WSUI Executive

White Paper

June, 2001

Epicentric, Inc.

Executive Summary

With the explosive growth of XML Web services as the foundation for the next-generation Internet, XML-based technical and business standards such as SOAP, WSDL, and UDDI are emerging to enable companies to build applications that interoperate using standard XML messaging.   Although a good foundation for machine-to-machine communication has been built, what has been missing until now is a standard to address the "last mile", namely the integration of applications described as Web services, as embedded applications within user-facing Web sites. The Web Services User Interface (WSUI) initiative is a proposed standard that addresses the lack of a standard user-interface layer for Web services and allows Web services with SOAP-RPC, XML, or WSDL SOAP interfaces to be described as complex, multi-page end-user applications that can be easily embedded in Web sites.

The benefits of an XML-based interface to an application that make Web services an ideal machine-to-machine transport layer, i.e. separating implementation from interface and enabling diverse implementation platforms to interoperate, are leveraged by WSUI. By standardizing the display layer of Web services, vendors will be able to more easily embed their applications in Web application delivery platforms without engaging in expensive, vendor-specific implementations.

WSUI leverages existing standards and adds a single new technology, the WSUI Schema, for vendors to support, in order to easily share applications.   A WSUI component represents a single application such as a stock quote, shipping tracker, or calendar application, which can be exposed as a series of multi-page views and contain embedded workflow. By embedding a WSUI implementation, vendors of Web site publishing platforms can easily integrate WSUI services without lengthy integration efforts (or even knowledge of) third-party services and applications. Customers benefit by being able to expose custom-built applications as WSUI services to be embedded in internal Web sites as well as partner Web sites, and because WSUI separates application implementation platforms from application delivery platforms, customers will experience less of a lock-in effect on any one application development platform.

Business Problem

The objective of WSUI is to create an easily implemented standard to enable SOAP, WSDL, and XML Web services to be integrated at runtime into a user interface for the purpose of displaying a Web application to an end-user. The backers of WSUI have deep experience in integration of multiple Web applications as embedded components of other Web service publishing
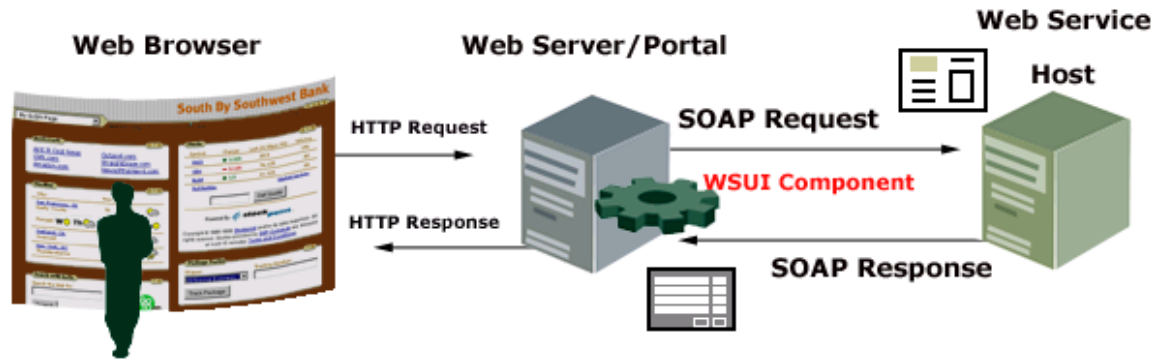
platforms. Even though some vendors have created proprietary JSP, ASP or XML SDKs to expose their application functionality to be integrated into third-party Web sites, the lack of a single standardized mechanism for exposing Web service applications has meant that most work done by vendors and customers cannot be leveraged more than once. By building a WSUI interface to their applications, application and service vendors can expose this to any platform, which supports WSUI. The business benefits are obvious, both to vendors in terms of reduced integration costs, and customers in terms of reduced vendor lock-in and reduced implementation costs.

