CCLRC
Daresbury Laboratory

# Experiences with WSRP Testing and Development

Xiaobo Yang, Xiao Dong Wang and Rob Allan

*CCLRC e-Science Centre*
*Daresbury Laboratory*

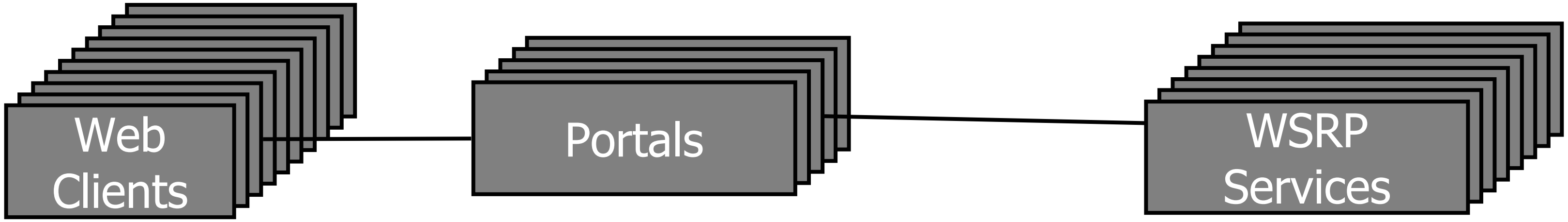
18th January 2006, Portsmouth, UK

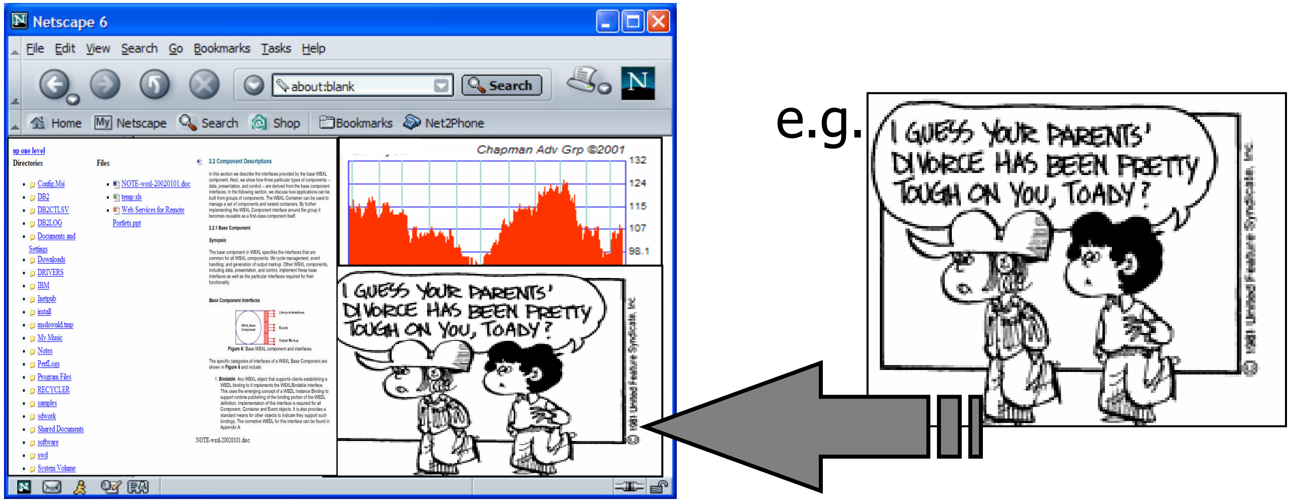
# Overview

- What and why WSRP?
- WSRP support test of selected open-source portal frameworks
- Development of WSRP consumer for Sakai
- Conclusion remarks

# What is WSRP?

- WSRP ( Web Services for Remote Portlets )
  - An OASIS standard to solve the interoperability issue between portal frameworks
  - Defines presentation-based and interactive Web Services




Netscape 6
File Edit View Search Go Bookmarks Tasks Help
about:blank
Search
Home
My Netscape
Search
Shop
Bookmarks
Net2Phone
Chapman Adv Grp ©2001
132
124
115
107
98.1
I GUESS YOUR PARENTS' DIVORCE HAS BEEN PRETTY TOUGH ON YOU, TOADY?
© 1981 United Feature Syndicate, Inc.

e.g.

# Why WSRP?

- Why WSRP
  - Plug and play using Web services
  - Deploy service once, reuse anywhere
  - Reuse both business logic and presentation
  - Avoids multiple copies of code
  - Integrate third-party portlets from remote servers
  - Transparent to end users and portal administrators
  - Allows local maintenance and content aggregation
  - Look and feel consistent even from remote portlets

CCLRC
Daresbury Laboratory

# WSRP Test of Selected Open-Source Portal Frameworks

# Who Supports WSRP?

- Open-Source Community Portal Frameworks
  - eXo Platform v1.0
  - Liferay Enterprise/ Professional v3.6.1
  - StringBeans v3.0
  - uPortal v2.4 (soon to be v3.0)
  - Sakai v2.1.0
  - WSRP4J
  - …
- Commercial Vendors
  - BEA WebLogic Portal v8.1
  - IBM WebSphere Portal Enable/ Extend for Multiplatforms v5.1
  - Sun One Portal v6.2
  - …
- All support WSRP Consumer and Producer, except uPortal v2.4 only has WSRP Consumer support and Sakai v2.1.0 only has Producer

