The Coming Functionality Mashup

Dr. Charles Severance
University of Michigan
IMS Technical Advisory Board co-Chair

http://www.dr-chuck.com/
csev@umich.edu
Outline

- My guest lecturer for my Ruby class got sick yesterday
- Short Sakai Update
- Content Mashup
- Functionality Mashup
- Standards and Technology for Functionality Mashup
- The Tool Interoperability Project
- Thoughts on University Campus Under Construction
- Summary
Sakai Update (quick)
Sakai Foundation Overview

Collaboration, Teaching, and Learning

FOSS – 100% free to use, modify and contribute

Sakai is 3.5 years old

100+ people developing and testing Sakai releases

Non-profit Sakai Foundation since January 2006

100+ Higher Education and 15 company members

Six paid staff members

Overview Video: http://www.dr-chuck.com/media.php?id=64

Michael Korcuska
Executive Director
www.sakaiproject.org
mkorcuska@sakaifoundation.org
Sakai Copyright

July 20, 2006 – Met in Barcelona with Dr. Valverde and others from OUC and lafarga.cat – discussed UOC concerns about Sakai’s ECL 1.0 license

October 19-20, 2006 – Intellectual Property Summit at Indianapolis IN, USA – attended by Malcolm Bain of UOC/lafarga.cat – The meeting produces ECL 2.0 based on Apache 2.0

April 2007 – ECL 2.0 approved by the Open Source Initiative (www.opensource.org)

September 2007 – Sakai Foundation Board switches to ECL 2.0 for Sakai version 2.5
Content Mashup
Content Mashup

- Read only – public materials
- Publish / subscribe
- Push content
- A highly personalized “newspaper replacement”
Functionality is interactive, functionality usually needs identity, functionality is live software. How do we mash this up?
Map Mashup

iChoice hotels

Proprietary Glue

Mapquest
MIT Simile: Timeline Mashup

Dr. Chuck's Page / Back to map

Sakai User's Meeting
IMS Ahf-Lab
Meeting on usability and
Seth Teriault produces the
NCeSS eCollaboration
Portals and portlets v
Os Vle Next Generation
Univendat de

JISC/CNI Meeting
York, UK (URL)
Thu, 06 Jul 2006 04:00:00 GMT
Fri, 07 Jul 2006 04:00:00 GMT

Simile

Depending on browser you may or may not be able to drag the timeline around Google-style. The way to navigate that works on all browsers is to double click on some area of the timeline and that will become the new center. If dragging seems not to work (Safari for example) simply double click and be patient.

