



IBM Software Group

Custom Administration

WebSphere Portal Server v5.1

Lotus software



ON DEMAND BUSINESS™

IBM Proof of Technology

IBM Product Introduction + Exploration

© 2005 IBM Corporation

IBM Software Group | Lotus software

© 2005 IBM Corporation



Session Objectives

- After completing this session, you should be able to:
 - ▶ Be familiar with the XMLAccess Interface
 - ▶ Understand Portal Scripting
 - ▶ Differentiate between Interactive and Scripted modes in Portal Scripting
 - ▶ Modify the “Look and Feel” of the Portal through Themes and Skins



Portal Configuration: XMLAccess

XMLAccess – Configuration Interface

- Provides a batch processing interface for portal configuration updates.
- Accessed using a command line tool XMLAccess
- **Notes:**
 - ▶ A complete XML export of a portal configuration is not sufficient to re-create the portal.
 - ▶ The XML configuration interface is not designed to deal efficiently with large volumes of data.
 - ▶ User must be Administrator and password is passed unencrypted. Use only within own network.

Changes to XMLAccess from v5.0x to v5.1

- WSRP and JSR168 Support
 - ▶ New Portal Resource Type **wsrp-producer**
 - ▶ **web-app** - uid is **portlet-app** + **.webmod**
 - ▶ **portlet-app** - **groupid** (WSRP)
 - ▶ **portlet** - **handle** attribute and **provided** flag (WSRP)
 - ▶ **portletinstance** - use **preferences** instead of **parameters**
 - **Preferences** are name/multi-value
 - **Parameters** are name/value
- Virtual Portals
 - ▶ Only import/export one virtual portal at a time
 - ▶ Only "master" Administrator can import/export virtual portals through XMLAccess
- cleanup-users

Tasks (New in Portal v5.1)

- Tasks - Delayed Page Deletions
 - ▶ Scheduler Bean to schedule when you want to delete pages marked for deletion
 - ▶ Define Bean, dayOfWeek, dayOfMonth, startTime

```
<?xml version="1.0" encoding="UTF-8"?>
<request type="update" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="PortalConfig_1_3.xsd">
  <portal action="locate">

    <!-- This sample creates a scheduler task in the Application Server that will run at the specified date and time
    You can create only one single task that is regularly scheduled
    For example, you cannot create a task that runs both weekly and monthly
    Uncomment one of the following task definitions that best match your needs
    The "run once" task is enabled by default -->

    <!-- Run once, starts immediately -->
    <task action="create" bean="ejb/vpsSchedulerTask"/>

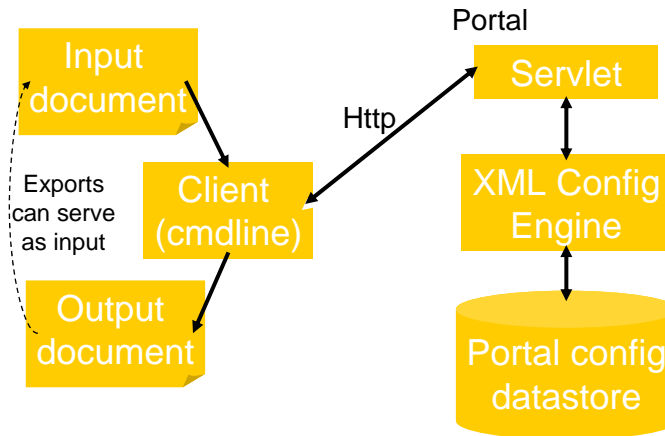
    <!-- Run daily at 12:30 -->
    <task action="create" bean="ejb/vpsSchedulerTask">
      <startTime>12:30</startTime>
    </task>

    <!-- Run weekly on early Friday morning -->
    <task action="create" bean="ejb/vpsSchedulerTask">
      <dayOfWeek>5</dayOfWeek>
      <startTime>7:00</startTime>
    </task>

    <!-- Run monthly at the latest time possible -->
    <task action="create" bean="ejb/vpsSchedulerTask">
      <dayOfMonth>31</dayOfMonth>
      <startTime>23:59</startTime>
    </task>

  </portal>
</request>
```

How does XMLAccess work?