# Why Investigation?

- WSRP support is confusing
  - WSRP consumer support without producer, e.g. uPortal v2.4
  - WSRP producer support without consumer, e.g. wsrp4j + pluto, Sakai v2.1.0
  - WSRP consumer and producer, e.g. eXo v1.0, Liferay Enterprise/Professional v3.6.1, StringBeans v3.0
    - Do they inter-operate?
- Investigate is based on our previous experience in portal framework evaluation
  - Interoperability issue between different portal frameworks
  - Ease of use
  - Functionality
  - Stability
  - Security
  - …

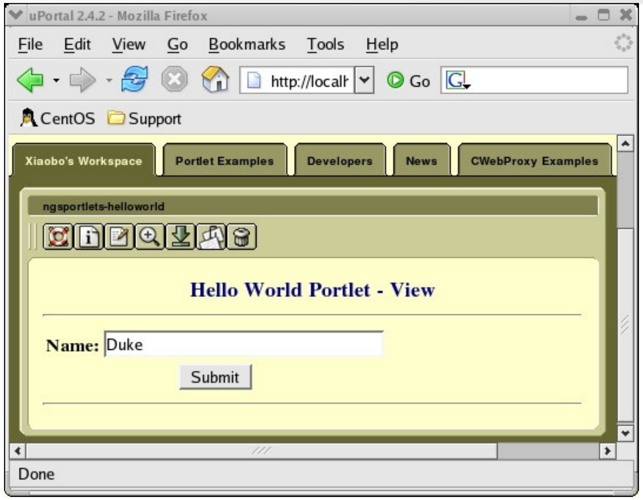
# Investigation Scenarios

- Popular Portal frameworks selected as used to support e-Research
  - eXo platform v1.0
  - Liferay professional v3.6.1
  - StringBeans v3.0
  - uPortal v2.4.2
  - WSRP4J (CVS download)

Aims to test all combinations!

- Test portlets selected
  - HelloWorld portlet
    - Text input box, button, text message feedback, javascript URL, view/help/edit mode
  - LdapBrowser portlet
    - Portlet URL, image URL

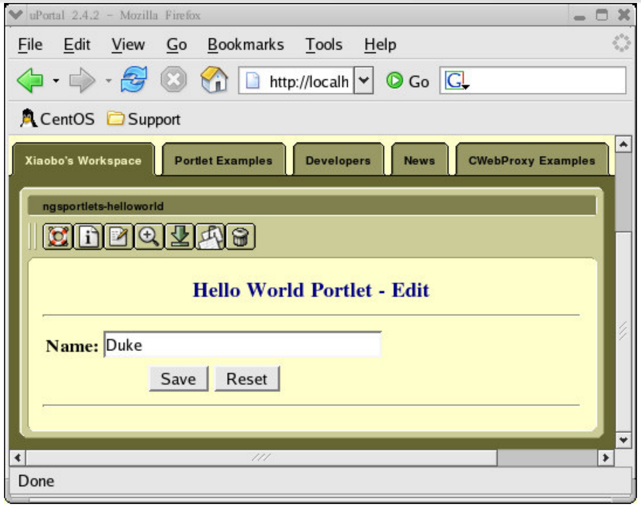
# HelloWorld Portlet



**Hello World Portlet - View**

**Name:** Duke

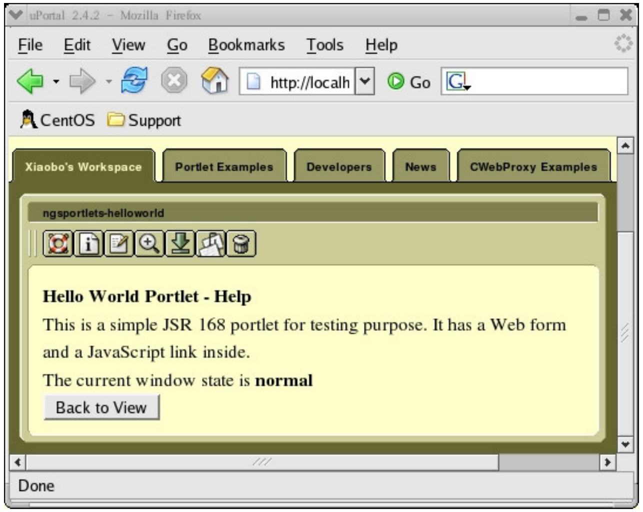
*view* mode



**Hello World Portlet - Edit**

**Name:** Duke

*edit* mode

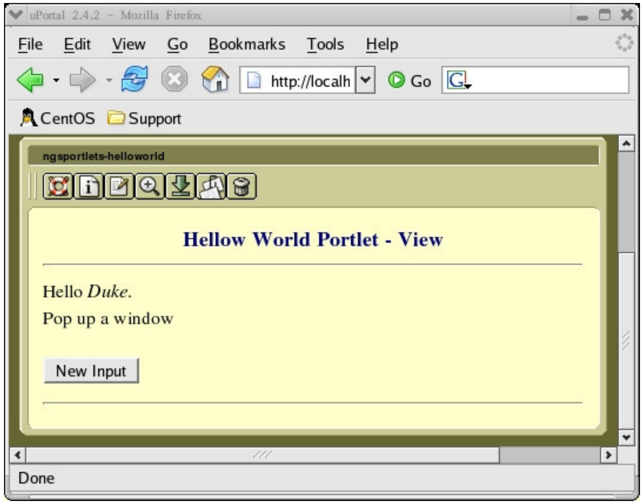


**Hello World Portlet - Help**

This is a simple JSR 168 portlet for testing purpose. It has a Web form and a JavaScript link inside.

