

uPortal Migration: 3.0 & Beyond

Drew Wills

JA-SIG Spring Conference, San Diego CA

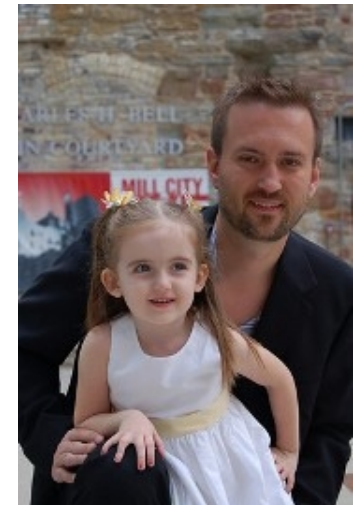
March 7th, 2010

© Copyright Unicon, Inc., 2006. This work is the intellectual property of Unicon, Inc. Permission is granted for this material to be shared for non-commercial, educational purposes, provided that this copyright statement appears on the reproduced materials and notice is given that the copying is by permission of Unicon, Inc. To disseminate otherwise or to republish requires written permission from Unicon, Inc.

Facilitator: Drew Wills



- Joined Unicon November 2000
- JA-SIG Committer since 2004
- Contributed to several uPortal implementations:
 - California Polytechnic State University ("*Cal Poly*")
 - California State University, Chico
 - University of Colorado System
 - Yale University
 - Johns Hopkins University
 - University of Illinois
 - Lynn University
 - myPearson Portal
- Launched Cernunnos February 2007



Participants

- Please share a bit about yourself:
 - Name & institution
 - Do you run uPortal? What version?
 - Do you plan to deploy or upgrade uPortal in the next 12 months?



Agenda

1. uPortal 3.0 – 3.2 Highlights
2. Implementing uPortal 3.2
3. Import/Export Migration Tools
4. Migration Walk Through

UPortal 3.0 – 3.2 Highlights

Why you should bother upgrading

New in uPortal 3.0

- Maven2 build & project management
- Unified Spring context
- Import/Export tools for portal data
- Universality theme
- Bundled with CAS 3.2
- CAS Proxy Tickets for JSR-168 portlets
- Over *150* new bug fixes, enhancements, and features since 2.6!

<http://www.ja-sig.org/wiki/x/foCQ>

New in uPortal 3.1

- Database-backed portal statistics
- Identity- & Attribute-Swapping portlets for admins
- DLM Fragment Admin portlet; *supports delegation*
- Fine-grained permissions for channel management
- Fluid Skinning System for modular CSS
- Completely div-based XHTML theme
- Over 260 new bug fixes, enhancements, and features since 3.0!

<http://www.ja-sig.org/wiki/x/WR3P>

New in uPortal 3.2

- New user interface for portlet administration
- New portlet lifecycle features (created, approved, published, & expired); *supports delegation*
- CONFIG mode for custom portlet configuration
- iPhone theme (universality) for both uPortal & CAS
- Improved configuration UI for Web Proxy Portlet
- 5 new bundled portlets: Weather, GoogleMaps, Translation, Dictionary/Thesaurus, CAS Proxy Test
- Over 200 new bug fixes, enhancements, and features since 3.1!

<http://www.ja-sig.org/wiki/x/T4WuAQ>

Implementing uPortal 3.2

Changes to Implementation Since 2.x

uP3 – General Approach

- Tackling the vendor branch

Where did everything go?

- Create a new branch for 3.2
- Compare and catalog local code to its base version
- Re-evaluate changes then merge the need modifications

uP3 - Configuration

- Spring code wiring instead of Factories
- Most configuration is still in portal.properties

<http://www.ja-sig.org/wiki/x/EAWDAQ>

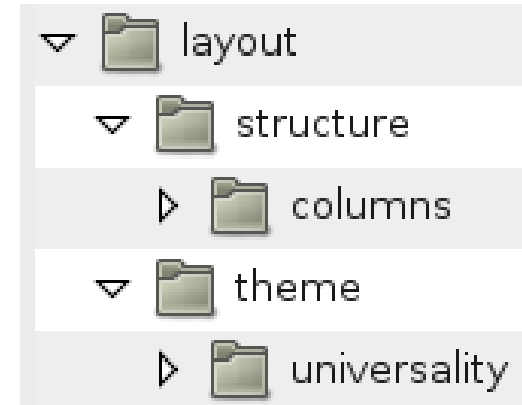
- Details Person Directory and LDAP transition
- Other configuration files will be added soon

<http://www.ja-sig.org/wiki/x/QgKF>

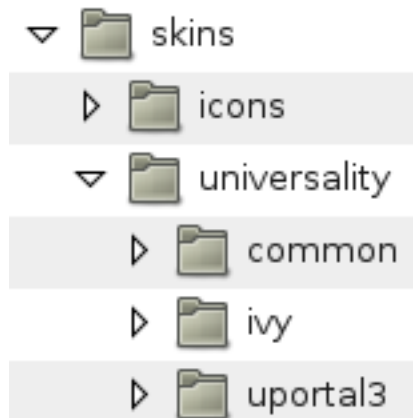
- Details property level changes for configuration

New Theme & Skin

- Simple Directory Structure
 - Separate Structure & Theme
 - Meaningful Directory Names



- Flexible Skin Directories
 - Allows for shared files
 - Skins are assumed to be theme specific