Reference: The Very Cool MIT Simile Timeline Project

Proprietary Glue
Simile Timeline Mashup Code

```html
<script src="http://simile.mit.edu/timeline/api/timeline-api.js" type="text/javascript">
</script>

var tl;
function onLoad() {

var eventSource = new Timeline.DefaultEventSource();
var bandInfos = [
    Timeline.createBandInfo(
        eventSource: eventSource,
        date: "Jun 28 2006 00:00:00 GMT",
        width: "80%",
        intervalUnit: Timeline.DateTime.WEEK,
        intervalPixels: 100
    ),
    Timeline.createBandInfo(
        eventSource: eventSource,
        showEventText: false,
        trackHeight: 0.5,
        trackGap: 0.2,
        date: "Jun 28 2006 00:00:00 GMT",
        width: "20%",
        intervalUnit: Timeline.DateTime.MONTH,
        intervalPixels: 200
    )
];
bandInfos[1].syncWith = 0;
bandInfos[1].highlight = true;

    tl = Timeline.create(document.getElementById("my-timeline"), bandInfos);
    Timeline.loadXML("events.php", function(xml, url) {
        eventSource.loadXML(xml, url);
    });

    var resizeTimerID = null;
    function onResize() {
        if (resizeTimerID == null) {
            resizeTimerID = window.setTimeout(function() {
                resizeTimerID = null;
                tl.layout();
            }, 500);
        }
    }
}
```
```
MashUp: Google Analytics

<script src="http://www.google-analytics.com/urchin.js" type="text/javascript"></script>
_uaacct = "UA-423997-1";
urchinTracker();
</script>
Beyond Anonymous Functionality Mashup

- Think “portal”, “Eclipse”, even “Windows”
- There are no rich standards in this area so everyone invents their own
- Sakai is an example of a one off functionality mashup with identity and authorization used to select and assemble tools together flexibly to produce an application.
Distributed Hybrid Earthquake Engineering Experiments: Experiences with a Grid Shaking Grid... - group of 27

Distributed linear hashing and parallel projection in main memory databases... - group of 10

Grid portals: A scientist's access point for grid services (draft 1)

High Performance Computing
Approaching Functionality Mashup Standards and Technologies

Standards are critical to making functionality mashup something we can commonly use to assemble applications.
Functionality Mashup

What is the standards equivalent of “RSS” for functionality mashup?
Background in Standards

- Open Software Foundation (OSF) (1989–90)
- UNIX International (1990)
- IEEE POSIX
  - Vice Chair POSIX (1992–1996)
- IEEE Standards Advisory Board
- IMS
  - Technical Board coChair 2005–2007
  - IMS Common Cartridge WG – 2006
Conflict and Consensus: The Role of Standards

Charles Severance, Michigan State University

As one organization "invented" the turf of another, we get to see the cards held in the hands of the players.

THE PAST

While we certainly need to practice listening about these conflicts, we must try to remember how serious these conflicts seemed during their time. One of the most serious technology wars began in IBM's inhouse efforts to promote the Data General Corporation's (DG) Rangeley and TANDEM computers. In the early 1980s, both companies pushed to standardize software interfaces that would allow different systems to work together. The battle over these standards was a major issue during this time.

INTERVIEW: The Role of Standards

**LEONARD COHEN:** What are some of the key conflicts you've seen in the past 20 years? One of them was the battle over the Samba project.

**CHARLES SEVERANCE:** The Samba project was a key conflict because it involved issues of interoperability. The project was started to allow Windows NT and Unix systems to work together. The conflict was over the standards used to implement these systems.

**COHEN:** What were the main issues in the Samba conflict?

**SEVERANCE:** The main issue was the use of NetBIOS, a protocol for sharing resources over a network. NetBIOS is inefficient and not as secure as newer protocols. The conflict was over whether to use NetBIOS or switch to a newer protocol.

**COHEN:** How did the Samba project resolve these conflicts?

**SEVERANCE:** The Samba project eventually resolved the conflict by adopting a protocol that was compatible with both NetBIOS and newer protocols. This allowed the project to continue and the Samba project has since become a widely used standard.

**COHEN:** What lessons can we learn from the Samba project as we look to the future of standards?

**SEVERANCE:** The Samba project shows the importance of flexibility and the need to adapt to changing needs. It also highlights the need for open standards and the benefits of collaboration among different groups.

**COHEN:** What is your prediction for the future of standards?

**SEVERANCE:** As technology continues to evolve, we can expect more conflicts and controversies over standards. The key will be to find a balance between innovation and compatibility. It will be important to involve all stakeholders and find common ground for everyone.

**COHEN:** Thank you, Charles, for your insights. Your views are valuable as we look towards the future of standards.

**SEVERANCE:** Thank you, Leonard. It's been a pleasure discussing these issues with you.

---

