OpenSocial Update and Q&A

Dan Peterson, Evan Martin, Charlie Jiang
The Trouble with Developing Social Apps

Which site do I build my app for?
Let’s work on that…

Using OpenSocial, I can build apps for all of these sites!
What’s offered by OpenSocial?

• Activities
  – What are people up to on the web

• People/Profile Info
  – Who do I know, etc.

• Persistent datastore
  – Handles key/value pairs
Today: 350 Million User Reach
Where is OpenSocial live today?

**Live to Users:**
- MySpace
- orkut
- Hi5
- Freebar
- Friendster
- Webon from Lycos
- IDtail
- YiQi

**Live Developer Sandboxes:**
- Hyves
- iGoogle
- imeem
- Netlog
- CityIN
- Tianya
- Ning
- Plaxo Pulse
- Mail.ru
- *Yahoo (here at Hack Day)*

**Individual Developer Links:**
OpenSocial “Containers”
What’s in OpenSocial?

• JavaScript API
• REST Protocol
• Templates
• Proxied HTML content
OpenSocial’s JavaScript API

• OpenSocial JS API
• Gadget JS API
• Gadget XML schema

• OpenSocial v0.7 is live
• OpenSocial v0.8 is being deployed now

• Specs and release notes: [http://opensocial.org](http://opensocial.org)
OpenSocial’s REST Protocol

- Access social data without JavaScript
- Works on 3rd party websites / phones / etc
- Uses OAuth to allow secure access
- Open source client libraries in development
  - Java, PHP, Python, <your fav language here>

- Being deployed with OpenSocial v0.8

- Spec’s available at http://opensocial.org
OpenSocial Templates

• Writing JavaScript is hard
• Writing templates is easy
• Templates also give
  – Consistent UI across containers
  – Easy way to localize
  – More interesting content options when inlining into container (activities, profile views)
  – Ability to serve millions of dynamic pages per day without a server
Try out templates today!

- Samples and docs: [http://ostemplates-demo.appspot.com](http://ostemplates-demo.appspot.com)
- Sample app: [http://ostemplates-demo.appspot.com/friends.html](http://ostemplates-demo.appspot.com/friends.html)
- Discussion group: [http://tech.groups.yahoo.com/group/os-templates/](http://tech.groups.yahoo.com/group/os-templates/)
- Code is all in Shindig, can download, use, and even submit patches to improve
- So…
  - Get involved and provide comments, and
  - Build some apps
Proxied HTML content

- HTML and OSML pages served from your own servers
- Proxied through OpenSocial container for social goodness
- Yahoo! leading the way with Y!AP
- Proposed OpenSocial 0.9 feature
OpenSocial is what you make it.

- OpenSocial is an open source project.
- The spec is controlled by the community.
- Anyone can contribute and have a voice.
- [http://groups.google.com/group/opensocial/](http://groups.google.com/group/opensocial/)
  - “OpenSocial and Gadgets spec” subgroup

- OpenSocial Foundation
  - Get involved to nominate and elect board reps
  - [http://www.opensocial.org(opensocial-foundation/](http://www.opensocial.org(opensocial-foundation/)}
A note on compliance

• OpenSocial is designed for many sites

• Building an app:
  – Technology
  – Policy

• OpenSocial Compliance Tests
OpenSocial Compliance test in orkut

<table>
<thead>
<tr>
<th>Test Suite</th>
<th>Passed</th>
<th>Failed</th>
<th>Warnings</th>
<th>Unverified</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activity Test Suite</td>
<td>5</td>
<td>0</td>
<td>12</td>
<td>0</td>
<td>17</td>
</tr>
<tr>
<td>AppData Test Suite</td>
<td>6</td>
<td>5</td>
<td>0</td>
<td>9</td>
<td>20</td>
</tr>
<tr>
<td>Environment Test Suite</td>
<td>11</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td>21</td>
</tr>
<tr>
<td>gadgets.io:* Test Suite</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td>20</td>
</tr>
<tr>
<td>gadget.Prefs Test Suite</td>
<td>15</td>
<td>2</td>
<td>0</td>
<td>17</td>
<td>44</td>
</tr>
<tr>
<td>gadgets.util:* Test Suite</td>
<td>11</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>MakeRequest Test Suite</td>
<td>9</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>9</td>
</tr>
</tbody>
</table>

**Test Results**

- **[MKRT001]** (P2): `makeRequest()`, html, default: PASS [1059 ms]
- **[MKRT001]** (P2): `makeRequest()`, html_default: PASS (got [])
- **[MKRT001]** (P2): `dataResponse not null`: PASS (got ['text': 'sample', 'errors': [], 'data': 'sample'])
- **[MKRT001]** (P2): `dataResponse.data not null`: PASS (got 'sample')
- **[MKRT001]** (P2): `dataResponse.text not null`: PASS (got 'sample')
- **[MKRT001]** (P2): `Result is of ContentType TEXT`: PASS (got true)
- **[MKRT001]** (P2): `response.data`: PASS (got 'sample')
- **[MKRT001]** (P2): `response.text`: PASS (got 'sample')