XMLAccess Syntax

- The command line client can be invoked by the xmlaccess script
 - Actual client is contained in the file tools.jar in the <wp_root>/bin directory.
- The syntax for the command is:

xmlaccess

-in	XML_file	<commands to be processed>
-out	result.xml	(opt)< the output file with results of processing>
-user	user	<portal userid to used for authorization>
-pwd	password	<password for user>
-url	PortalConfigURL	<the portal url for processing the command>

example: http://portal.yourco.com/wps/config

The order of the options is of no significance

XMLAccess Input Structure

- The top level structure of an XML request or response is always as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<request xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="PortalConfig_1.3.xsd"
  type="export/update">
  <portal . . . > definition of configuration parts to be exported
or updated
</portal>
  <status . . . > success or failure indication for the processing
</status>
</request>
```

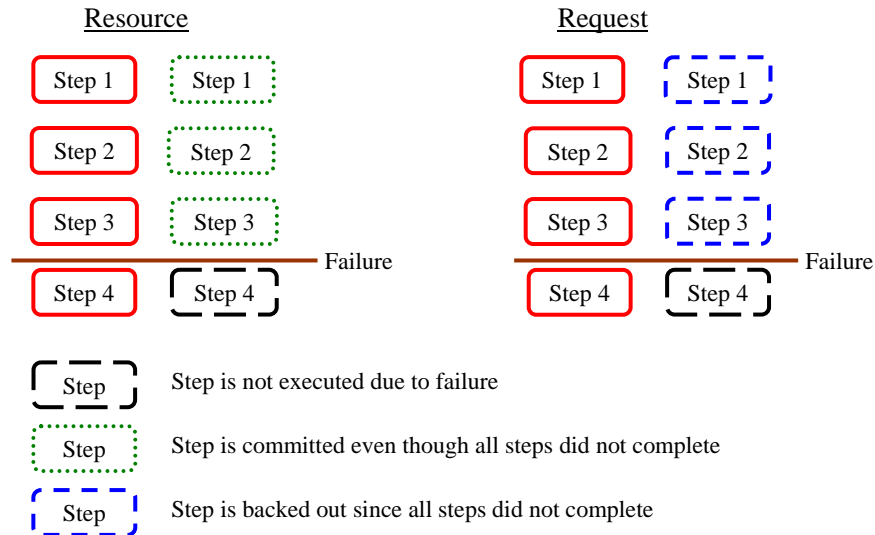
XMLAccess Actions

Action Resulting processing

- locate** Identify the portal resource corresponding to this XML element usually needed as a context for other actions
- create** Create a new portal resource with the given attributes. A new resource is always created, even if another resource with the given name already exists.
- update** Update the configuration of the corresponding portal resource with the given configuration data (attributes and dependent configuration data elements); if no corresponding portal resource can be found, it is created.
- delete** Delete the portal resource corresponding to this XML element.
- export** Include an XML representation of the portal resource corresponding to this element in the output of the XML command.

Action	Allowed actions in subelements
locate	locate, create, update, delete, export
create	locate, create, update
update	locate, create
delete	no subelements allowed
export	no subelements allowed

Transactional Support



Migration Example: Export Portal Configuration

```
<?xml version="1.0" encoding="UTF-8"?>
<request xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"
xsi:noNamespaceSchemaLocation="PortalConfig_1.3.xsd"
type="export">
  <portal action="export"/>
</request>
```

To execute:

```
C:\WebSphere\PortalServer\bin> xmlaccess.bat -in
export.xml -user wpsadmin -pwd wpsadmin -url
http://<hostname>/wps/config -out exportResults.xml
```

Import into New Environment

- Portal Applications (WARS)
 - Must be accessible to the Portal when the "import" is run
 - `<url>file://localhost/C:/myportlets/Special.war</url>`
- Theme Files
 - Theme must be installed as there will be a `themeref="ObjectID"`
- Resulting XML file from previous screen is now imported into the new environment

To execute:

```
C:\WebSphere\PortalServer\bin> xmlaccess.bat -in
exportResults.xml -user wpsadmin -pwd wpsadmin -url
http://<hostname>/wps/config -out exportOut.xml
```

ObjectIDs and XMLAccess

- ObjectID uniquely defines the Portal Resource in the Portal
 - Themes, Skins, Portlet Applications, Pages, Labels, WSRP Producers, etc.
- Uses of the ObjectID
 - Look up a resource
 - Link between resources (`content-parentref=""`)
 - Set ObjectID with `create` and `update`

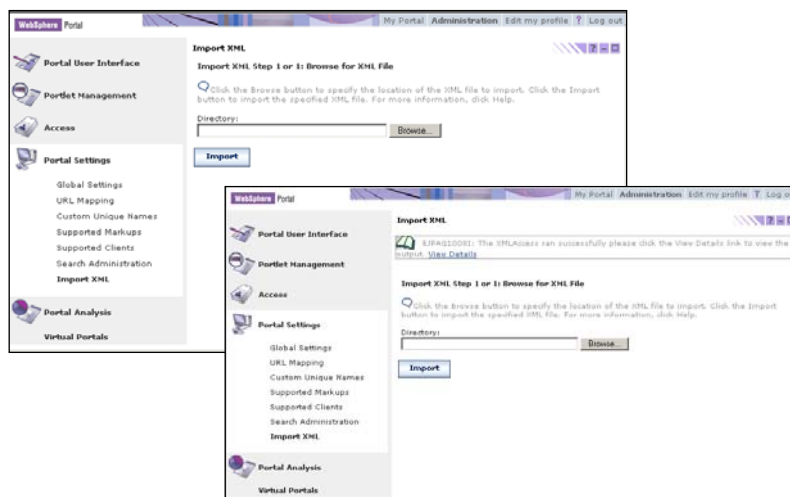
```
<content-node action="update" active="true" content-parentref="_6_0830M8ONSBO115PK_35" objectid="_6_0830M8ONSBO115PK_35" ordinal="1400" themeref="_J_0830M8ONSBO115PK_35" type="label" uniqueness="IT">
  <supported-markup markup="html" update="set"/>
  <localedata locale="en">
    <title>IT</title>
  </localedata>
  <access-control externalized="false" owner="uid=wpsadmin,cn=users,dc=ibm,dc=com" private="false">
    <role-block actionset="Privileged User" type="inheritance"/>
    <role actionset="Privileged User" update="set">
      <mapping subjectid="cn=itgroup" subjecttype="user_group" update="set"/>
    </role>
  </access-control>
</content-node>
```

Custom Unique Names and XMLAccess

- Uniquely identify a Portal Resource in a “user-friendly” way instead of with ObjectID
- Ability to export particular resource without searching for ObjectID
- Example to Export Page Structure for “IT”

```
<?xml version="1.0" encoding="UTF-8"?>
<request
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="PortalConfig_1.2.xsd"
  type="export">
  <portal action="locate">
    <content-node uniqueness="IT" type="label" create-
      type="explicit" export-descendants="true" action="export"/>
  </portal>
</request>
```

Administration Update: Import XML



The screenshot shows the 'Manage Pages' section of a web application. At the top, there are navigation links: 'My Portal', 'Administration', and 'Edit my profile'. Below this is a 'Manage Pages' heading and a help icon. A text box explains the controls: 'Use the controls below to work with your pages. Browse or search for pages to work with. Click New to create new pages, labels and urls, and deactivate pages, re-order, edit properties and layout, move pages, assign permissions and delete. For more information, click Help.' Below this is a search bar with 'Title starts with' selected and a 'Search' button. The 'Content Root' section shows 'My pages' with options to 'Add, Edit, Delete, and Reorder pages'. There are buttons for 'New page' and 'New label'. A table lists the pages. The 'My Portal' page is highlighted. The 'Delete' icon (a trash can) is circled in red.

My Portal **Administration** **Edit my profile**

Manage Pages















Use the controls below to work with your pages. Browse or search for pages to work with. Click New to create new pages, labels and urls, and deactivate pages, re-order, edit properties and layout, move pages, assign permissions and delete. For more information, click Help.

Search by: Search:

Content Root

My pages Add, Edit, Delete, and Reorder pages

Page 1 of 2 Jump to page:

Title	Unique name	Status	
 My Portal	vps.My Portal	Active	   
 Administration	vps.Administration	Active	   
 Hidden Page for Workplace Directory Search	lotus.workplace.hiddenpage.PeoplePicker	Active	     
 Page Customizer	vps.Page Customizer	Active	   
 sunil	6_0_6C	Active	     
 vps.Page Properties	vps.Page Properties	Active	   

18

Portal Scripting

- Portal is accessed with “user identity” and not by the super administrator
- Integrated with “wsadmin” command line tool
 - JACL
 - \$<object> <method> <param1> <param2>
 - Portal Scripting Objects
 - \$Portal, \$Content,
 - \$Layout, \$Portlet, \$Look, \$Access, \$PacList
- Distributed Portal Administration
 - Development Builds Solutions
 - Operations Deploys Solutions
- Interactive vs. Scripted

Interactive Mode

- Interact directly and dynamically with the Portal Server to perform simple admin tasks that only need to be performed once
 - Ex. Add a portlet to a page (one time)
 - <WPS_HOME>\bin