New Theme & Skin

Theme File Summary

- universality.xsl - Core file, has instructions and configuration
- page.xsl - Contains base portal page layout templates
- navigation.xsl - Contains portal navigation templates
- columns.xsl - Per-column layout and formatting templates
- content.xsl - Portlet/Channel container templates
- preferences.xsl - User Preferences UI templates
- components.xsl - Common utility templates
- messages.xml - I18n capable messages file, contains all text

Import/Export Migration Tools

Data Agility

Origins of Import/Export

- Drew Wills performed several upgrades in 2006-2007:
 - California Polytechnic State University "Cal Poly" (2.1 to 2.5)
 - University of Colorado System (2.4 to 2.5)
 - California State University, Chico (2.4 to 2.6)
 - Johns Hopkins University (2.5.2 to 2.6.1+)
 - *Johns Hopkins University (2.6.1 to 3.0.1+)
- Database record synthesis was a big challenge common to all of them

Origins (Cont.)

- The earliest Import/Export was created to make these assignments easier for myself
- uPortal Developers' Meeting at Johns Hopkins, April 2007:

<http://www.ja-sig.org/wiki/x/KQIs>

- Yale University Sponsorship (Q4 2007)

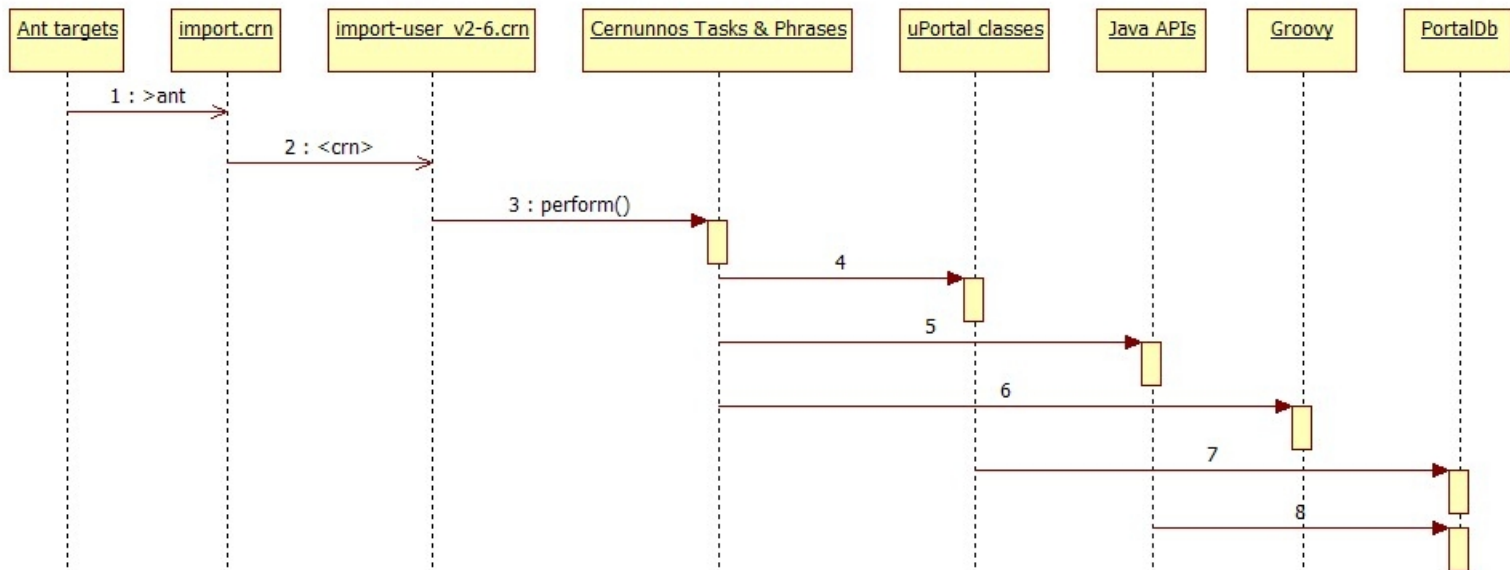


Overview

- Import/Export solution “stack”
 - **XML**: domain-specific schema
 - **Apache Ant**: target-based interface
 - **Cernunnos**: component integration
 - **uPortal**: custom Java technology
- Originally, Import/Export was developed as an “overlay” – source files may be added but not altered

Overview (Cont.)

uPortal Import/Export Sequence Diagram



The Cernunnos Project

- Cernunnos helps you be more productive
- *Here's how it works...*
 - You don't have to tell components, subsystems, or objects *how* to work together
 - You just have to tell them to do so
 - This simple difference reduces busywork and bulk dramatically
- In other words: *Cernunnos shifts from the **Jigsaw Paradigm** to the **LEGO Paradigm***



Import/Export Features

- Offers Ant-based create/update/delete operations on most portal entities
- uPortal 3.1.0 Supports the following entities:
 - Entity Types
 - Structures
 - Themes
 - Users
 - Groups
 - Groups with Members
 - Channel Types
 - Channels
 - Memberships
 - Permissions
 - Permission Sets
 - Layouts
 - DLM Fragment Definitions

Features (Cont.)

