Web Service Usage Scenarios

Jorgen Thelin
Chief Scientist
Cape Clear Software Inc.

Current Usage Scenarios

Web Services – Current Usage Scenarios

- Enterprise Application Integration
- Re-use of Existing Business Logic
- Deploying Applications across Firewalls
- EJB Component Reuse
- Ad-hoc Reuse

Enterprise Application Integration

- Usage Scenario Description:
  - SOAP can be used to integrate Java and EJBs with logic deployed in other enterprise systems such as CORBA and .NET.
  - The best initial projects for Web Services in organizations often involve the reuse of existing back-end systems – with Web Services used to expose them in a new way.
  - This approach has the added benefit that the focus of the project has been the Web Services rather than developing some new business logic.
- Security Implications:
  - For internal integration, the security implications for this have tended to depend on factors such as the sensitivity of the internal information being passed around and whether the information ever moves beyond the internal firewall at any point (which can happen if, for example, branch offices are connected over the Internet).

Re-use of Existing Business Logic

- Usage Scenario Description:
  - Exposing back-end logic to multiple types of clients at the same time, such as Visual Basic and Java GUIs.
  - Many projects have an attractive value proposition for using mainstream developers (Visual Basic programmers, for example) to develop the front-end clients while reserving the EJB programmers (a relatively small percentage of the very best software developers) for developing business logic.
- Security Implications:
  - The security implications for this have tended to depend on factors such as the sensitivity of the internal information being passed around (with authentication and access control being common security solutions) and whether the information ever moves beyond the internal firewall at any point (which can happen if, for example, branch offices are connected over the Internet)
  - SSL has tended to be used in the cases where deployments have been across a combination of intranets and the Internet.

Deploying applications across firewalls

- Usage Scenario Description:
  - SOAP (when HTTP is used as the transport layer) can be used to integrate applications or clients across firewalls.
  - This has been particularly useful for projects deadlines that need to avoid the organizational issues usually involved with firewalls.
  - This also has been useful for projects that involved integrating with business partners with heterogeneous firewall security requirements.
- Security Implications:
  - The security implications of what is essentially a shortcut are often ignored due to tight deadlines.

EJB Component Reuse

- Usage Scenario Description:
  - The UDDI repository can be used by organizations to make their existing business systems available for reuse within their organizations.
  - The value proposition to organizations for such projects is not just the rapid return on investment but also new opportunities.
- Security Implications:
  - Because this is for internal use, organizations to date have been happy with the various user identification systems in the UDDI registries.

Ad-hoc Reuse

- Usage Scenario Description:
  - Web Services technology allows organizations to expose existing (business) logic for reuse in ad-hoc EAI projects.
  - This is done by generating WSDL for existing logic (typically component-based logic such as Java, CORBA, or Enterprise JavaBeans) and registering them in a UDDI registry.
  - An EAI project can then be reduced to looking up the registry for a suitable service.
    - An example is a company implementing the logic for credit card validation once, but making it available for reuse anywhere it is needed.
- Security Implications:
  - The security implications for such projects have tended to be as varied as the projects.

Emerging Usage Scenarios

Web Services – Emerging Usage Scenarios

- Point-to-point system integration
- Enterprise application integration
- Technology integration
- Business partner collaboration
- Composite business processes
- Reducing I.T. lifecycle costs
- I.T. investment protection

Point-to-point system integration

- Usage Scenario Description:
  - Web Services are ideal when 'Lite' internal integration needs exist within an organization. 'Lite' integration is the transfer of data between two or more systems. A typical scenario is when a company's employee information needs to be passed into various downstream applications.
  - The threshold, however, stands at more complex integration technology:
    - for example, transaction processing, business process automation, and so on. Web Services excels at communicating data, but currently not at operational processing.
  - When composition of business services is required in a single atomic operation with complex workflow, Web Services do not yet provide such mechanisms.
- Security Implications:
  - The security implications for such point-to-point integration projects will largely depend on factors such as the sensitivity of the internal information being passed around and whether the information ever moves beyond the internal firewall at any point (which can happen if, for example, branch offices are connected over the Internet).
  - Simple communication security technology such as SSL is usually sufficient to address the security problems here.

Enterprise application integration

- Usage Scenario Description:
  - Bridging across a complex architecture comprised of multiple systems residing on multiple platforms using different object models based on different programming languages has previously required complex and expensive EAI technology, but Web Services provides a more effective communication technology for this than traditional EAI technology.
  - However in many instances, Web Services currently lack many of the enterprise features of an EAI solution, especially around process management, transactions, administration, and so on, although this will change over time.
- Security Implications:
  - The security implications for such integration projects will probably be the most critical technical issue.
  - There are currently no standards for mapping security features across all the different possible technologies being integrated, and this is even true when using established EAI technology to some extent.
  - Web services platform products are now starting to provide a unifying security layer when integrating disparate technologies by including implementations of all the basic security features such as user authentication, access control, activity auditing and reporting that are required for enterprise applications.