```
wpscript.bat -port 8882 -user wpsadmin -password wpsadmin
```

- Default connection type is SOAP
- Default port for unmanaged cell is 8882
- Log onto Portal

```
wsadmin> $Portal login wpsadmin wpsadmin
```

```
wsadmin>test_var [$Content find all uname "wps.MyTasks"]
6_003124L1PE0CHTJS_6F
wsadmin>Content test_var
6_003124L1PE0CHTJS_6F
wsadmin>Content details
name: wps.MyTasks
id : 6_003124L1PE0CHTJS_6F
type: Composition
no children

wsadmin>foreach test_lay [$Layout search all] ($Layout details $test_lay)
name: Vertical
id : 7_003124L1PE0CHTJS_9P
type: container
ori.: Vertical

wsadmin>
```

Scripted Mode

- Reproducible Administrative tasks
- wpscript activates a JACL script to complete a task
- Can be created by developer and run by operator
- All changes are applied immediately to the Portal Server

wpscript.bat -port 8882 -f testme.jacll wpsadmin wpsadmin

```
C:\WebSphere\PortalServer\bin>wpscript -port 8882 -f wpsl_wps.jacll -user wpsbind -pas
sword wpsbind bookmark
Invoked script for wpsadmin with portal script
[MSX7209] Connected to process "WebSphere_Portal" on node wpSive using SOAP connect
[MSX7209] The type of process is: J2EEProcess
connected to portal 00312X.IPECHTJS_48 on node wpSive in cell wpSive
logged in as "uid=wpsadmin,o=default,organization"
Example Script 2 started and login completed :->
name: BookmarksApp
id : 1_1_00312X.IPECHTJS_48
type: webmodule
dir : Bookmarks.war
version 0.0
name: Bookmarks
id : 2_2_00312X.IPECHTJS_48
type: Application
name: Bookmarks
id : 3_3_00312X.IPECHTJS_48
type: portlet
name: Bookmarks
id : 4_4_00312X.IPECHTJS_48
type: servlet
version 0.0
Preparing to logout ...
C:\WebSphere\PortalServer\bin>
```

Example: Portal Scripting