- Import/Export does *not* support:
 - ALM
 - Though there is a utility for converting from fragments.xml
 - User customizations to DLM fragments (pre-3.2)
 - Duplicate group names
 - Object types that do not have non-sequential unique identifiers (*i.e.* username, fname) must have unique names

File Types

- Import/Export uses the following file extensions:
 - .entity-type*
 - .structure*
 - .theme*
 - .template-user*
 - .user*
 - .group
 - .group_membership*
 - .channel-type*
 - .channel*
 - .membership
 - .permission
 - .permission_set*
 - .fragment-layout*
 - .layout*
 - .fragment-definition*
 - .batch
- Portal entities are **imported in this sequence** to deal with dependencies
- Types with an asterisk (*) are created by **crn-export -Dtype=all**

Example: uP3 guest Layout

```
<layout script="//org/jasig/portal/io/import-layout_v3-0.crn" username="guest">
  <root unremovable="Y" immutable="N" hidden="N" name="Root folder">
    <header unremovable="Y" immutable="Y" hidden="N" name="Header folder">
      <channel unremovable="" immutable="" hidden="" fname="header"/>
      <channel unremovable="" immutable="" hidden="" fname="portal/login/general"/>
      <channel unremovable="" immutable="" hidden="" fname="session-locales-selector"/>
      <channel unremovable="" immutable="" hidden="" fname="fragment-admin"/>
    </header>
    <tab unremovable="N" immutable="N" hidden="N" name="uPortal News">
      <column unremovable="N" immutable="N" hidden="N" name="Column 1">
        <structure-attribute type="folder">
          <name>width</name>
          <value>40%</value>
        </structure-attribute>
        <channel unremovable="" immutable="" hidden="" fname="uportal-news"/>
        <channel unremovable="" immutable="" hidden="" fname="uportal-powered-sites"/>
      </column>
      <column unremovable="N" immutable="N" hidden="N" name="Column 2">
        <structure-attribute type="folder">
          <name>width</name>
          <value>60%</value>
        </structure-attribute>
        <channel unremovable="" immutable="" hidden="" fname="jasig-news"/>
        <channel unremovable="" immutable="" hidden="" fname="unicon-news"/>
      </column>
    </tab>
    <footer unremovable="Y" immutable="Y" hidden="N" name="Footer folder">
      <channel unremovable="" immutable="" hidden="" fname="footer"/>
    </footer>
  </root>
  <profile name="Guest Web Browser Profile">
    <description>A sample profile for web browsers</description>
    <structure name="DLM Tabs and columns"/>
    <theme name="DLM XHTML"/>
  </profile>
</layout>
```

Example: uP3 guest User

```
<user script="classpath://org/jasig/portal/io/import-user_v3-0.crn" username="guest">  
  <default-user>defaultTemplateUser</default-user>  
</user>
```


Using Import/Export

- Ant tasks:
 - **crn-export**: Creates XML representations of the specified object(s)
 - **crn-import**: Modifies the database to match the specified XML document(s)
 - **crn-delete**: Removes the specified object(s) from the database
 - **crn-make-data-xml**: Synthesizes information in your database and some default data to create a slim data.xml file

```
>ant crn-export -Ddir=mylayouts -Dtype=all-layouts
```

Command Reference for uPortal 3.1.0

```
>ant [crn-export | crn-delete] -Dtype=type [-Ddir=dir]  
[-Dsysid=sysid]
```

```
>ant crn-import [-Ddir=dir] [-Dpattern=regex]
```

type = all | layout | all-layouts | all-permissionst |
all-permission_sets† | all-membershipst | channel |
all-channels | channel-type | all-channel-types | group
| all-groups | group_membership† | all-
group_membership† | user | all-users | theme | all-
themes | structure | all-structures | entity-type |
all-entity-types | fragment-definition | all-fragment-
definitions

dir = file system directory (relative or absolute)

sysid = username††, fname, channel type name, group
name, theme name, structure name, entity type name, or
fragment definition name

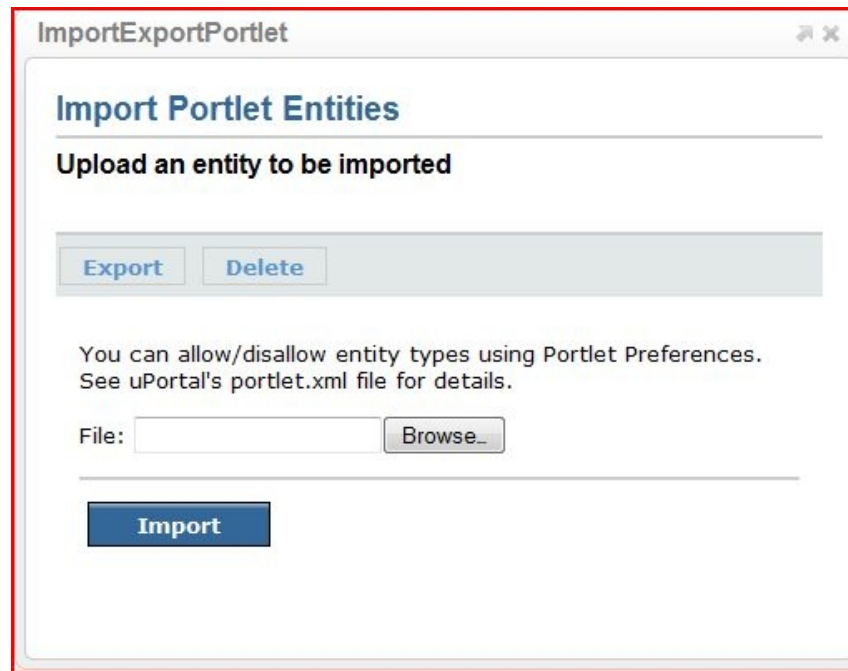
regex = A regular expression used to match files in *dir*
(e.g. `.*layout`)

