uPortal Developers Meeting Minutes
University of British Columbia
March 26-29, 2001

Attendees

Susan Bramhall – Yale University
George Lindholm, Paul Zablosky, Wilson Lo, Bruce Jolliffe, Dave Frazer, Zed Shaw,
Doug Loewen, Vikrant Joshi – University of British Columbia
John Laker, Jim Farmer – University of Delaware
Justin Tilton – Instructional Media + Magic, Inc.
Ken Weiner*, Peter Kharchenko, Bernie Durfee – IBS
Bill Brooks, Cal Poly San Luis Obispo
Andrew Draskoy, Memorial University of Newfoundland
Greg Barnes, University of Washington
Ann Badertscher, Sigma Systems

* Minute recorder

Meeting purpose

During the previous meeting at the University of Delaware, the uPortal team decided to
release uPortal 2.0 by May 13, 2001, just in time for the CUMREC conference in
Phoenix, Arizona. Discussion at this month's meeting included the resolution of uPortal
2.0 technical issues as well as lessons learned from UBC's experiences with
administrating a live implementation of uPortal 1.0.

Announcements

- [Cal Poly San Luis Obispo](#) plans to launch *My CalPoly*, an implementation of
  uPortal 1.5, on June 25, 2001. *My CalPoly* will initially be targeted towards a
  student body of 16,000.

- Two more uPortal training sessions have been scheduled:

  1) College of the Holy Cross, Worcester, MA, April 2-4, 2001
  2) Villanova University, Villanova, PA, May 22-24, 2001

  Each of these classes will be taught by Ken Weiner and Bernie Durfee of IBS. For
  more info, see http://www.interactivebusiness.com/uPortal_Training.htm

Since February

The following is a summary of progress since the previous meeting:

Holy Cross progress – Bernie Durfee

Bernie gave the group a peek at the current state of MyHC, as it prepares to go live within the next few weeks. The first phase of MyHC includes services for students. They are in the final stages of testing. Authentication is via LDAP. Roles are managed via uPortal's reference implementation. Channels include UBC's IMAP email channel, class schedule and course guide browsing, and grade reporting. The channels are all "thin" clients to EJBs running on the IBS Enterprise Application Server (EAS).

Publishing channel – John Laker

John presented the XML that a channel author would have to write if he/she wanted the channel to be publishable. The group decided to refer to this document as a "channel declaration". The document contains information for prompting the user for channel parameters. It also controls which of these parameters can be specified or overwritten at subscribe time, and which can be edited by the user after the channel has been subscribed to.

DbLoader tool – Ken Weiner

Until now, a set of database scripts were included with uPortal. Because of syntactical differences in different databases, separate scripts had to be maintained for each database. For example, there was a createTables-hypersonicsql.script, createTables-oracle.script, etc. It was too difficult to maintain all of these scripts. Ken presented a new tool for uPortal developers and administrators to use for populating a relational database with uPortal's table schema and "starter" data. This tool, called DbLoader, uses database metadata to map generic database types to local database types. For example, the generic type LONGVARCHAR would be converted to TEXT for MS SQL Server and LONG for Oracle. To use this tool, run org.jasig.portal.tools.DbLoader and set the portal base directory via a system parameter. For example,
`java –Dportal.home=/usr/local/myPortal –classpath … org.jasig.portal.tools.DbLoader`
DbLoader reads the definition of tables from a file called tables.xml and then generates DROP TABLE and CREATE TABLE statements. Next it reads data.xml and generates INSERT statements. See the JavaDoc in Dbloader.java for more instructions. With DbLoader, there will no longer be a need for the database scripts and they will be removed shortly. If scripts are needed, DbLoader will be able to generate them.

Error channel – Peter Kharchenko

A new channel has been written which gets instantiated in place of a channel that reports an error or a channel that is determined to be unrenderable by the framework. Channel authors now have the ability to throw PortalExceptions. PortalExceptions that channels can throw include:

GENERAL_RENDERING_EXCEPTION
INTERNAL_TIMEOUT_EXCEPTION
AUTHORIZATION_EXCEPTION
RESOURCE_MISSING_EXCEPTION

The error channel allows a user to ask the problematic channel to "Retry" (call renderXML again) or "Restart" (reinstantiate the channel). The presence of this error channel greatly increases the robustness of the framework.

Login channel – Ken Weiner

A Login channel has been added. It lets a user log in to the portal. There are only stylesheets written for web browsers right now. Stylesheets for WML and PalmHTML still need to be written before users can sign in on cell phones and Palm Pilots.

LDAP Security Provider – Ken Weiner

Ken has checked into the rel-1-0-patches branch a security provider implementation that authenticates against an LDAP directory. The code was written by Russ Tokuyama from the University of Hawaii. If there are requests from the uPortal community, the code will also be checked into 2.0.