The current window state is **normal**

*help* mode


uPortal 2.4.2 - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://localh
Go
CentOS
Support
ngsportlets-helloworld
New Input
Done

**Hellow World Portlet - View**

Hello *Duke*.

Pop up a window

button “Submit” clicked

# Test Results

- Producer: WSRP4J
  - Consumer: eXo platform
    - Able to list remote portlets
    - No *edit* mode support
    - HelloWorld portlet works fine
    - LdapBrowser portlet only works for first query, further query through portlet URL or image URL does not work, image was not displayed
  - Consumer: Liferay
    - Able to list remote portlets
    - *Edit* mode is functional
    - *Help* mode doesn't work – blank screen
    - HelloWorld and LdapBrowser portlets work fine
    - Image is not displayed
  - Consumer: StringBeans
    - Able to list remote portlets
    - No action can be taken
    - Server must be restarted when a new producer is created
  - Consumer: uPortal
    - Unable to list remote portlets
    - Portlet handle must be defined to publish a remote portlet
    - Can not go back to *view* mode after entering *edit* or *help* mode
    - Both portlets work fine
    - Image was not displayed

# Test Results (continued)

- Producer: eXo Platform
  - Consumer: eXo platform
    - Same as WSRP4J acting as producer
  - Consumer: Liferay
    - Fails to connect to eXo producer
    - Reason: SOAP message shows Liferay fails to execute the getServiceDescription( ) operation
  - Consumer: StringBeans
    - Able to list remote portlets with markupURL and serviceDescriptionURL defined plus registration handle (extracted from eXo to eXo SOAP message) provided
    - No action can be taken
  - Consumer: uPortal
    - Fails to connected to eXo producer
    - Reason: remote portlet handle can not be defined

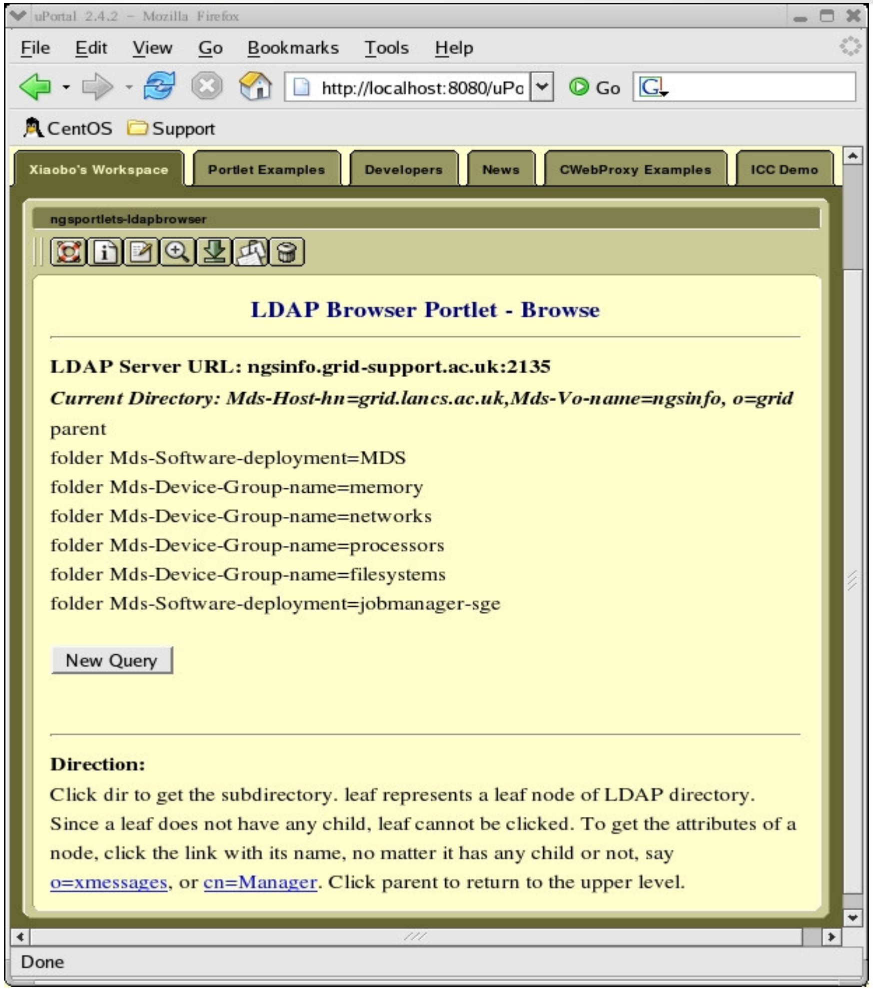
# Test Results (continued)

