Web Services for Interactive Applications

OASIS WSIA Technical Committee

The business payoff...

- Business want to distribute web applications through *multiple distribution channels* and *multiple deployment mechanisms*.

- Business want to easily create new web applications by *leveraging* a world wide pallet of *user interface assets*. 
Producers and consumers of end-user experience
Business Scenario
Business Scenario - OEM Value Chain

Software Vendor (Visualize)
- Develop and market StockPlot software
- Ship StockPlot software on CD-ROM
- Software License based on number and type of processors

Service Producer (FAME)
- Develop and market iChartz service
- Host StockPlot service
- Consume StockPlot service
- Host iChartz service
- Subscription and/or transaction fees

Service Consumer (Divine)
- Develop and market Financial Market Insights website
- Consume iChartz service
- Host Financial Market Insights website
- Subscription and/or transaction fees

End user
- View graphs of financial data
- Access Financial Market Insights website
- N/A

Typical Revenue model

Primary activity
- Ship StockPlot software on CD-ROM
- Software License based on number and type of processors
- Subscription and/or transaction fees
- N/A
Why little focus on the UI in web services to date?

- Lack of perception of the UI as a distributed app problem
  - single-tier UI v. multi-tier value chains of UI

- Lack of focus on the need to support diversity of the UI
  - multiple combinations of “who”, “what”, “how”

- It’s a messy problem
  - XHTML, WML lack semantics of data-oriented XML
  - how to raise level of abstraction without introducing new and hard-to-get-adopted markup languages?
Our approach...

- Start by supporting the current web: opaque markup
  - simple service-oriented interface for expressing UI metadata
  - properties, actions, markup
  - aligned with remote portlet proposals (WSRP)

- Extend with page-level descriptions
  - flow composition across pages
  - data and UI state initialization

- Extend with sub-page level descriptions
  - MVC decomposition as separate services
  - XML event synchronization of data and presentation
Charter of the WSIA Technical Committee: Goals

The aims of the WSIA TC shall be as follows:

- Create an XML and web services centric framework for interactive web applications. The designs must achieve two main goals: enable businesses to distribute web applications through multiple revenue channels, and enable new services or applications to be created by leveraging existing applications across the Web.

- To harmonize WSIA as far as practical with existing web application programming models (e.g. Portals, Macromedia Flash, etc.), with the work of the W3C (e.g. XForms, DOM, XML Events, XPath, XLink, XML Component API task force), emerging web services standards (e.g. SOAP, WSDL, WSFL), and with the work of other appropriate business information bodies.

- Ensure that WSIA applications can be deployed on any tier on the network and remain target device and output markup neutral.

- Ultimately, to promote WSIA to the status of an international standard for the conduct of XML and Web Services based web application development, deployment and management.
Charter of the WSIA Technical Committee: Deliverables

- The primary deliverable of the WSIA TC is a coordinated set of XML vocabularies and Web Services interfaces that will allow businesses to:
  - Deliver web applications to end users through a diversity of deployment channels: directly to a browser or mobile device, indirectly through a portal, or by embedding into a 3rd party web application; and
  - Create web applications that can be easily modified, adapted, aggregated, coordinated, synchronized or integrated by simple declarative means to ultimately leverage a worldwide pallet of web application components.
WSIA Positioning in the Web Services standards stack

<table>
<thead>
<tr>
<th>Layer</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TPA</td>
<td>Service negotiation</td>
</tr>
<tr>
<td>UDDI</td>
<td>Publishing &amp; discovery</td>
</tr>
<tr>
<td>WSIA</td>
<td>Web Applications</td>
</tr>
<tr>
<td>WSFL, others</td>
<td>Flow Descriptions</td>
</tr>
<tr>
<td>WSDL</td>
<td>Service Descriptions</td>
</tr>
<tr>
<td>???</td>
<td>Transactions / Reliability / Routing</td>
</tr>
<tr>
<td>SOAP / XML Protocol</td>
<td>Message / Protocol</td>
</tr>
<tr>
<td>HTTP, HTTPR, SMTP, MQ</td>
<td>Transport</td>
</tr>
<tr>
<td>Internet, intranet</td>
<td>Network</td>
</tr>
</tbody>
</table>
Alignment with WSRP Remote portlet scenario

- Visual, user-facing web services including presentation.
- All Remote Portlet Web Services have exactly the same interface and therefore the same WSDL definition
  - *No* service specific UI presentation needed at intermediary applications like portals
  - Pluggable, *no* programming effort
  - Once created many channels can be reached
WSIA Summary

- OASIS TC – formed January 2002

- WSIA Web Site
  - http://www.oasis-open.org/committees/wsia/

- Related Efforts
  - XFORMS
  - WSRP