- **[MKRT002]** (P2): `makeRequest()`, xml, default: PASS [1064 ms]
- **[MKRT002]** (P2): `makeRequest()`, xml_default: PASS [1064 ms]
- **[MKRT002]** (P2): `xml`: PASS [1064 ms]
- **[MKRT002]** (P2): `xml - DOM content-type`: PASS [1073 ms]
### OpenSocial Compliance Matrix

http://opensocial-compliance.appspot.com

<table>
<thead>
<tr>
<th>Priority</th>
<th>Environment Test Suite</th>
<th>hi5.com</th>
<th>friendster.com</th>
<th>myspace.com</th>
<th>google.com/lg</th>
</tr>
</thead>
<tbody>
<tr>
<td>P9</td>
<td>Environment.supportsField(ACTIVITY, MEDIA, ITEM)</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
</tr>
<tr>
<td>P9</td>
<td>Environment.supportsField(PERSON)</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
</tr>
<tr>
<td>P9</td>
<td>Environment.supportsField(ORGANIZATION)</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
</tr>
<tr>
<td>P9</td>
<td>Environment.supportsField(ADDRESS)</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
</tr>
<tr>
<td>P9</td>
<td>Environment.supportsField(PHONE)</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
</tr>
<tr>
<td>P9</td>
<td>Environment.supportsField(URL)</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
</tr>
<tr>
<td>P9</td>
<td>Environment.supportsField(EMAIL)</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
</tr>
<tr>
<td>P9</td>
<td>Environment.supportsField(NAME)</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
</tr>
<tr>
<td>P9</td>
<td>Environment.supportsField(ACTIVITY)</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
</tr>
<tr>
<td>P2</td>
<td>Environment.supportsField(PERSON) - Empty fields</td>
<td>11 tests</td>
<td>10 tests</td>
<td>11 tests</td>
<td>10 tests</td>
</tr>
</tbody>
</table>

### gadgets.io.* TestSuite

<table>
<thead>
<tr>
<th>Priority</th>
<th>Test cases</th>
<th>hi5.com</th>
<th>friendster.com</th>
<th>myspace.com</th>
<th>google.com/lg</th>
</tr>
</thead>
<tbody>
<tr>
<td>P9</td>
<td>gadgets.js.encodeValues(String) - With a String object</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
</tr>
<tr>
<td>P9</td>
<td>gadgets.js.encodeValues(()) - With a simple map</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
</tr>
<tr>
<td>P9</td>
<td>gadgets.js.encodeValues([[]]) - With reserved characters</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
</tr>
<tr>
<td>P9</td>
<td>gadgets.js.encodeValues(()) - Map with H18N values</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
</tr>
<tr>
<td>P9</td>
<td>gadgets.js.encodeValues(([])) - With unsafe characters</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
</tr>
<tr>
<td>P9</td>
<td>gadgets.js.getProxyURI(String) - With a String object</td>
<td>Failed</td>
<td>Failed</td>
<td>Failed</td>
<td>Failed</td>
</tr>
<tr>
<td>P9</td>
<td>gadgets.js.getProxyURI(String) - With valid URL</td>
<td>Failed</td>
<td>Failed</td>
<td>Failed</td>
<td>Failed</td>
</tr>
<tr>
<td>P2</td>
<td>gadgets.js.encodeValues([]) - With undefined parameter.</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
</tr>
<tr>
<td>P2</td>
<td>gadgets.js.encodeValues(null) - With null parameter.</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
<td>Pass</td>
</tr>
<tr>
<td>P2</td>
<td>gadgets.js.encodeValues() - With no parameters.</td>
<td>10 tests</td>
<td>8 tests</td>
<td>10 tests</td>
<td>8 tests</td>
</tr>
</tbody>
</table>

### gadget.Prefs Test Suite

<table>
<thead>
<tr>
<th>Priority</th>
<th>Test cases</th>
<th>hi5.com</th>
<th>friendster.com</th>
<th>myspace.com</th>
<th>google.com/lg</th>
</tr>
</thead>
<tbody>
<tr>
<td>P9</td>
<td>gadget.Prefs - getFloat()</td>
<td>Failed</td>
<td>Failed</td>
<td>Failed</td>
<td>Failed</td>
</tr>
<tr>
<td>P9</td>
<td>gadget.Prefs - getInt()</td>
<td>Failed</td>
<td>Failed</td>
<td>Failed</td>
<td>Failed</td>
</tr>
<tr>
<td>P9</td>
<td>gadget.Prefs - getArray(key)</td>
<td>Failed</td>
<td>Failed</td>
<td>Failed</td>
<td>Failed</td>
</tr>
<tr>
<td>P9</td>
<td>gadget.Prefs - prefs set - set string with pipe as array</td>
<td>Passed</td>
<td>Passed</td>
<td>Passed</td>
<td>Passed</td>
</tr>
</tbody>
</table>
A note for (potential) “containers”

• Apache Shindig
  – Reference implementation for OpenSocial
    • Production quality
  – Open source
    • Java
    • PHP
  – Compliant with OpenSocial v0.8 (0.8.1 too)
    • Release planned for September 30th

• Partuza
  – Social networking site frontend
  – Open source, PHP
Other comments

- Portable Contacts Alignment
- Caja for JavaScript security
Current State of Yahoo’s OpenSocial Support

Charlie Jiang
Supported Features

- OpenSocial JavaScript APIs
- Gadget Core APIs
- OAuth to Y! Services
Not Available Today

• Gadget XML Definition
  – Hack: You can put the following url in your App Base url to play around.
    • http://gadget.apps.yahoo.com/postproxy.php?url=${gadget_url}

• OpenSocial RESTful APIs
Upcoming Improvements

- An Extension to Support Proxied HTML
- An Extension to Support setSmallView()
- YML & OSML
- YQL
- Caja & 3\textsuperscript{rd} Party Libraries
Discussion