- Producer: Liferay
  - Consumer: eXo platform
    - Fails to retrieve remote portlet list although a producer is created
  - Consumer: Liferay
    - Able to list remote portlets but all items are empty
    - The markup fragment of HelloWorld and LdapBrowser portlet can be retrieved and displayed but no action can be taken
  - Consumer: StringBeans
    - Able to list remote portlets
    - No action can be taken
    - Always displays the default page
  - Consumer: uPortal
    - Using the portlet handle hidden in the portlet list, uPortal can load the default *view* page but no action can be performed

# Test Results (continued)

- Producer: StringBeans
  - Consumer: eXo platform
    - Able to list remote portlets
    - Can get markup fragment of the HelloWorld and LdapBrowser portlets
    - Different modes can be switched but no action is replied correctly
  - Consumer: Liferay
    - Able to list remote portlets
    - HelloWorld and LdapBrowser portlet can be displayed but no action is responded correctly, the same page is always displayed
  - Consumer: StringBeans
    - Able to list remote portlets
    - HelloWorld portlet works fine except sometimes buttons are not correctly handled
    - LdapBrowser portlet behaves similar to eXo to eXo, i.e., portlet URL and image URL does not work
  - Consumer: uPortal
    - Fails to connected to StringBeans producer
    - Portlet handle can not be defined

# Display Images



### LDAP Browser Portlet - Browse

**LDAP Server URL: ngsinfo.grid-support.ac.uk:2135**

***Current Directory: Mds-Host-hn=grid.lancs.ac.uk,Mds-Vo-name=ngsinfo, o=grid***

parent

folder Mds-Software-deployment=MDS

folder Mds-Device-Group-name=memory

folder Mds-Device-Group-name=networks

folder Mds-Device-Group-name=processors

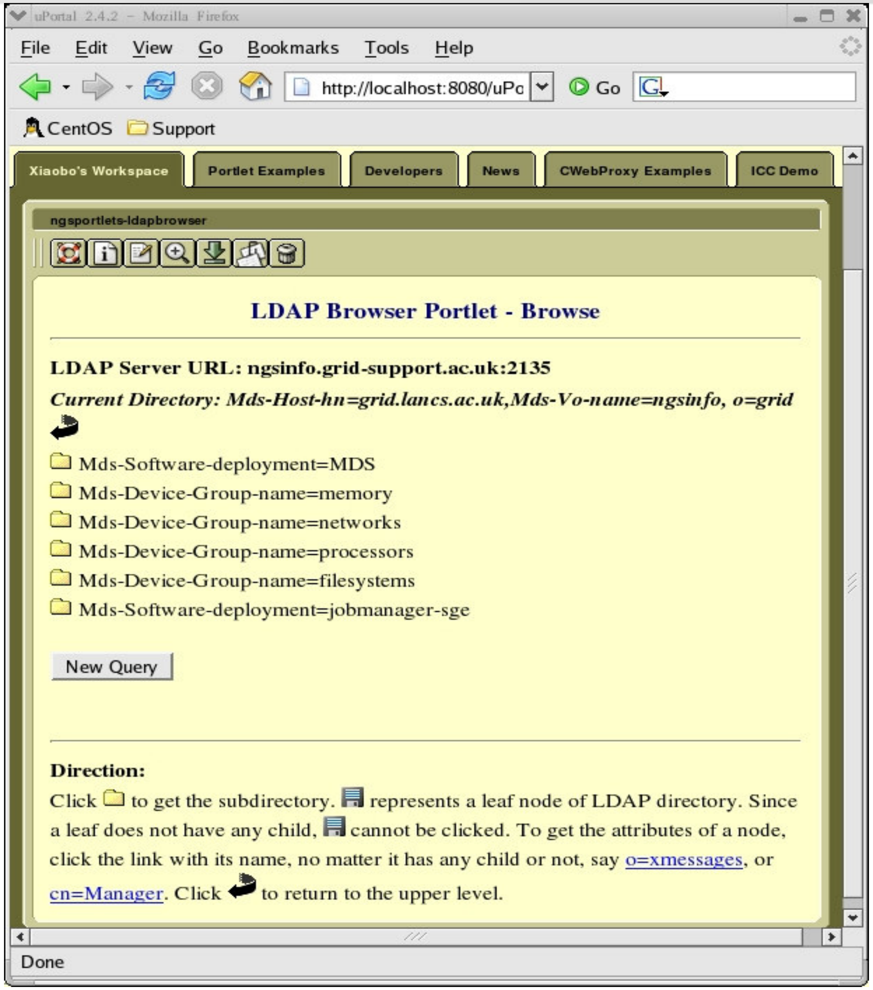
folder Mds-Device-Group-name=filesystems

folder Mds-Software-deployment=jobmanager-sge

New Query

**Direction:**

Click dir to get the subdirectory. leaf represents a leaf node of LDAP directory. Since a leaf does not have any child, leaf cannot be clicked. To get the attributes of a node, click the link with its name, no matter it has any child or not, say o=xmessages, or cn=Manager. Click parent to return to the upper level.