**LEONARD COHEN:** What are some of the key conflicts you've seen in the past 20 years? One of them was the battle over the Samba project.

**CHARLES SEVERANCE:** The Samba project was a key conflict because it involved issues of interoperability. The project was started to allow Windows NT and Unix systems to work together. The conflict was over the standards used to implement these systems.

**COHEN:** What were the main issues in the Samba conflict?

**SEVERANCE:** The main issue was the use of NetBIOS, a protocol for sharing resources over a network. NetBIOS is inefficient and not as secure as newer protocols. The conflict was over whether to use NetBIOS or switch to a newer protocol.

**COHEN:** How did the Samba project resolve these conflicts?

**SEVERANCE:** The Samba project eventually resolved the conflict by adopting a protocol that was compatible with both NetBIOS and newer protocols. This allowed the project to continue and the Samba project has since become a widely used standard.

**COHEN:** What lessons can we learn from the Samba project as we look to the future of standards?

**SEVERANCE:** The Samba project shows the importance of flexibility and the need to adapt to changing needs. It also highlights the need for open standards and the benefits of collaboration among different groups.

**COHEN:** What is your prediction for the future of standards?

**SEVERANCE:** As technology continues to evolve, we can expect more conflicts and controversies over standards. The key will be to find a balance between innovation and compatibility. It will be important to involve all stakeholders and find common ground for everyone.

**COHEN:** Thank you, Charles, for your insights. Your views are valuable as we look towards the future of standards.

**SEVERANCE:** Thank you, Leonard. It's been a pleasure discussing these issues with you.

---

**COHEN:** What are some of the key conflicts you've seen in the past 20 years? One of them was the battle over the Samba project.

**SEVERANCE:** The Samba project was a key conflict because it involved issues of interoperability. The project was started to allow Windows NT and Unix systems to work together. The conflict was over the standards used to implement these systems.

**COHEN:** What were the main issues in the Samba conflict?

**SEVERANCE:** The main issue was the use of NetBIOS, a protocol for sharing resources over a network. NetBIOS is inefficient and not as secure as newer protocols. The conflict was over whether to use NetBIOS or switch to a newer protocol.

**COHEN:** How did the Samba project resolve these conflicts?

**SEVERANCE:** The Samba project eventually resolved the conflict by adopting a protocol that was compatible with both NetBIOS and newer protocols. This allowed the project to continue and the Samba project has since become a widely used standard.

**COHEN:** What lessons can we learn from the Samba project as we look to the future of standards?

**SEVERANCE:** The Samba project shows the importance of flexibility and the need to adapt to changing needs. It also highlights the need for open standards and the benefits of collaboration among different groups.

**COHEN:** What is your prediction for the future of standards?

**SEVERANCE:** As technology continues to evolve, we can expect more conflicts and controversies over standards. The key will be to find a balance between innovation and compatibility. It will be important to involve all stakeholders and find common ground for everyone.

**COHEN:** Thank you, Charles, for your insights. Your views are valuable as we look towards the future of standards.

**SEVERANCE:** Thank you, Leonard. It's been a pleasure discussing these issues with you.
Grab the Chance to Work on the Leading Edge

Charles Severance, Michigan State University

The possibilities for career growth through standards are endless.

With the information I learned at each meeting, I was able to come back to my job with new perspectives on the technical issues I was facing. At times, my clients thought I had a crystal ball on my head, so to speak. But in the long run, it was well worth it. The bottom line is that standards are not the end, but the beginning.

S

OSI Retrospect and Prospect

Jerry Foley, Ecom Corporation

Charles Severance, Michigan State University