Ant Command Notes

- You cannot delete *all* or *all-* of anything
- † *permissions and memberships records cannot be deleted directly. They will be removed, where appropriate, whenever an entity that they reference is deleted*
- †† *specify username for **sysid** when referring to both users and layouts*

ImportExport Portlet

- This Portlet provides access to uPortal Import/Export capabilities from the portal UI



The screenshot shows a web portlet window titled "ImportExportPortlet". Inside, there is a section titled "Import Portlet Entities" with a subtitle "Upload an entity to be imported". Below this, there are two buttons: "Export" and "Delete". A text block states: "You can allow/disallow entity types using Portlet Preferences. See uPortal's portlet.xml file for details." Below this text is a "File:" label followed by an empty text input field and a "Browse..." button. At the bottom of the portlet is a large blue "Import" button.

- You can restrict allowable operations at deploy/publish time with Portlet Preferences

ImportExport Portlet (cont.)

- Import Operations: any valid document
- Export Operations:

```
layout | channel | channel-type | group | user  
| theme | structure | entity-type |  
fragment-definition
```

- Delete Operations (use *extreme caution*):

```
entity-type | channel | channel-type | group |  
layout | structure | theme user | fragment-  
definition
```

Recommendation: `entities.location`

- Replace the hard-coded location of entity files with a property: `entities.location`
- Advantages:
 - You can replace your whole universe with one property change

```
>ant initdb
```

- You can even do...

```
>ant initdb -Dentities.location=myDir
```

- You can maintain both your data and uPortal sample data in your SCC

Recipe #1 (Cont.)

- Directions:
 - Add the property `entities.location` to `build.properties`

```
entities.location=  
  uportal-impl/src/main/resources/properties/db/entities
```

Recipe #1 (Cont.)

- Directions (Cont.):
 - Enhance build.xml to use this property for the location of entity documents

```
[~line #24]
<property name="entities.location"
value="properties/db/entities" />
...
[~line #147]
<target name="db-import" depends="prodPrompt"
description="Imports the default XML files
to the database">
  <echo message="Importing data" />
  <antcall target="crn-import">
    <param name="dir" value="${basedir}/${
entities.location}" />
  </antcall>
</target>
```


Recommendation: env/ Directories

- Organize settings & data for environments like prod/dev/local into separate directories
- Advantages:
 - You can keep data for what you're working on separate from existing data
 - You can easily work on multiple data sets independently
 - You can "promote" new data from dev -> test or from test -> prod using automated tools
 - You can evolve your data in SCC (just like source files)

env/ Directories (Cont.)

- Directions:

- `mkdir env` or

`uportal-impl/src/main/resources/properties/env`

- Within `env`, create subdirectories for each environment you support and place an `entities/` directory within each, *e.g.*

- `env/local/entities/`
 - `env/dev/entities/`
 - `env/test/entities/`
 - `env/prod/entities/`

env/ Directories (Cont.)

- Load your environment like this:

```
>ant crn-import -Ddir=env/<env.name>/entities
```

- Or enhance build.xml:

```
[~line #147]
<target name="db-import" depends="prodPrompt"
description="Imports the default XML files to the database">
  <echo message="Importing data" />
  <antcall target="crn-import">
    <param name="dir" value="${basedir}/$
{entities.location}" />
  </antcall>
  <antcall target="crn-import">
    <param name="dir" value="${basedir}/env/$
{env}/entities" />
  </antcall>
</target>
```

```
>ant initportal -Denv=local
```

Recommendation: Environmental Tokens

- Use Ant tokens and/or Maven filters to *inject* environment-specific settings at build time
- Advantages:
 - You can run some administrative Ant tasks for any environment from any machine
 - You can maintain configuration for every environment in SCC

Migration Walk Trough

Phase 1: Local Customizations

- Follow these steps:
 - `>svn export` the current codebase of your portal
 - `>svn export` the tag corresponding to release on which it is based
 - e.g. <https://www.ja-sig.org/svn/uPortal/tags/rel-2-5-2/>
 - Use Beyond Compare (or similar tool) to identify local customizations

Beyond Compare

Session: <untitled> Filters: *.*

C:\OLD_HOME\clients\JHU\old-portal_rel-2-5-2\source\org\ C:\OLD_HOME\clients\JHU\old-portal_jhu-prod\source\org\j