### LDAP Browser Portlet - Browse

**LDAP Server URL: ngsinfo.grid-support.ac.uk:2135**

***Current Directory: Mds-Host-hn=grid.lancs.ac.uk,Mds-Vo-name=ngsinfo, o=grid***

↩

📁 Mds-Software-deployment=MDS

📁 Mds-Device-Group-name=memory

📁 Mds-Device-Group-name=networks

📁 Mds-Device-Group-name=processors

📁 Mds-Device-Group-name=filesystems

📁 Mds-Software-deployment=jobmanager-sge

New Query

**Direction:**

Click 📁 to get the subdirectory. 💾 represents a leaf node of LDAP directory. Since a leaf does not have any child, 💾 cannot be clicked. To get the attributes of a node, click the link with its name, no matter it has any child or not, say o=xmessages, or cn=Manager. Click ↩ to return to the upper level.

Relative image URLs replaced by absolute URLs in uPortal's consumer

# Summary

- WSRP implementation is still at an early stage, further development is clearly needed
- Producer comparison:
  - WSRP4J producer can be accessed by all selected WSRP consumers
  - eXo platform producer is accepted by eXo and StringBeans (with careful settings)
  - Liferay producer is far away from being mature since none of the consumers connected to it is functional
  - StringBeans producer is only functional with another StringBeans portal consuming it
- Consumer comparison:
  - None of the consumers is fully functional
  - eXo platform consumer has no *edit* mode support and has problems handling portlet/image URLs
  - Liferay consumer has problem in supporting *help* mode
  - StringBeans consumer does not work with producers other than StringBeans. Also it has problem handling portlet/image URLs. Each time a new producer is defined, server must be restarted
  - uPortal consumer needs portlet handler to publish a channel

# Test Summary

Based on the above results:

| *Producer* <br> *Consumer* | eXo Platform | Liferay | StringBeans | uPortal | WSRP4J |
|---|---|---|---|---|---|
| eXo Platform | √ | × | × | | √ |
| Liferay | × | × | × | | √ |
| StringBeans | √ | × | √ | | √ |
| uPortal | × | × | × | | √ |

Nothing is fully functional!

CCLRC
Daresbury Laboratory

# Development of WSRP Consumer for Sakai

# Requirements

- A lot of portlets are now available to use, how to integrate them into the Sakai framework?
  - A set of Grid portlets developed for the NGS Portal
  - Sakai VRE Demonstrator has some communication portlets
  - Portlets come from portal frameworks
  - Open-source community

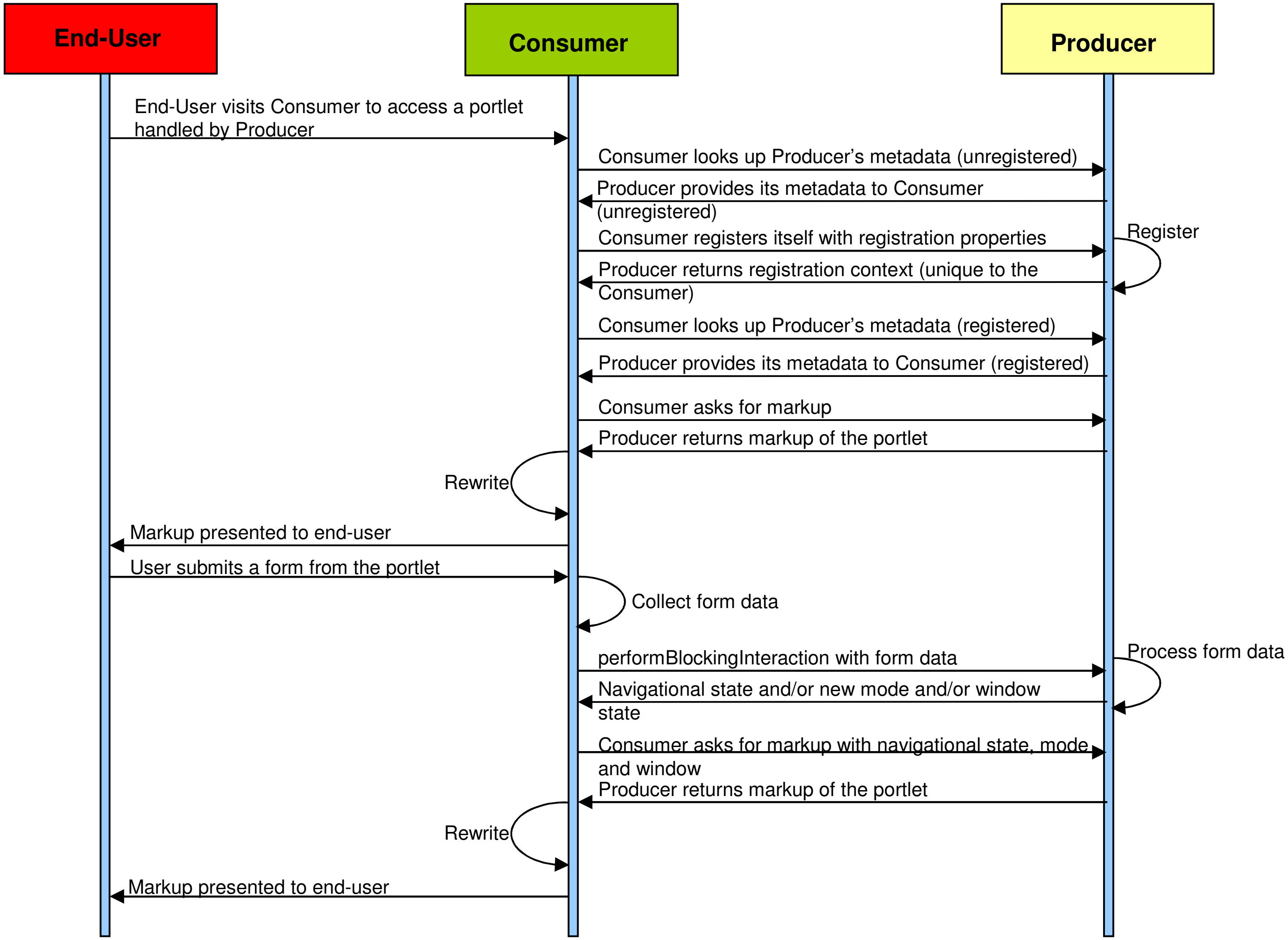
- We want to make use of both business logic and presentation layer
  - Re-write presentation layer is time-consuming