```
$Content select the root
$Content select [$Content create label WorkPlace]
$Content select [$Content create label WorkAtHome]
$Content create page IdlingAtHome
$Content create page WorkingAtHome
$Content select the parent
$Content select [$Content create label
WorkInOffice]
$Content create page IdlingInOffice
$Content create page WorkingInOffice
```

Create a new content "tree" with pages



Portal Scripting vs. XMLAccess

- Script can access the Portal as any user not just as a "super administrator"
- "Fine-tuned" administration
- Easier automation of repetitive administrative tasks
 - Ex. "everytime we find a portlet named "Weather" drop a "StockQuote" portlet next to it"

My Stock

Thursday, November 21, 2002 2:51:00 PM EST

IBM	84.31	2.70	3.31%
\$COMPQ	1,465.53	46.18	3.25%
\$DJI	8,836.10	213.10	2.47%
\$SPX	933.17	19.02	2.08%
\$N225	8,668.06	208.44	2.46%
\$GDAX	3,212.99	0.00	0.00%

Data delayed at least 20 minutes.

[Symbol Lookup](#)

Copyright 2002 Pinnacor Inc.

My Weather

Thursday, November 21, 2002 17:18 GMT

	Raleigh, North Carolina	56° Overcast
	Stuttgart, Germany	50° RainShower
	Beijing (Peking), China	33° Snow
	Tokyo, Japan	49° Partly Cloudy
	Iowa City, Iowa	38° Overcast

Themes and Skins

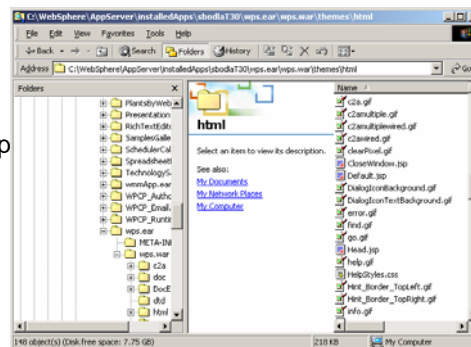
Customization through Themes and Skins

- Provides Portal “Look and Feel”
 - ▶ Branding
 - ▶ Navigation
- Decoration
- Applies to highest level Label or Page
- Not User-centric
- Portal Ships with several themes and skins
 - ▶ Lightweight Theme



Contents of a Theme

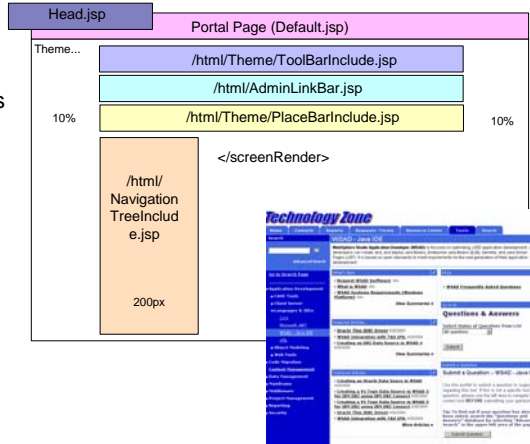
- JSPs
 - ▶ Default.jsp,
 - ▶ PlaceBarInclude.jsp,
 - ▶ PageBarInclude.jsp,
 - ▶ ToolBarInclude.jsp,
 - ▶ NavigationTreeInclude.jsp
- Images
 - ▶ Brand Images, Portal Navigation, bullets
- Cascading Style Sheets
 - ▶ Styles.css, Help.css



<WAS_ROOT>/installedApps/wps/wps.ear/wps.war/themes/html

Theme Layout

- Default.jsp
 - acts as the “master” for layout of the Theme files
- Head.jsp
 - contains references to CSS to Javascript.
<TITLE> tags
- NavigationTreeInclude.jsp, PlaceBarInclude.jsp, PageBarInclude.jsp
 - handle Navigation
- ToolBarInclude.jsp and AdminLinkBar.jsp
 - handle links for Log In/Administration/etc



Theme Layout: Portal Navigation

- Page navigation is divided into different “node” levels based on how you have designed your Portal pages.
 - Navigation Level 1 is “My Portal”
- Navigation can be Horizontal, Vertical, DropDown, MouseOver, DHTML, etc.
- Tags exist that allow you to manipulate how this navigation is visible to your users


```
<wps:navigation startLevel="2" stopLevel="2">
  <wps:navigationLoop>
```
- Dynamically loops through pages and displays in Theme using the “Model API”

Disable Theme Caching

- By default the Portal Theme will be cached. As a result, changes to dynamic elements will not be reflected until the cache is cleared. There are two ways to do this:
 - Change the file:

```
<WAS_ROOT>\config\cells\<node>\application\wps.ear  
  \deployments\wps.war\WEB-INF\ibm-web-ext.xml
```

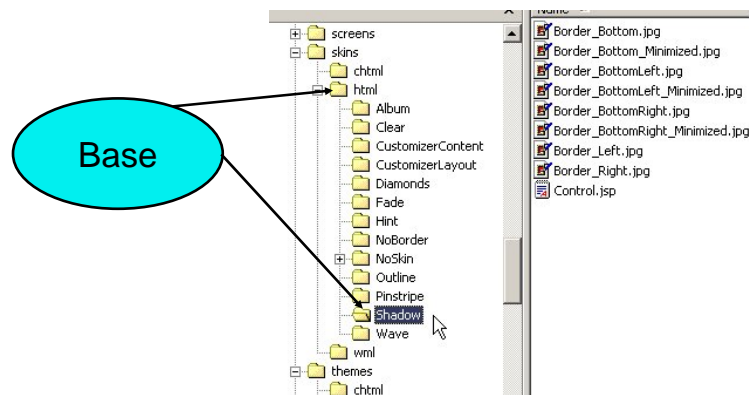
Set reloadingEnabled="true" (two places)

Restart Portal.

- Delete the /temp directory
- Touch Default.jsp

Contents of a Skin

- JSPs (Control.jsp)
- Images (Portlet Controls, Decorations, Preview image, etc..)



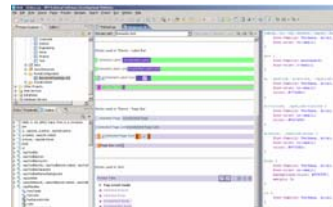
Resolving Theme and Skin Assets

- Themes and Skins are searched starting with the most specific subdirectory and moving up to the more general, higher-level directory
- Aggregation

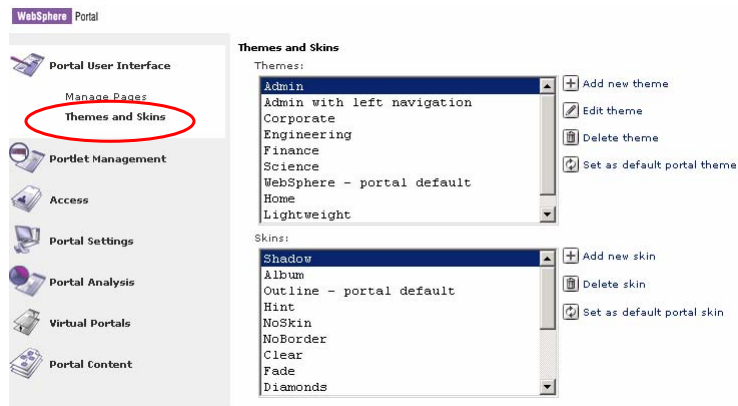
Location in the /themes directory	Location in the /skins directory
1. /locale_region	1. /locale_region
2. /locale	2. /locale
3. /client	3. /client
4. /theme_name	4. /skin_name
5. /markup	5. /markup

Theme/Skin Development Options

- Any Text Editor
 - Easier for those already familiar with Theme development
 - More flexible, yet more room for error
- Rational Application Developer v6.0
 - Provide Portal Toolkit as part of the Install
 - Can create a new Portal Project (v5): Themes, Skins, etc.
 - Visual Design capability for Theme development



Themes and Skins Administration



Session Summary

- Now that you have completed this session, you should be able to:
 - ▶ Understand XMLAccess
 - ▶ Have the ability to perform administrative tasks either through XMLAccess or Portal Scripting
 - ▶ Understand when you would need to use each
 - ▶ Be able to customize a Portal to use your own Themes and Skins
 - ▶ Understand how to deploy a Theme and Add Skins to that Theme

Lab Exercise 1.4

- Lab 1.4 – Custom Administration
 - ▶ Create a Custom Unique Name to be used with XMLAccess
 - ▶ Import a page navigation for “IT”
 - ▶ Add portlets to pages under the “IT” label
 - ▶ Export the entire configuration for the Portal
 - ▶ Customize a Theme to use “dynamic stylesheets”
 - ▶ Install the Theme and add it to the AJRSToys pages