As a recent IEEE meeting, I spoke up a conversation about OSI networking with Jerry Foley. As we talked, it occurred to me that enough time has passed to talk openly about the myths and realities regarding the ISO OSI networking effort and how those relate to TCP/IP networking as the two technologies converged over the late 1970s and 1980s. In late 1979, working within the ANSI Information Systems Infrastructure, Foley was appointed to write a study on “standards for distributed information systems.” From this study ANSI’s Open Systems Interconnection (OSI) committee resulted, and for the next 10 years Foley served on the committee about 12 of them as chair. Foley was also OSI OBI (OSI Body of Information) and he represented the US State Department to the CCITT (now ITU) work on ISO. The Inter-society committee is the Manufacturing Automation Protocol, which implemented OSI and managed MAPP implementations in Central American plants.

Charles Severance

We must keep in mind that OSI research work was driven by the need to be in the lead in networking, but when it was time to deploy OSI, there were many problems. For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

OSI was a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.

For example, OSI and TCP/IP are not as different as people believe. OSI is a multi-layered protocol stack, with each layer having a specific function. TCP/IP is a single-layer protocol stack, with each function contained in a single layer.

Nevertheless, OSI was a significant step forward in networking technology, and it helped to establish the standards that are still in use today.
Functionality Mashup Technical Needs

Aggregating System

- Discovery
- Preferences
- Identity
- Context
- Roles/Attrib
- Markup
- Storage
- Services

Provisioning
Placement
Run-time

Tool / Capability / Code
How IMS Tool Interoperability 1.0 Works

1. LMS System
2. Sakai IMS Proxy
3. Samigo, ConceptTutor, Etc
4. Session And Services Bootstrap
5. Sakai APIs
6. Sakai
7. IMS TI Outcome Request

External Tool

Application Code

Launch

Outcome
<table>
<thead>
<tr>
<th>Time</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>8 AM</td>
<td></td>
</tr>
<tr>
<td>9 AM</td>
<td></td>
</tr>
<tr>
<td>10 AM</td>
<td></td>
</tr>
<tr>
<td>11 AM</td>
<td></td>
</tr>
<tr>
<td>12 PM</td>
<td></td>
</tr>
<tr>
<td>1 PM</td>
<td></td>
</tr>
<tr>
<td>2 PM</td>
<td></td>
</tr>
</tbody>
</table>

**OXIDATIVE PHOSPHORYLATION**

The enzymatic phosphorylation of ADP to ATP coupled to electron transfer from a substrate to molecular oxygen. Oxidative phosphorylation has the following attributes:

- Occurs in the mitochondrion (and some other compartments).
- Electrons are passed down the electron-transfer chain.
- Redox reactions of the electron-transfer chain move protons to the intermembrane space.
- Phosphorylation of ADP is coupled to re-entry of protons into the matrix through the enzyme ATP synthase.
Sakai can consume IMS TI tools *and* produce / export its tools over IMS Tool Interoperability.

A Sakai calendar can be shown in a portal using this approach.
OXIDATIVE PHOSPHORYRATION

The enzymatic phosphorylation of ADP to ATP coupled to electron transfer from a substrate to molecular oxygen. Oxidative phosphorylation has the following attributes:

- occurs in the mitochondrion (and some other compartments)
- electrons are passed down the electron-transfer chain
- redox reactions of the electron-transfer chain move protons to the intermembrane space
- phosphorylation of ADP is coupled to re-entry of protons into the matrix through the enzyme ATP synthase
Looking Forward IMS Tool Interoperability

- IMS Tool Interoperability 1.0 Extensions based on use
  - REST and SOAP–Lite Bindings
  - Security Function and Configuration Extensions
  - Outcome Request schema – IMS TI 1.1

- IMS Learning Tool Interoperability 2.0
  - Builds on IMS TI 1.0
  - Improves the integration into LMS systems
  - Defines extension points within LMS systems such as “Add New Resource”
  - Modeled on Blackboard Building Blocks
  - Led by: Bruno van Haetsdale of Wimba and Chris Moffat of Microsoft
Tool Interoperability Research

One of the primary goals of the Sakai project was to define a “Tool Portability Profile” which made it possible to build truly portable tools that could be moved between Learning Management Systems.
One of the primary goals of the Sakai project was to define a “Tool Portability Profile” which made it possible to build truly portable tools based on OKI OSIDs that could be moved between Learning Management Systems.
One of the primary goals of the Sakai project was to define a “Tool Portability Profile” which made it possible to build truly portable tools based on OKI OSIDs that could be moved between Learning Management Systems.

While the Sakai project is very successful and achieved many goals, there is work yet to be done to complete this goal.