# WSRP Interfaces

- WSRP 1.0 defines four interfaces
  - ServiceDescription: a *required* interface provides meta-data for a consumer to interact with each portlet the producer hosts. It also gives information about the producer's capabilities.
  - Markup: a *required* interface for interacting with user requests and generating markup fragments.
  - Registration: an *optional* interface to set up a relationship between a producer and a consumer.
  - PortletManagement: an *optional* interface used for both managing the life-cycle of hosted portlets and portlets' persistent state.

CCLRC Daresbury Laboratory

# WSRP Sequence Flow


End-User

Consumer

Producer

End-User visits Consumer to access a portlet handled by Producer

Consumer looks up Producer’s metadata (unregistered)

Producer provides its metadata to Consumer (unregistered)

Consumer registers itself with registration properties

Register

Producer returns registration context (unique to the Consumer)

Consumer looks up Producer’s metadata (registered)

Producer provides its metadata to Consumer (registered)

Consumer asks for markup

Producer returns markup of the portlet

Rewrite

Markup presented to end-user

User submits a form from the portlet

Collect form data

performBlockingInteraction with form data

Process form data

Navigational state and/or new mode and/or window state

Consumer asks for markup with navigational state, mode and window

Producer returns markup of the portlet

Rewrite

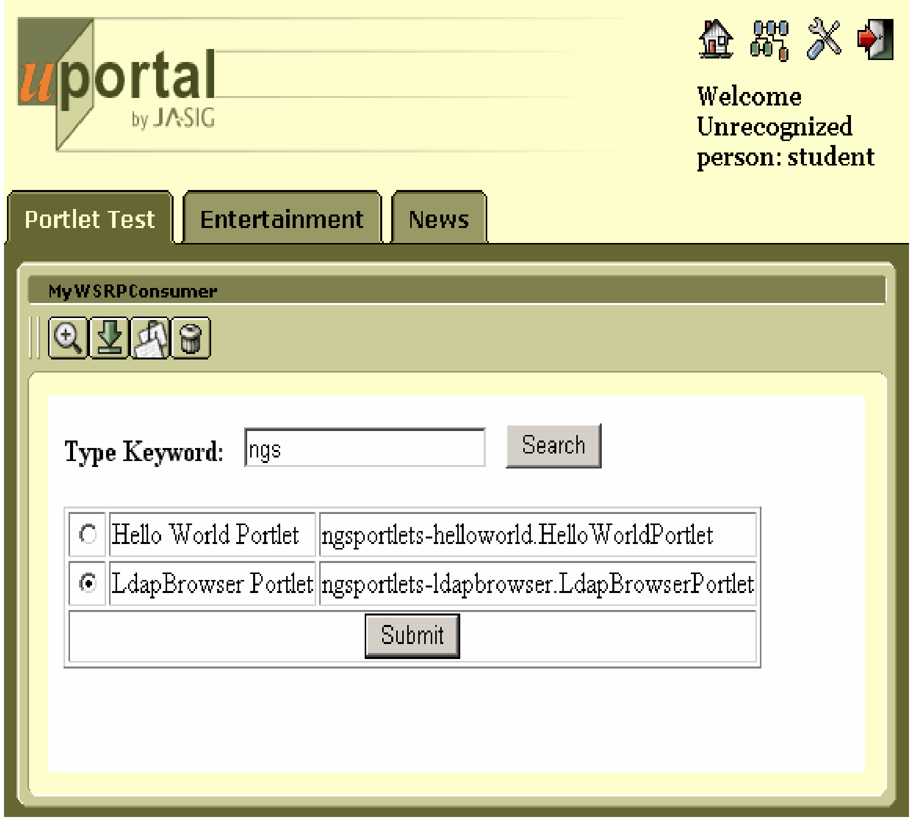
Markup presented to end-user

VRE Developers Workshop
18th January 2006, Portsmouth, UK

# Development of WSRP Consumer

- WSRP consumer is a broker sitting between producer and client
  - User interactions must be re-directed to producer as consumer is not responsible for handling such interactions
- A Web-based WSRP consumer has been developed and tested with WSRP4J producer
  - Developed as a Java servlet on top of WSRP4J consumer (ProxyPortlet)
  - Four interfaces implemented
  - UDDI registration tested
    - Portlet and producer’s information published in UDDI
    - Client queries UDDI registry to get a list of portlets meet its requirement
    - Client selects a portlet which in turn calls the WSRP consumer