Name	Size	Modified	Name	Size	Modified
lib	20.6 MB	4/21/2008 1:04:42 pm	lib	28.1 MB	4/21/2008 1:06:04 pm
properties	1.21 MB	4/21/2008 1:04:38 pm	properties	1.37 MB	4/21/2008 1:05:57 pm
source	5.29 MB	4/21/2008 1:04:17 pm	source	5.35 MB	4/21/2008 1:05:53 pm
org	5.29 MB	4/21/2008 1:04:17 pm	edu	12,855	4/21/2008 1:05:53 pm
jasig	5.29 MB	4/21/2008 1:04:17 pm	org	5.34 MB	4/21/2008 1:05:32 pm
portal	5.29 MB	4/21/2008 1:04:36 pm	portal	5.34 MB	4/21/2008 1:05:53 pm
car	108 KB	4/21/2008 1:04:36 pm	car	106 KB	4/21/2008 1:05:53 pm
channels	940 KB	4/21/2008 1:04:36 pm	channels	932 KB	4/21/2008 1:05:53 pm
concurrency	121 KB	4/21/2008 1:04:32 pm	concurrency	117 KB	4/21/2008 1:05:50 pm
container	283 KB	4/21/2008 1:04:32 pm	container	279 KB	4/21/2008 1:05:49 pm
core	7,769	4/21/2008 1:04:31 pm	core	7,524	4/21/2008 1:05:47 pm
groups	342 KB	4/21/2008 1:04:31 pm	groups	342 KB	4/21/2008 1:05:47 pm
filesystem	33,508	4/21/2008 1:04:31 pm	filesystem	32,454	4/21/2008 1:05:47 pm
ldap	28,508	4/21/2008 1:04:31 pm	ldap	27,664	4/21/2008 1:05:47 pm
local	9,819	4/21/2008 1:04:30 pm	local	9,555	4/21/2008 1:05:47 pm
pags	37,756	4/21/2008 1:04:30 pm	pags	41,391	4/21/2008 1:05:47 pm
ComponentGroupServiceDes...	2,797	10/17/2004 12:01:46 pm	ComponentGroupServiceDes...	2,696	7/6/2005 12:17:52 pm
CompositeEntityIdentifier.java	4,364	3/7/2005 1:09:07 pm	CompositeEntityIdentifier.java	10,470	7/19/2006 9:55:27 pm
CompositeServiceIdentifier.ja...	1,406	10/17/2004 12:01:46 pm	CompositeServiceIdentifier.ja...	1,352	7/6/2005 12:17:52 pm
EntityGroupImpl.java	20,914	3/7/2005 1:09:07 pm	EntityGroupImpl.java	20,206	7/6/2005 12:17:52 pm
EntityGroupNameFinder.java	2,048	10/19/2004 8:49:56 am	EntityGroupNameFinder.java	1,974	7/6/2005 12:17:52 pm
EntityGroupNameFinderFact...	956	3/7/2005 1:09:07 pm	EntityGroupNameFinderFact...	920	7/6/2005 12:17:52 pm
EntityImpl.java	2,644	10/17/2004 12:01:46 pm	EntityImpl.java	2,537	7/6/2005 12:17:52 pm

4/24/2008 7:11:01 PM Load Comparison: C:\OLD_HOME\clients\JHU\old-portal_rel-2-5-2 <-> C:\OLD_HOME\clients\JHU\old-portal_jhu-prod
 4/24/2008 7:12:08 PM Rules-Based Comparison of 907 Files
 4/24/2008 7:12:39 PM Rules-Based Comparison of 907 Files

186 file(s), 1.11 MB (76 GB free on C:) 189 file(s), 1.12 MB (76 GB free on C:)

What To Do with Discrepancies

- Files present in your portal, not in tag
 - Port forward everything that's still in use (easy)
- Files present in tag, not in your portal
 - These are unexpected, consider them individually
- Files that are different
 - *Hope* for few of these, handling these is the bulk of the effort in an upgrade
 - Make a list (spreadsheet) of all substantive differences