One of the primary goals of the Sakai project was to define a “Tool Portability Profile” which made it possible to build truly portable tools based on OKI OSIDs that could be moved between Learning Management Systems.

While the Sakai project is very successful and achieved many goals, there is work yet to be done to complete this goal.

This is not just about Sakai – it is about enabling the general purpose mash up of learning functionality.
My tool interoperability research intends to deliver portable and interoperable tools. For me IMS Tool Interoperability was the missing standard to build tools to enable language independence and cross-platform interoperability.

### Tool Producers – Tool Functionality

- **Ruby Tool**
  - Session

- **Java Servlet**
  - Session

- **PHP Tool**
  - Session

- **Perl Tool**
  - Session

__IMS TI__

---

### Tool Consumers (LMS Systems)

- **Sakai**
- **Moodle**
- **Angel**
- **Portal (168)**
- **BlackBoard**
- **SharePoint**
OKI OSIDs are a set of interoperable APIs that can provide run-time services for these tools. We can use the IMS TI ability to establish a session remotely to partially populate an OSID Context for each user.

**Tool Producers – Tool Functionality**

- **Ruby Tool**
  - Session
  - OSIDs

- **Java Servlet**
  - Session
  - OSIDs

- **PHP Tool**
  - Session
  - OSIDs

- **Perl Tool**
  - Session
  - OSIDs

**Tool Consumers (LMS Systems)**

- Sakai
- Moodle
- Angel
- Portal (168)
- BlackBoard
- SharePoint

**IMS TI**
My Spare Time Goals

• Encourage the adoption and use of IMS Tool Interoperability and OKI OSIDs

• Enable learning tools to be built in any programming language or web application framework

• Participate in the IMS and OKI standards activities to evolve and enhance these standards

• Build and distribute reference implementations of the IMS TI producer and consumer modules in many languages
IMS TI Progress To Date

- IMS TI Consumers
  - Sakai Tool
  - JSR-168 Portlet (Pluto, uPortal, Oracle Portal, etc)
- IMS TI Sample Producers
  - Java
  - Ruby
  - PHP
- IMS TI 1.0 Extensions
  - SOAP-Lite and REST Bindings – Makes non-Java producers practical
  - Configuration and Permission functionality
Using OKI OSIDs in Portable Tools

• One of the great challenges in using OKI OSIDs is one of “bootstrapping”

• The OSID Context contains the information that makes the rest of the OSIDs function – it gives them “context”

• But – “Where does the OSID Context come from?”

• A Chicken and Egg Problem – which came first?

• IMS Tool Interoperability provides a mechanism to populate the OSID Context as part of the Launch Request/Response and session setup
OKI Progress to Date

- Proof of concept of IMS TI populating OSID Context
- Started discussions regarding a Ruby Binding for OKI OSIDs
IMS Tool Interoperability - Java Test Tool

This is an IMS Tool Interoperability test end point written in Java. It only supports REST profile of the IMS Tool Interoperability Specification. REST supports is currently an extension (i.e., not part of the approved IMS TI standard). It uses the information provided in the IMS Tool Interoperability LaunchRequest to populate and OKI Osid Context.

Testing OKI...

guest.getDisplayName() = Ballesté Crevillén, Alex

At this point the tool is just a prototype to demonstrate feasibility and explore potential extensions to IMS Tool Interoperability 1.0 to better support tools.

Here is the XML from the IMS Tool Interoperability LaunchRequest which created this session. This work is also exploring approaches to develop Java implementations for OKI OsIDs to be used in Campus Project.

<?xml version="1.0" encoding="UTF-8"?>
<LaunchRequest>
<DeploymentProfile>
IMS Tool Interoperability - PHP Test Tool

This is an IMS Tool Interoperability test end point written in PHP. It only supports REST profile of the IMS Tool Interoperability Specification. REST support is currently an extension (i.e. not part of approved IMS TI standard). It uses the information provided in the IMS Tool Interoperability Launch Request to populate an OSID Context.

Testing OK!

agent.getDisplayName = Alex, Ballesté Crevillén

At this point the tool is just a prototype to demonstrate feasibility and explore potential extensions to IMS Tool Interoperability 1.0 to better support tools.

Here is the XML from the IMS Tool Interoperability LaunchRequest which created this session.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<LaunchRequest>
	
</LaunchRequest>
```
This is an IMS Tool Interoperability test end point written in Ruby. It only supports the REST profile of the IMS Tool Interoperability Specification. REST support is currently an extension (i.e. not part of the approved IMS TI standard). It uses the information provided in the IMS Tool Interoperability Launch Request to populate and OKI OSID Context.