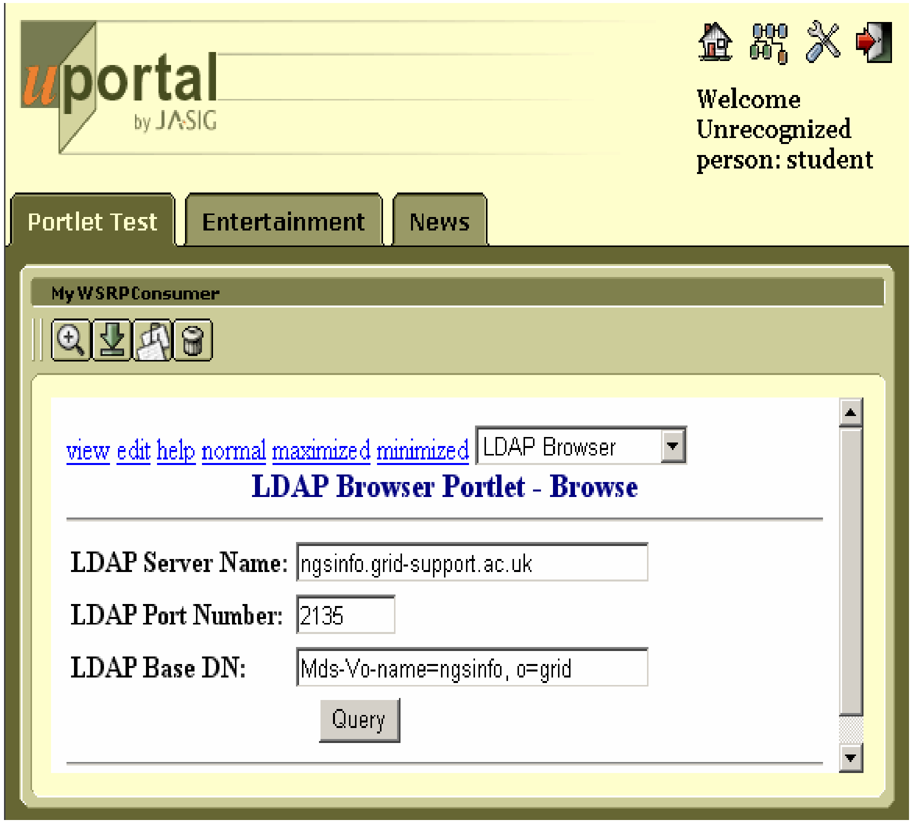
# UDDI for Portlet & Producer Registration



Welcome Unrecognized person: student

Type Keyword: ngs

| | | |
|---|---|---|
| ○ | Hello World Portlet | ngsportlets-helloworld.HelloWorldPortlet |
| ● | LdapBrowser Portlet | ngsportlets-ldapbrowser.LdapBrowserPortlet |