Customizations Spreadsheet

modifications.xls [Compatibility Mode] - Microsoft Excel

File	Description	IGNORE	RETAIN	DONATE	GCCC	UP-JIRA	ESM	CMD	What Happened?
source\org\jasig\portal\channels\codemol\CViewer.java	Support for Opera user-agent strings			X			X		committed 2-6-patches & trunk/
source\org\jasig\portal\channels\portlet\CPortletAdapter.java	Loading PortletDefinition preferences	X				UP-1040			
source\org\jasig\portal\channels\webproxy\CWebProxy.java	Caching enhancements			X	X				
source\org\jasig\portal\channels\CGeneric\SLT.java	Caching enhancements			X	X				
source\org\jasig\portal\channels\CHeader.java	Enhancements to CHeader for PortalAdmins & SSLVPN		X						port
source\org\jasig\portal\channels\CInlineFrame.java	Enhanced to choose stylesheet title by user agent string	X							AV (2007/12/20) Moved to red b/c I got this one backwards -- 2.8.2 had more features than JHU prod.
source\org\jasig\portal\content\ainer\binding\PortletApplicationUnmarshaler.java	Some commons-logging and safer call to PortletDefinition.setExpirationCache()			X		UP-1173	X		committed 2-6-patches & trunk/
source\org\jasig\portal\content\ainer\om\entity\PortletEntityImpl.java	Changes to the way portlet preferences are managed	X				UP-1040			
source\org\jasig\portal\content\ainer\om\portlet\PortletDefinitionImpl.java	Changes to the way portlet preferences are managed	X				UP-1040			
source\org\jasig\portal\content\ainer\ervlet\ServletRequestImpl.java	Logging enhancements	X							
source\org\jasig\portal\groups\CompositeEntityIdentifier.java	Added utility methods for compound keys	X							
source\org\jasig\portal\layout\dm\ILFBuilder.java	Loop detection and logging	X							
source\org\jasig\portal\layout\dm\ROBMDistributedLayoutStore.java	jhuDlm\workaround -- this class changed a lot, let's be sure the problem still occurs		?						
source\org\jasig\portal\ldap\LdapServerImpl.java	Added connection pool features		X				X		AV (2007/12/14) -- Moving these 2 to the blue column for 2 reasons: (1) I have a feeling I'll be starting too many initiatives on uPortal-dev; and (2) 3.0 is replacing custom LDAP code w/ spring LDAP code. Ported.
source\org\jasig\portal\ldap\LdapServices.java	Added connection pool features		X				X		
source\org\jasig\portal\security\provider\PersonImpl.java	Changed the way guest user(s) are identified. Do what Yale did			X					Get info from andrew.p. and susan
source\org\jasig\portal\security\provider\RemoteUserPersonManager.java	Changed the way guest user(s) are identified. Do what Yale did			X					
security\provider\RemoteUserSecurityContext.java	Special treatment for PortalAdmins. Do what Yale did			X					
source\org\jasig\portal\serialize\BaseMarkupSerializer.java	Performance optimization			X	X				committed 2-6-patches & trunk/
source\org\jasig\portal\serialize\HTML.dtd.java	Performance optimization			X	X				committed 2-6-patches & trunk/
source\org\jasig\portal\serialize\HTML.Serializer.java	state.unescaped = true; (???)	X							
source\org\jasig\portal\serialize\Printer.java	Enhancements to printText() method			X	X				committed 2-6-patches & trunk/
source\org\jasig\portal\serialize\HTML.Serializer.java	Performance optimization			X	X				committed 2-6-patches & trunk/
source\org\jasig\portal\services\person\dir\support\legacy\PersonDirInfoAdaptor.java	Logged a warning when an LDAP connection can't be obtained from the pool	X							
source\org\jasig\portal\services\person\dir\support\mergen\MultiValuedAttributeMerger.java	Added a lot of debug messages	X							
	Added user info caching -- retain JHU-specific code until a better caching strategy is implemented -- use	X							

Customizations by Codebase Region

- properties/ (Configuration)
 - These will always be customized
 - Watch for new settings since the previous release
 - **Good Idea:** compare properties/ directories of old and new releases
- lib/ (.jar files)
 - Usually some added jars for use by added code
 - Dependencies are managed by Maven now; you will need `<groupId>`, `<artifactId>`, and `<version>`

Codebase Region (Cont.)

- webpages/ (UI resources)
 - Structure/Theme files
 - Default location: stylesheets/org/jasig/portal/layout/
 - **Recommendation:** use Universality or add a new theme
 - use Import/Export to "install" a new Structure/Theme
 - `>ant crn-import -Ddir=<dir> -Dpattern=<filename>`
 - adjust .user files to refer to the new Structure/Theme
 - **Recommendation:** "piggy-back" your new theme on universality – import as many un-altered stylesheets as possible

Codebase Region (Cont.)

- webpages/ (Cont.)
 - Options for channel UI files
 - **Drop** (easiest)
 - **Contribute** (good for the community)
 - **Port Forward** (maybe, for very small changes)
 - **Fork** (better for substantial changes)

Codebase Region (Cont.)

- source/ (Java files, mostly):
 - This region is the heart of this analysis
 - **Changes that break APIs are challenging**
 - Changes that address bugs logging, error handling, performance, *etc.* are manageable
 - You have the same options as before:
 - **Drop** (easiest -- make this the most common choice)
 - **Contribute** (good for the community)
 - **Port Forward** (best to avoid)
 - **Fork** (possible where there are good interfaces)

Contributing Enhancements

- An upgrade is a great time to "clean house" and improve uPortal for everyone
- uportal-dev email list

<http://www.ja-sig.org/wiki/display/JSG/uportal-dev>

- uPortal Issue Tracker (JIRA)

<http://www.ja-sig.org/issues/browse/UP>

- uPortal Wiki

<http://www.ja-sig.org/wiki/display/UPC/Home>

Phase 2: Create a New Project

- Get the new source code:
 - `>tar -xzf uPortal-3.1.0.tar.gz`
 - `>svn export`
 - <https://www.ja-sig.org/svn/uPortal/tags/rel-3-1-0-RC1/>
 - `>svn import My.School.edu/sandbox`
- Drop in your properties/
- Drop in your structure, theme, and skin resources
- Apply any retained modifications

Phase 3: Data Export

- uPortal 3's database schema is new; *you can't run the new codebase against your existing DB*
- You have to create a new database and migrate your data to it
- There are two approaches you can take
- Both approaches rely on Import/Export in the new portal *and the old portal*

Availability of Import/Export

- Import/Export tools exist for uPortal versions:
 - 2.5.4 (not released)
 - 2.6.2 (not released)
 - 3.0.0+
- Using Import/Export with versions before uP3:
 - **Option 1:** Patch Import/Export into your existing portal codebase
 - **Option 2:** Use Import/Export with a fresh checkout

Option 1: Patch Your Existing Portal

- Use this approach if you plan to stay on your current uPortal release
- Add the following files from the rel-2-6-patches/ branch:

- import-export.xml (contains Ant tasks)

```
>ant -f import-export.xml [import|export|delete]
```

- package org.jasig.portal.io

- (all files)

- package org.jasig.portal.tools.chanpub

- UrlChannelPublisher.java

Option 1: Patch (Cont.)