Discussions

IChannel – The group is comfortable with the channel interface as it was defined at the last developer's meeting. There are still some things to do: change receiveEvent method to accept a more general portal event, pass SecurityContext to the channels. George Lindholm expressed a concern to have IXMLChannels work in 2.0 without having to change their interface to IChannel. As a result, the group decided to add an IXMLChannel interface that simply extends IChannel.

Authentication – Jim Farmer summarized the 3 possible ways of handling authentication as seen by Carl Jacobsen:

1) Local to the uPortal database – included with uPortal
2) Local to the institution – most common so far, requires a new SecurityProvider
3) Remote, e.g. Shibboleth

The group asked that Yale provide some examples of how to properly use the security interfaces provided as of uPortal 1.0. No one was sure how Yale intended for a channel to gain access to a security context. Also, there was some confusion as to how to properly implement chaining security contexts. These same interfaces are also included in 2.0.

Having the portal save credentials on behalf of users is an issue that came up again. Jim referred to this as a "wallet" feature. The thought of doing this makes many people cringe while others find it a reasonable solution given its effect on usability. The group decided that such a solution should be made an option as long as it comes with warnings about why it isn't a secure solution but will likely not get implemented in 2.0.

The IPerson implementation will get passed to each channel as part of ChannelRuntimeData. The IPerson interface has a method called getAttribute that will return standard user attributes such as email address and phone number. The group decided to come up with a list of standard attributes that channels may request. At authentication time, some or all of the person attributes will be obtained from a directory or some other means. IPerson would return null for all attributes not obtained during authentication. The standard list of attributes supported will be taken from EduPerson and a means for mapping these standard attribute names to local attribute names must be developed.

Database – The uPortal database needs to be further normalized. UBC is going to take a leading role in its normalization. Factories must create all the DBImpl's just as the security providers are currently created.

Paul Zablosky proposed that channels need a way to get keys for storing information on behalf of users. In some cases, it may not be sufficient to store information keyed by a user name or user ID as is done in CBookmarks. The group decided that that best way to do this is to create a key-generating service that issues keys that have no particular semantics attached to them.

Authorization & Roles – Bernie reviewed the authorization infrastructure once again. Nothing has changed since the last developers meeting.

Performance & Production Issues – UBC determined that changes were necessary in uPortal 1.0 and 1.5 in order to make it scale to more concurrent users. One of their optimizations included the restriction that guest users would all share the same memory instance of the guest layout.   Previously all guest users had separate guest layouts. Unfortunately, guest users will no longer be able to interact with the guest page by minimizing channels, but this is a small price to pay for improved performance. UBC can now serve 450-500 concurrent users using a machine with 640MB. Zed and Vikrant also wrote some tools for monitoring sessions. Zed will be checking in UBC's optimizations to the 1-0-patches CVS branch. There will likely be a 1.6 release of uPortal that contains UBC's improvements. For Zed's presentation, see
http://codebox.itservices.ubc.ca/zshaw/

Keywords – Debra Rundle (Princeton University), who was not present at this
developer's meeting, asked the group to discuss the role that keywords would play in
uPortal. The group sees two ways to initially implement keywords:

1) A channel publisher supplies keywords to a channel at publish time.
2) A channel's contents are examined to see if keywords can be extracted
3) During subscription time, a user could type one or more keywords to return a list
   of matching channels

The group agreed that this feature would be a useful one to have in uPortal 2.0 if it can be
developed in time for its release in May.

User profile management – Peter presented his progress on user profile management. A
user profile is a combination of structure and theme stylesheets that identifies a unique
rendering for a given user layout. A user will be able to choose one or more of these
profiles and assign a name to each one. The group had a discussion about the method
that the portal should allow the user to map each profile to some set of characteristics
associated with the user's session. For example, should users map their profiles to the
browsers they log in with (user-agent) or the combination of their browser and their
computer (user-agent/cookies)? The group chose the former option to keep things
simpler for 2.0.   To handle cases where the portal doesn't recognize a user's browser,
new JSPs or servlets are needed. These will present user interfaces for mapping a
browser to a profile and alert the system administrator that a new browser type has been
encountered.

Previously it was decided that there would be one abstract user preferences management
channel for the portal. Justin Tilton recommended that it should be possible to have
stylesheet-dependent user preferences channels. For example, if a user is using a
tab/column layout structure, then the user preferences channel should have related
terminology such as "move this column to the left" and "delete this tab".

Presentation architecture – Peter and Justin have decided to add an attribute to the folder
element in the user layout XML. This attribute will mark a folder as a "header" or
"footer". Stylesheets can use this attribute to optionally put content at the top or bottom
of the page. This content would need to be non-movable and non-removable. The group
doesn't know exactly how these restrictions would be implemented, but someone needs
to start looking at it for 2.0 soon.