Welcome Unrecognized person: student

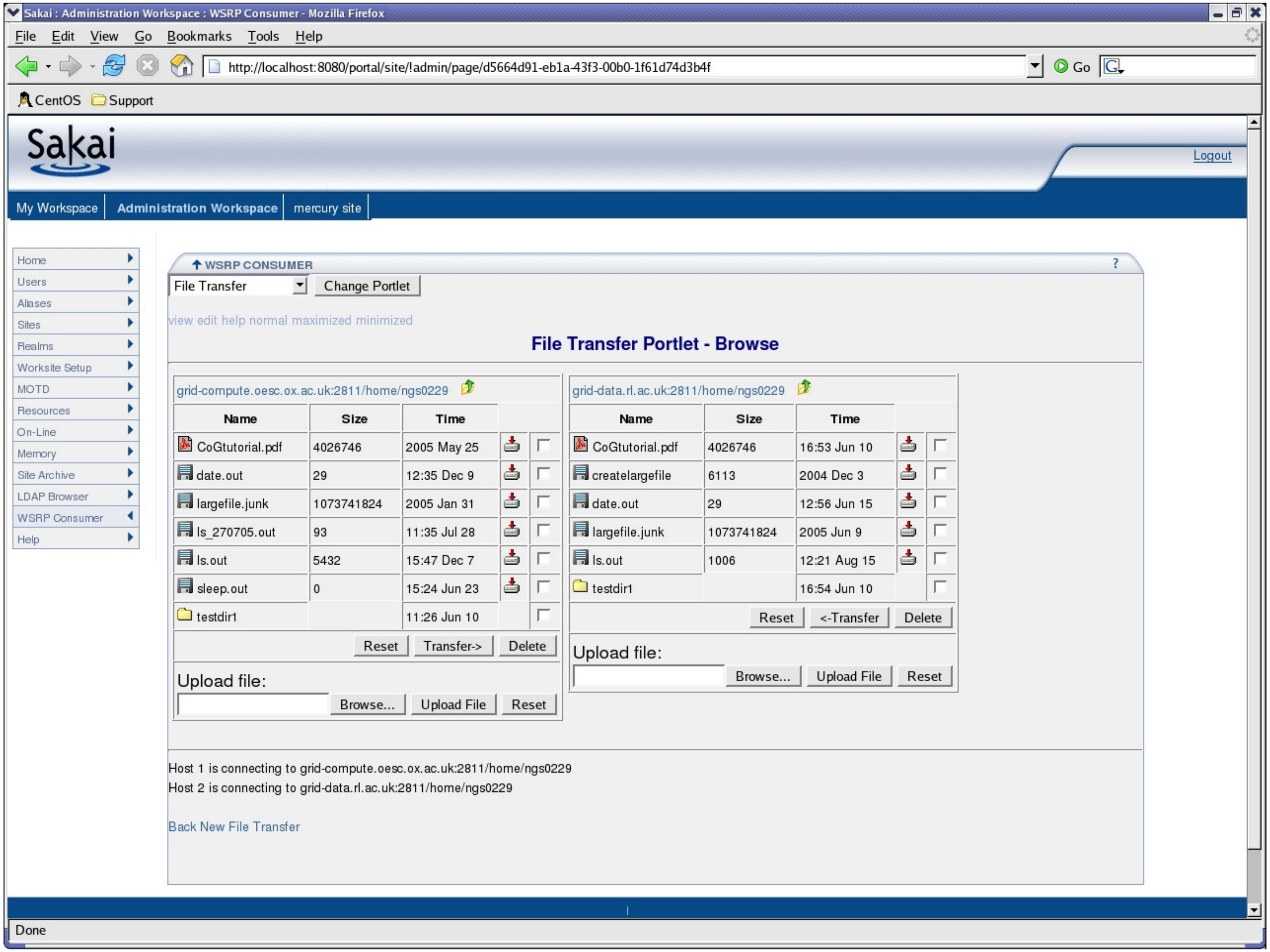
**LDAP Browser Portlet - Browse**

LDAP Server Name: ngsinfo.grid-support.ac.uk

LDAP Port Number: 2135

LDAP Base DN: Mds-Vo-name=ngsinfo, o=grid

# FileTransfer Portlet Running in Sakai


Sakai : Administration Workspace : WSRP Consumer - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://localhost:8080/portal/site/!admin/page/d5664d91-eb1a-43f3-00b0-1f61d74d3b4f
Go
CentOS
Support
Sakai
Logout
My Workspace
Administration Workspace
mercury site
Home
Users
Aliases
Sites
Realms
Worksite Setup
MOTD
Resources
On-Line
Memory
Site Archive
LDAP Browser
WSRP Consumer
Help
WSRP CONSUMER
?
File Transfer
Change Portlet
view edit help normal maximized minimized
Done

File Transfer Portlet - Browse

grid-compute.oesc.ox.ac.uk:2811/home/ngs0229

| Name | Size | Time | | |
|---|---|---|---|---|
| CoGtutorial.pdf | 4026746 | 2005 May 25 | | □ |
| date.out | 29 | 12:35 Dec 9 | | □ |
| largefile.junk | 1073741824 | 2005 Jan 31 | | □ |
| ls_270705.out | 93 | 11:35 Jul 28 | | □ |
| ls.out | 5432 | 15:47 Dec 7 | | □ |
| sleep.out | 0 | 15:24 Jun 23 | | □ |
| testdir1 | | 11:26 Jun 10 | | □ |

grid-data.rl.ac.uk:2811/home/ngs0229

| Name | Size | Time | | |
|---|---|---|---|---|
| CoGtutorial.pdf | 4026746 | 16:53 Jun 10 | | □ |
| createlargefile | 6113 | 2004 Dec 3 | | □ |
| date.out | 29 | 12:56 Jun 15 | | □ |
| largefile.junk | 1073741824 | 2005 Jun 9 | | □ |
| ls.out | 1006 | 12:21 Aug 15 | | □ |
| testdir1 | | 16:54 Jun 10 | | □ |

Host 1 is connecting to grid-compute.oesc.ox.ac.uk:2811/home/ngs0229
Host 2 is connecting to grid-data.rl.ac.uk:2811/home/ngs0229

Back New File Transfer

# Issues

- Security
  - How to authenticate client?
  - How to apply authorisation policy?
  - How to make use of UDDI efficiently?
  - …
- Inter-portlet communication
  - Will WSRP 2.0 solve the issue?
- Customisation
  - Provides contents according to client’s capability
- WSRP client needs to be a real broker
  - How to support file upload/download
- Interoperable?
  - WSRP consumers from different vendors have a lot of issues talking to WSRP producers provided by other vendors

# Summary

- Based on WSRP4J, a WSRP consumer has been written for Sakai to integrate existing portlets
- UDDI registry has been explored to store portlet and producer information
- There are many issues to be solved (?WSRP 2.0)

# Conclusion Remarks

- WSRP makes it possible for VREs to easily integrate existing portlets
- UDDI can be utilised to store portlet and producer’s information
- There are still many issues with WSRP – interoperability and security are two biggest issues (Will WSRP 2.0 solve these issues?)
- Interoperability must be solved so that consumers can access producers provided by different vendors
- Security is needed to protect remote portlets

CCLRC
Daresbury Laboratory

# Questions?

Thank you!