- Add the following files (Cont.):
 - package org.jasig.portal.tools.dbloader
 - dbunload.crn
 - StringListPhrase.java
 - UP_GROUP.default-data.xml
 - UP_USER_PROFILE.default-data.xml
 - UP_SEQUENCE.default-data.xml
 - UP_USER.default-data.xml

Option 1: Patch (Cont.)

- Add the following files (Cont.):
 - JAR files in lib/compile/ (assumes 2.6.1 GA):
 - cernunnos-1.0.0-M6.jar (replace M3)
 - commons-lang-2.3.jar
 - groovy-all-1.0.jar
 - groovy-engine-1.0-jdk14.jar
 - retrotranslator-runtime-1.2.1.jar
 - script-api-1.0-jdk14.jar

Option 2: Use a Fresh Checkout

- Use this approach if you plan to migrate to a newer uPortal release

```
>svn co https://www.ja-sig.org/svn/uPortal/branches/rel-2-6-patches/
```

- Configuration Instructions:
 - configure files in properties/
 - personDirectory.xml
 - rdbm.properties
 - configure files in properties/groups/
 - compositeGroupServices.xml
 - PAGSGroupStoreConfig.xml (if applicable)
 - Add your JDBC driver to lib/ (2.5.x) or lib/container-common/ (2.6.x)

Simple Migration Method

- Gathers *all* portal data
- Suitable for migrating your prod data to a new prod DB
- Directions:

- `>ant crn-make-data-xml` (Optional)

replace uPortal's data.xml with the file in the root directory

- `>ant crn-export -Ddir=myData -Dtype=all`

(Optionally) replace the contents of properties/db/entities/ or use `entities.location` (preferred)

Advanced Migration Method

- Ignores "regular" users
- Suitable for quickly provisioning new portal environments
 - Functional testing
 - Scalability testing
 - Development
 - Local
- Also suitable for dropping user layout customizations

Advanced Migration Directions

- `>ant crn-make-data-xml` (Optional)

Replace uPortal's data.xml with the version in the root directory

- Edit the `included_users_file` setting in `export.properties`
- Enter the location of a file with one username on each line; DLM fragment owners need not be listed
- `>ant crn-export -Ddir=myData -Dtype=all`

Phase 4: Configuring uPortal 3

Configuration Required for Import

- **Database**
- **LDAP**
- **User Attributes**
- **Groups**
- **Layout Fragments**

Database Configuration

- `rdbm.properties`
 - `uportal-impl/src/main/resources/properties/`
 - Hibernate Dialects for Database Type Selection
- `pom.xml`
 - JDBC Driver Configuration
 - Driver must be available to Maven
 - `mvn install:install-file`
 - Run a Maven repository manager for the project

Example: uP3 JDBC Driver

rdbm.properties

```
##### Oracle 10g - example
hibernate.connection.driver_class=oracle.jdbc.driver.OracleDriver
hibernate.connection.url=jdbc:oracle:thin:@mydb.school.edu:1654:MYPROD
hibernate.connection.username=myuser
hibernate.connection.password=mypassword
hibernate.dialect=org.hibernate.dialect.Oracle10gDialect
```

pom.xml

```
...
<properties>
  <!-- The final name of the uPortal web-application -->
  <uportal.docbase>portal</uportal.docbase>

  <!-- The JDBC Driver used by uPortal -->
  <jdbc.groupId>com.oracle</jdbc.groupId>
  <jdbc.artifactId>ojdbc5</jdbc.artifactId>
  <jdbc.version>11.1.0.6.0</jdbc.version>
...

```

LDAP Configuration

- IdapContext.xml
 - uportal-impl/src/main/resources/properties/contexts/
 - Spring-LDAP used for connections
- One for Authentication
 - defaultLdapServer - ContextSourceLdapServerImpl
 - May need its own LdapContextSource
- One for Attributes (and future development)
 - defaultLdapContext - LdapContextSource

Example: uP3 LDAP

ldapContext.xml

```
<!--  
| Legacy LDAP configuration, needed for use by deprecated LDAPServices code.  
| Required for:  
| - SimpleLdapSecurityContext  
+-->  
<bean id="defaultLdapServer" class="org.jasig.portal.ldap.ContextSourceLdapServerImpl">  
  <property name="ldapContextSource">  
    <bean class="org.springframework.ldap.core.support.LdapContextSource">  
      <property name="url" value="ldap://ldap.myuniv.edu:389"/>  
      <property name="userName" value="cn=AuthUser, ou=People, dc=myuniv, dc=edu"/>  
      <property name="password" value="foobar"/>  
      <property name="pooled" value="false"/>  
    </bean>  
  </property>  
  <property name="uidAttribute" value="uid"/>  
  <property name="baseDN" value="ou=People, dc=myuniv, dc=edu"/>  
</bean>  
  
<!--  
| Standard Spring ContextSource declaration, used by all new code post 3.0.0.  
| Required for:  
| - Person Directory LDAP Attributes  
+-->  
<bean id="defaultLdapContext" class="org.springframework.ldap.core.support.LdapContextSource">  
  <property name="url" value="ldap://ldap.myuniv.edu:389"/>  
  <property name="base" value="ou=People, dc=myuniv, dc=edu"/>  
  <property name="userName" value="cn=AuthUser, ou=People, dc=myuniv, dc=edu"/>  
  <property name="password" value="mypass"/>  
  <property name="pooled" value="false"/>  
</bean>
```