Page rendering – Many universities have been expressing their desire to render html
pages as portal channels. This was possible in 1.5 but not yet in 2.0. It is more difficult
to do in 2.0 because of the strict requirement that channels emit their content as well-
formed XML. Pages with HTML that is not XHTML could not be parsed without first
converting to XHTML. There are tools such as JTidy that can be used to accomplish this
conversion. Andrew Draskoy has done some work with CGenericXSLT to make it
robust enough to act as a Web Proxy. He demonstrated that this new channel could be

used to put existing web applications into channels. His demo used Tomcat's Number Guess JSP example. It worked! The group, impressed with Andrew's work, has asked him to contribute this new channel to the 2.0 repository. It will be renamed to CWebProxy. Susan Bramhall is also contributing an IXMLChannel version of CInlineFrame which had already existed in 1.5. This channel will smoothly handle page rendering although it only works on some of the latest browsers such as IE 5 and Netscape 6.

Bugzilla – Although uPortal has a Bugzilla database set up, the uPortal developers haven't fully utilized its capabilities. Zed proposed that the developers start using Bugzilla to manage all assigned tasks. To prepare for this, it is necessary to revisit the "components" in Bugzilla and make sure that they are in sync with the Areas of Responsibility identified on the uPortal website. Once all our to-do's are entered into Bugzilla, the developers will be asked to provide status on a regular basis.

General issues – George Lindholm described his need to have a way to send files for use in the IMAP mail channel. The group decided that a separate JSP or servlet should be included to accommodate this need.

Any request parameters needed by the framework should be prefixed by "up_". All framework developers were reminded of this convention.

Zed Shaw proposed an event model rather than a "polling method" for channel rendering. Some side discussion between Peter Kharchenko and Zed ensued. This model may be implemented into uPortal at some point after 2.0 is released.

To assist channels in finding services, Zed and Bernie Durfee have proposed using JNDI. The group agreed that this is a good idea, so Zed and Bernie will be setting up a mechanism to do this for 2.0. Other uses for JNDI in uPortal include obtaining channel references and data sources.

After 2.0 is released, the development team will be taking a look at allowing for multi-threaded channels. This will reduce memory requirements for channels that don't need to preserve state. The same channel instance could be reused across different users and their layouts.

Because of the ambiguity of the word "channels", the group came up with more specific terms for channel concepts:

Channel type = the java class, e.g. CGenericXSLT
Channel declaration = the xml document describing a channel type's requirements for being published
Channel definition = a published channel with a global ID unique within the UP_CHANNELS table

Channel instance = an instance of a channel definition in a user's layout with an instance ID unique within the layout

Documentation – Bill Brooks presented the current state of uPortal documentation for 1.0 and 2.0. The uPortal documentation will be maintained as XML conforming to the DocBook standard. The documentation sources will be maintained in CVS. When distributions are made, the documentation will be processed by XSLT and turned into HTML, PDF, and possibly other types of documents. The HTML and PDF docs will also be accessible via the uPortal website in the documentation section. Anyone wishing to build the docs can do so at any time using XSLT stylesheets provided in the CVS repository. Bill's presentation can be found at http://mis105.mis.udel.edu/ja-sig/uportal/minutes/uPortal_documentation.ppt

For 2.0, the group made a decision to include at least the following information as part of the documentation:
A list of required libraries and versions needed to compile and run uPortal.
A list of possible ways for malicious channels to misbehave, e.g. calling System.exit()
A list of the standard CSS classes uPortal provides for use by channels
A list of exceptions the channel could throw
A list of known security holes in uPortal
A list of steps for making uPortal production-ready, e.g. replace database

IPlanet Calendar

Representatives from IPlanet presented the IPlanet Calendar server to the group. The group was very interested in finding out how to write code that would communicate with the calendar server. The reps told the group that there was a Java API available, but was not able to show the group any information on it. Ken Weiner planned to follow up with the reps to see if there really is a Java API. On the web, there was info on a C or C++ API called CSAPI, but this didn't interest the group of Java developers. For now, the group's understanding is that there is a way to send requests to the calendar server via http and get back results in XML.

uPortal Developers Meeting Minutes
University of British Columbia – March 26-30, 2001

uPortal 2.0 Task List

Key:
  Low = nice to have   Small = less than a day
  Med = should have   Med = less that a week
  High = Must have   Large = less than a month