Testing OKI...

agent.getDisplayName() = Charles Severance

At this point the tool is just a prototype to demonstrate feasibility and explore potential extensions to IMS Tool Interoperability 1.0 to better support tools. This work is also exploring approaches to develop Ruby bindings and implementations for OKI OSIDs.

Here is the XML from the IMS Tool Interoperability LaunchRequest which created this session:

```xml
<?xml version='1.0' encoding='UTF-8'?>
<LaunchRequest>
  <DeploymentProfile>
    <CoreSettings>
      <ProxyTool>
        <IMSTIVersion>
          <major>1</major>
          <minor>0</minor>
        </IMSTIVersion>
        <ProxyToolDescription>DO A LOGIC</ProxyToolDescription>
        <ProxyToolName>Ruby Loop Back Endpoint</ProxyToolName>
        <ProxyToolVersion>
```

IMS TI 1.0 Function and Security Extensions

<TIR>
  <LaunchService/>  <!-- Not allowed for this placement -->
  <RestLaunchService>
    <ServiceName>PHP Test End Point</ServiceName>
    <ServiceLocation>
    </ServiceLocation>
    <Options>
      <option>Width</option>
      <option>Height</option>
      <option>Color</option>
    </Options>
    <Permissions>
      <permission>Read</permission>
      <permission>Update</permission>
      <permission>Write</permission>
    </Permissions>
  </RestLaunchService>
</TIR>
This is the IMS Proxy Portlet developed for Sakai controlling configuration, roles, and functions for a particular tool placement from within the LMS. This information will be passed to the remote tool in the Launch Request.
Process Flow

Teacher sets up and configures LMS Placement using IMS TI Descriptor.

Student selects tool and a Launch Request is sent to the consumer. This contains information to build an OSID Context associated with the user session in the producer.

When the student requests markup in the context of their session, the OSID implementations pull the OSID Context from the session.
Roadmap Forward (in my spare time)

- Develop New Consumers for IMS TI 1.0 (with extensions)
  - Moodle
  - Angel
  - BlackBoard Building Block
  - Microsoft SharePoint
- New IMS TI Producer Reference Implementations
  - Perl
More Spare Time Roadmap

- Work on Ruby/Rails Binding for OKI OSIDs
- Work with targeted application providers to support adoption of IMS TI 1.0
- Eliminate iFrames
  - WSRP – Web Services for Remote Portals
  - Proxy / Bridge Portlet / Web Clipping in the IMS TI Consumer
  - Create REST Binding for WSRP
Resources Available

• So far this is just me and members of the Sakai Community
  • Universitat de Lleida
• Interested in others with resources to help
• Participating in the OKI and IMS formal processes
• There is no rush – this needs to be done right
Thoughts on University Campus Under Construction
Campus Project – In a Bodega

- Service Oriented Architecture
- Tool Pattern allows deployment in Sakai *or* Moodle
- OKI as Middleware
- Includes cross-deployment and configurations
- Proxy Tool Pattern
- Strong funding by Catalan Government
- www.campusproject.org
My Thoughts

- University Campus – A Bold Choice and Challenge
  - University Campus project chose to support both Moodle and Sakai
  - Developing a new pattern for tools that work in either Sakai or Moodle
  - Interoperability based on OKI OSIDs and an OKI Bus

- Some Ideas *
  - Broaden the impact of the UC work
  - Use and extend IMS Tool Interoperability

* Since I am an academic, I assume infinite time and infinite resources
Initial Campus OKI OSIDs

- Authentication
- Authorization
- Configuration
- Locale
- Logging
- Identifier
- Messaging

This set is a good choice because it keeps the project scope feasible. And these OSIDs are used by nearly every learning application ever built.
Campus Architecture
Adding IMS Tool Interoperability to Campus

Java / PHP Application

Session

OSID Context

Tool Runtime

OSID Implementations

Launch Request

OKI BUS

OKI Services

Moodle / Sakai Gateway

Proxy Tool

IMS TI Descriptor

Java/PHP Application

OSID Interface

OSID Implementation

Proxy Web Service

Web Services

OSID Implementation

OSID Interface

Moodle Gateway

Sakai Gateway

Moodle Platform

Sakai Platform

Oki Bus

Consumer

Provider
Adding IMS Tool Interoperability to Campus

The primary value in using IMS Tool Interoperability as part of the Campus Architecture is to provide a standards-based protocol to exchange configuration information between the LMS and the Tool and establish the OSID Context. For Campus, this OSID context can also contain information which properly configures the OKI bus so that applications can access other OKI services in the LMS.
Sharing Tool Context

We can support multiple LMS systems and tools that federate identity by including an appropriate bus end point in each launch request.

Trust is granted when two LMS placements share the IMS TI Descriptor.
Campus Architecture

Gateway de Sakai
Sakai

Gateway de Moodle
Moodle
Summary

Standards are emerging to enable functionality mashup – particularly IMS standards for learning functionality mashup.

Much more work is required. With several strong open source LMS systems in the market place, we can begin to tackle the hard technical issues in real Tool Interoperability.

This will require an iterative process of building reference implementations, extending standards, and then improving standards.
Thank you for your time...

- csev@umich.edu
- http://www.dr-chuck.com/