Attributes Configuration

- personDirectoryContext.xml
 - uportal-impl/src/main/resources/properties/contexts/
 - Person Directory 1.5
 - <http://www.ja-sig.org/wiki/display/PDM15>
- Slight change since 3.0
- Includes search functionality

Example: uP3 Attributes

personDirectoryContext.xml

```
<bean class="org.jasig.services.persondir.support.jdbc.SingleRowJdbcPersonAttributeDao">
  <constructor-arg index="0" ref="PersonDB" />
  <constructor-arg>
    <value>
      SELECT FIRST_NAME||' '||LAST_NAME AS FIRST_LAST, FIRST_NAME, LAST_NAME, EMAIL, USER_NAME
      FROM UP_PERSON_DIR
      WHERE {0}
    </value>
  </constructor-arg>
  <property name="usernameAttributeProvider" ref="usernameAttributeProvider" />
  <property name="queryAttributeMapping">
    <map>
      <entry key="username" value="USER_NAME" />
      <entry key="givenName" value="FIRST_NAME" />
      <entry key="sn" value="LAST_NAME" />
      <entry key="mail" value="EMAIL" />
    </map>
  </property>
  <property name="resultAttributeMapping">
    <map>
      <entry key="FIRST_LAST" value="displayName" />
      <entry key="FIRST_NAME" value="givenName" />
      <entry key="LAST_NAME" value="sn" />
      <entry key="EMAIL" value="mail" />
      <entry key="USER_NAME">
        <set>
          <value>uid</value>
          <value>username</value>
        </set>
      </entry>
    </map>
  </property>
</bean>
```

Groups Configuration

- `compositeGroupServices.xml`
 - `uportal-impl/src/main/resources/properties/groups`
 - Same groups config from 2.x
- Copy configuration files from existing uPortal install

DLM Configuration

- dlm.xml
 - uportal-impl/src/main/resources/properties
 - Same DLM config from 2.5 and later
- Copy configuration files from existing uPortal install

Theme & Skin Updates

- Migrating Theme Customizations
 - Start with universality and modify
 - Base theme has significant improvements from 2.x
- Skinning
 - Make a copy of default universality skin
 - Update .theme file with your new skin name
 - Force layouts to use the new theme

uPortal UI at This Conference

- *"How to use JavaScript in uPortal 3"*
 - JavaScript in portlets in uPortal 3; jQuery, namespacing
 - **Monday 3/2 (Tomorrow) 2:00 PM – 3:00 PM**
- *"Fluid Layout Reorderer Showcases"*
 - Showcasing the Drag & Drop UI in uPortal 3.1
 - **Wednesday 3/4 10:00 AM – 11:00 AM**

Phase 5: Data Import

- Setup base portal entities
- Copy exported data.xml into uP3
- Configure import.properties
- Point crn-import to uP2 data export
 - Using entities.location technique described earlier

Base Entities

- Requirements for an uPortal install
 - Users:
 - system, defaultTemplateUser, fragmentTemplate, guest
 - Layouts
 - system, defaultTemplateUser, fragmentTemplate, guest
 - Define headers/footers in defaultTemplateUser.layout
 - Entity Types – All included
 - Channels referenced by base layouts
 - New Entities – Refer to release notes

Base Entities

- Channel Types
 - Some pruning can be done here
- Theme and Layout files (optional)
 - Using those included with uP3 allows for easier transition to new theme and skin
 - Can be forced on layouts via `import.properties`

Import Properties

- Provides important defaults
 - Layout structure name
 - Layout theme name
 - Default user name
- Provides option to force each default
 - Useful when switching to the new theme
 - No need to updated thousands of user and layout files by hand

Testing the Database Config

```
$ ant dbtest

Buildfile: build.xml

dbtest:

....SNIP....

    [echo] Invoking DbTest
    [java] WARN The uPortal database is not initialized, the database tests will not be
performed.
    [java] INFO HSQL Database Engine (1.8.0) / HSQL Database Engine Driver (1.8.0)
database/driver
    [java] Connected To: jdbc:hsqldb:hsql://localhost:8887
    [java] WARNING: uPortal tables do no exist, not all meta-data tests were executed.
    [java]
    [java] Database name:      'HSQL Database Engine'
    [java] Database version:   '1.8.0'
    [java] Driver name:       'HSQL Database Engine Driver'
    [java] Driver version:    '1.8.0'
    [java] Driver class:     'HSQL Database Engine Driver'
    [java] Connection URL:    'jdbc:hsqldb:hsql://localhost:8887'
    [java] User:            'SA'

....SNIP....

BUILD SUCCESSFUL
Total time: 10 seconds
```


Initializing the Database

- **initdb**
 - The only ant task you should need, calls others
- **db**
 - Creates tables from tables.xml
 - Populates tables from data.xml
- **db-hibernate**
 - Creates tables for Hibernate entities
- **Db-import**
 - Runs crn-import with default 'dir' argument

Questions?



Drew Wills

drew@unicon.net

<http://www.unicon.net/blog/19>

Eric Dalquist

eric.dalquist@doit.wisc.edu

<http://erics-notes.blogspot.com>