>UDDI(+)</td>
<td></td>
</tr>
</tbody>
</table>
## LTS Assessment, Including Net-Centric Objectives

<table>
<thead>
<tr>
<th>LTS Capabilities</th>
<th>Version</th>
<th>SOA Quality Attributes (SEI Technical Note)</th>
<th>Web Services Maturity and Impact (Authors’ Analysis)</th>
<th>NESI Enterprise Technology Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linking and Dialects</td>
<td>Version 2</td>
<td>Adaptability(+)</td>
<td>WS-BPEL(.)</td>
<td>Customized Delivery</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Operability and Deployability(.)</td>
<td>WS-Policy(-)</td>
<td></td>
</tr>
<tr>
<td>Auditing and Security</td>
<td>Future</td>
<td>Auditability(-)</td>
<td>WS-Policy(-)</td>
<td>Assured Sharing</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Reliability(.)</td>
<td>WS-Security(-)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Security(-)</td>
<td>WS-Trust(-)</td>
<td></td>
</tr>
<tr>
<td>New Features</td>
<td>Version 2</td>
<td>Testability(-)</td>
<td>WS-Policy(-)</td>
<td>Incremental Upgrade</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Extensibility(+)</td>
<td>WS-BPEL(.)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>UDDI(+)</td>
<td></td>
</tr>
<tr>
<td>Share Translations</td>
<td>Future</td>
<td>Usability(-)</td>
<td>SOAP(+)</td>
<td>Data Exchange</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Performance(-)</td>
<td>WS-Reliability(-)</td>
<td></td>
</tr>
</tbody>
</table>
Summary

We need a method to systematically assess the appropriateness of evolving technologies.

- Technologies change frequently, therefore the decisions based on technology should be reviewed regularly.

Quality attributes constitute a key dimension of technology assessments.

- For the LTS example, we assessed the impact and maturity dimensions.

Assess Web services standards regularly to reduce risk.

- Apply this assessment tool and the associated process to start, then tailor each to meet programs’ needs.
For More Information

Acquiring Evolving Technologies: Web Services Standards

Harry L. Levinson
Liam O'Brien

Technical Note
CMU/SEI-2006-TN-001

http://www.sei.cmu.edu/publications/documents/06.reports/06tn001.html
Contact Information

Harry L. Levinson
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213
412-268-4148
hll@sei.cmu.edu

Acquisition Support Program
http://www.sei.cmu.edu/programs/acquisition-support/