Technology integration

- Usage Scenario Description:
  - One of the largest categories of usage scenarios for web services at the moment is about the integration of diverse applications build on various different implementation technologies – i.e. true technology integration.
  - This can involve such simple things are Microsoft VB clients talking to Java EJB systems – something that just 12 months ago was considered virtually impossible to achieve.

- Security Implications:
  - Crossing a technology gap such as this usually highlights a corresponding security gap that needs to be addresses also.
  - So for example, a Microsoft VB (Visual Basic) program will most likely be obtaining user identity information from the Windows ActiveDirectory system and the native NT Authentication scheme, while a Java program this VB program needs to talk to may be using JAAS (Java Authentication and Authorization Services) technology to access an LDAP repository and the EJB (Enterprise JavaBeans) declarative security system to control access.
  - Web service platforms and security product vendors typically need to address the security gap associated with the technology gap being bridged in one of two ways:
    - Use products and technology that can "map" credentials and user information between the different security schemes (e.g. mapping Windows ActiveDirectory credentials to LDAP credentials). This can obviously prove increasingly harder as the number of technologies being used increases. This is where products such as Quadrasis' EASI product can add great value in an organization.
    - Provide a unifying security layer in the web services platform that to a large extent can replace the other existing security control mechanisms.

Business partner collaboration

- Usage Scenario Description:
  - Until the introduction of Web Services standards, business partners faced a difficult task to integrate their systems. Solutions were almost always once-off, customer integrations. They were difficult to implement and difficult to maintain. Changes at either partner could easily unravel the entire system. Collaboration between multiple partners was strictly the domain of very large companies.
  - For example, a yellow-pages site may be created for automotive parts vendors. A parts-provider may thus desire to provide a Web Service to integrate their services into the marketplace through the UDDI registry.
  - Web Services offer a standards-based way for business partners to collaborate. The usual business and organizational issues will still be the substantive amount of work that is done with a new business partnership. However, a common technology framework ensures that the focus is the business benefits rather than resolving technological integration problems.
- Security Implications:
  - The key security requirement here is for standards to exist to avoid the need to implement a custom security solution for each different partner being communicated with, in the same way that the interaction technology has typically converged to SOAP and WSDL.

Composite business processes

- Usage Scenario Description:
  - Once backend services are available in a standardized manner through exposing them with XML Web Services technologies and standards like SOAP and WSDL, it makes the task of reusing these core business services in new applications and new usage scenarios significantly simpler.
  - New business processes can be created by combining together the existing business process components in innovative and exciting new ways, without having to worry about the traditional technology barriers that have hindered much of this work in the past.
- Security Implications:
  - However, this can easily lead to exactly the same sorts of problems with security gaps as found in the Technology Integration usage scenarios unless all the web services being composed utilize the same set of XML security standards. This clearly highlights the importance of mature implementations of standards that have been widely adopted in the industry.

Reducing I.T. lifecycle costs

- Usage Scenario Description:
  - There are a number of factors that make Web Services a better choice than older technologies from the perspective of lifecycle costs:
    - Web Services are comparatively cheaper to implement, lowering the investment part of any return-on-investment calculation.
    - Web Services are generally quicker to implement (assuming productivity tools like CapeStudio are used). This results in a faster time to market and lower development costs.
    - Lower ongoing maintenance and transaction costs. For example, because tools like CapeStudio automatically expose application logic without coding, changes can be implemented quickly and seamlessly.
- Security Implications:
  - The trend towards the web services platform providing the unified security policy enforcement layer also creates considerable cost savings in that using a single security system considerably reduces staff training and operations costs.

I.T. investment protection

- Usage Scenario Description:
  - By allowing the functionality of existing I.T. systems to be published and re-used through SOAP, WSDL and UDDI is considerably more cost effective than re-designing from scratch.
  - Adding a web services interface onto an existing legacy system can provide a new lease of life for the system, and take away much of the immediate pressure to replace highly complex systems immediately.
  - Using web service technology as the standardized form for publishing and re-using application services also helps to protect future I.T. investment, by providing a degree of separation between the interface definition and the underlying implementation.
- Security Implications:
  - The use of web service security standards based on XML similarly provide a level of future proofing as the implementation of this security framework can be changed while still relying on the technology-neutrality of standards based on XML communications.

Issues from Usage Scenarios

- Decisions often driven by sensitivity of info
- Crossing firewall boundaries requires extra security measures
- Unified Web Services security layer is emerging
- Credential mapping does not really scale
- Gaps in the security infrastructure need to be avoided
- The need for standards is immense