Item	Person	Priority	Size
Write stylesheets for CLogin for Palm Pilot and cell phones	Ken Weiner	Low	Med
Change authenticate method signature of ISecurityContent to throw an exception if a service is not available	George Lindholm	Med	Small
Write CInlineFrame channel for page rendering in IFrame	Susan Bramhall	Med	Small
Enhance DbLoader to produce scripts and allow configuration for dropping/creating tables and inserting data	Ken Weiner	Med	Med
Complete CHttpProxy channel based on enhancements to CGenericXSLT	Andrew Draskoy	Low	Med
Rename ISpecialChannel to IPrivileged Channel	Ken Weiner	High	Trvial
Convert all jsp's to servlets	Zed Shaw & Peter Kharchenko	High	Large
Strive for forwards compatibility. Add IXMLChannel (mark as deprecated) which extends IChannel for easier porting of 1.0 IXMLChannels into 2.0	Ken Weiner	High	Small
Pass PortalEvent rather than LayoutEvent to channels	Peter Kharchenko	High	Small
Create Help, About, and Edit events for passing to channels	Peter Kharchenko	High	Small
Make sure channel declaration accounts for HasHelp, About, and Editable properties. Channel author must deal with these attributes.	John Laker	High	Med
Remove HttpServletRequest object from ChannelRuntimeData	George Lindholm	High	Med
Enhance subscribe channel to allow editing of parameters	John Laker	Med	Large
Incorporate roles and categories to publishing process	John Laker	High	Med
Publish the publishing channel	John Laker	High	Small
Create browser info object and pass it to	Peter Kharchenko	High	Med

channel via ChannelStaticData			
Standardize and document any framework-specific request parameters (prefixed with "up_")	Peter Kharchenko	High	Med
Create external jsp or servlet for facilitating sending files	George Lindholm	High	Med
Pass the SecurityContext to channels in ChannelStaticData	Ken Weiner - Try to delegate this to Yale	Low	Med
Come up with a set of standard attributes as recommended in the EduPerson standard. Document the exact names of these attributes and build a way to map these attribute names to local attribute names	Bruce Jolliffe & Vikrant Joshi	Low	Large
Make sure framework stylesheets understand userLayoutRoot	Justin Tilton	High	Small

Channel authoring document:
  -Exceptions you can throw
  -Events you may receive
  -CSS classes you can refer to
  -Person attributes you can ask for
  -How to make a channel publishable
  -What malicious things you should avoid doing
	Bill Brooks	Med	Large

Clean up obsolete CSS stylesheet stuff (.sdf files and extra database tables)	Peter Kharchenko	Med	Small
Make services available via JNDI lookups	Zed Shaw, Bernie Durfee	High	Large
Make channel instances available via JNDI lookups	Zed Shaw, Bernie Durfee	Low	Large
Explore java's SecurityManager for managing access control: e.g. not letting channels call System.exit()	Greg Barnes	Low	Med
Convert DBImpls to factories	Zed Shaw	High	Med
Normalize user layout xml and channel registry	George Lindholm, Zed Shaw & Doug Loewen	Med	Large
Replace references to user names with user IDs	George Lindholm & Doug Loewen	Med	Med

Framework deployment document
  -What implementations to replace for production
  -Malicious things channels can do (what to look for)
  -List of known security holes and concerns
	Bill Brooks, Zed Shaw, Ken Weiner	Med	Large

Sync AORs with Bugzilla	Ken Weiner & John Laker	Low	Small

Code auditor: Add a AOR for standards (logging, JavaDoc, Exception handling, code formatting)	Zed Shaw	Low	
Pass a new instance of Authentication to LayoutBean constructor	Zed Shaw & Peter Kharchenko	Med	Small
Rename LayoutBean to something more meaningful like LayoutProcessor	Paul Zablosky	Low	Med
Get rid of UtilitiesBean and move its methods to utils classes	John Laker	Med	Med
Add header and footer attribute to user layout xml	Peter Kharchenko & Doug Loewen	High	Small
Provide mechanism for users to get default layouts	Paul Zablosky, Doug Loewen, George Lindholm	High	Large
Make sure that header and footer folder are immutable.	Bernie Durfee	High	Med
Implement non-movable (must be child of non-movable) and non-removable channels and folders	Bernie Durfee	Low	Large
Convert CIMAPMail to IXMLChannel	George Lindholm	High	Med
Implement UBC Memory Management features	Zed Shaw	Low	Large
Add profile management to CUserPreferences	Peter Kharchenko	High	Large
Allow user to map profile to unrecognized browser (HTML) during initial login	Peter Kharchenko	High	Large
Implementing keywords into publishing and subscribing	Debra Rundle	Med	Large
Add these tasks to Bugzilla	Ken Weiner	Low	Small
Architecture overview docs	Bill Brooks	Med	Large
IMAP mail channel enhancements: Address book service, signatures, user prefs	Vikrant Joshi & George Lindholm	Low	Large
Set up local SourceForge at Udel	John Laker	Low	Large
Run code through code formatter just before release	John Laker	Med	Small