WSUI Philosophy

The challenge creating a standard user-interface layer for Web services is to create a standard general enough that it can be implemented relatively easily on disparate server platforms such as Java, .Net, and Perl, while still being usable enough to enable real business applications to be exposed. WSUI was created in the spirit of XML, SOAP, and UDDI, and leverages the basic XML technologies that are supported in open-source toolkits such as the Apache XML project (http://xml.apache.org) as well as vendor toolkits, which implement the core XML specifications. WSUI is specifically engineered to expose components, which in turn expose any number of end-user views in a browser, typically for the purpose of supporting form and menu-driven user interaction with the back-end services. Services, which can be called by a WSUI component, are XML, SOAP RPC, and WSDL SOAP services; more service types can easily be added in the future. WSUI's creators acknowledge that many Web services will never be interacted with via a user interface; however many of the services that end-users and customers are often concerned with aggregating are indeed user-facing, which is where WSUI offers a standard mechanism for integrating them via standards-based XML messaging.

WSUI Solution Architecture

WSUI makes a few key assumptions about the interested parties. A Service Provider exposes any number of XML or SOAP Services. A Service Consumer is a Web publishing platform or portal, which is running a WSUI container implementation and constructs Web pages based on user requests. Users interact with Web pages as they normally do to interact with a WSUI Service.

Web Browser   **Web Server/Portal**   **Web Service**

South By Southwest Bank

HTTP Request →

SOAP Request →

WSUI Component

← HTTP Response

Host

← SOAP Response

The general architecture of WSUI is to assume that the services exposed via XML or SOAP exist as-is; a WSUI descriptor file plus a series of XSLT transformation templates are all that is necessary for a Web publishing platform to integrate the service as an application that can be accessed by end-users. The actual WSUI file and XSLT templates can be developed by developers working on an end-user destination site to integrate services, which already exist; the WSUI/XSLT files can also be packaged along with a standard interface by an application or service vendor to facilitate other Web sites embedding their applications.

Use Case Scenarios

Scenario #1: Shipping Tracker

Shipping trackers are a now-standard example of a Web service, which have become components of many Web sites, where the actual application logic is hosted on a remote server. A shipping tracker is typically a simple 2-stage application that is composed of a front-page form for entering a tracking number, and receiving a result view.   Alternatively, Web sites that support personalization may support a slightly more complex interaction model whereby a personalization view allows a user to enter a tracking number(s) to track, and the front-page view automatically displays the shipping status. This is an example of a Web service application that can be delivered as a hosted service over the Internet to be embedded in other Web sites using WSUI.

Scenario #2: ERP Application

An ERP application provider might wish to expose a series of modular page components to expose their functionality that can easily be embedded in other Web sites. For instance, an HR application might expose a series of views such as Clock In/Clock Out, Anniversary List, Employee Overview, and Record Working Time. Previously, exposing such functionality has been done via client-side Javascript or ActiveX plugins that do not offer much display flexibility to delivery platforms, and cannot be customized by customers or other vendors. By using WSUI, these components can easily be integrated into any WSUI-compliant Web platform. This is an example of a Web service application which

can be delivered as a packaged interface to a new or existing software product which needs to be delivered as a packaged component to be embedded in other Web sites using WSUI.

Scenario #3: Custom Enterprise Application
A custom enterprise application often needs to be integrated into portals and Web sites in other offices or operated by other units of the organization. For instance, an expense tracking application written in Cold Fusion might need to be integrated into a Website running on a Java-based platform. By creating a SOAP interface to the application and then adding a WSUI layer, all other internal Web application platforms can "subscribe" to this service dynamically, dramatically reducing the costs of integration and allowing each department to run its Web operations on a platform of its choice. This is an example of a custom-built corporate application that can be built to expose its services as a Web service application to Web sites running within a corporate network using WSUI.

Conclusion

WSUI solves the business and technical problem of easily packaging and embedding multi-stage Web applications in other Web sites using a standards-based approach which leverages existing Web services standards such as XML, SOAP, UDDI, and WSDL. By adding WSUI as a component of the Web services stack, Web application vendors, platform vendors, customers, and developers can easily practice a develop once and share many times approach to Web application integration. By increasing the ease of integration and decreasing the friction of vendor lock-in while maintaining a flexible, clean approach to creating Web services, applications, the WSUI initiative is a comprehensive framework for creation of services as well as integration of these services into a user-facing application.

WSUI offers the following concrete benefits to vendors, customers, and developers:
- Standard mechanism for packaging Web services as user-facing applications.
- Leverages existing business and technical standards including UDDI, SOAP, WSDL, XML, XSLT, and XHTML.
- Easily implemented on any robust server-side platform that supports XML.
- Open specification and implementation, available for review and use by the Web services community.