



CREE and Portlet Standards

Robert Sherratt
University of Hull

Jon Hunter
EDINA



THE UNIVERSITY OF HULL



Center of
Excellence

Introduction

- The CREE Project
- Portlet standards
- CREE development and testing
- Future steps



CREE

- Contextual Resource Evaluation Environment
 - Part of the JISC Portals Programme
- Completed project - Feb 04 - Sep 05
- Evaluating the presentation and use of Internet search tools in portal and non-portal contexts
- Two strands
 - Technology development and testing
 - User requirements gathering and analysis



CREE: a partnership

- e-SIG, University of Hull
- OUCS, University of Oxford
- EDINA, University of Edinburgh
- Archaeology Data Service, University of York
- Newark & Sherwood College
- instructional media + magic, inc.



Search tools in context

- Usually accessed via home website
- CREE examined access in different contexts, away from home website
 - Access via a local, e.g., library, web page
 - Access via a virtual learning environment
 - Access via an institutional portal
- Taking search tools to the user embedded in the portal



CREE and portlet standards

- CREE examined the use of search tools alongside other portal functionality in context
- Standards have emerged to allow common methods for portlet design and structure
 - JSR 168
 - WSRP (Web Services for Remote Portlets)
- CREE has tested these and examined the relationship between them



Two standards - two roles

- Both JSR 168 and WSRP aim to facilitate access to services across portal frameworks
- JSR 168 - local services
- WSRP - remote services
- Can work together within the same portal framework

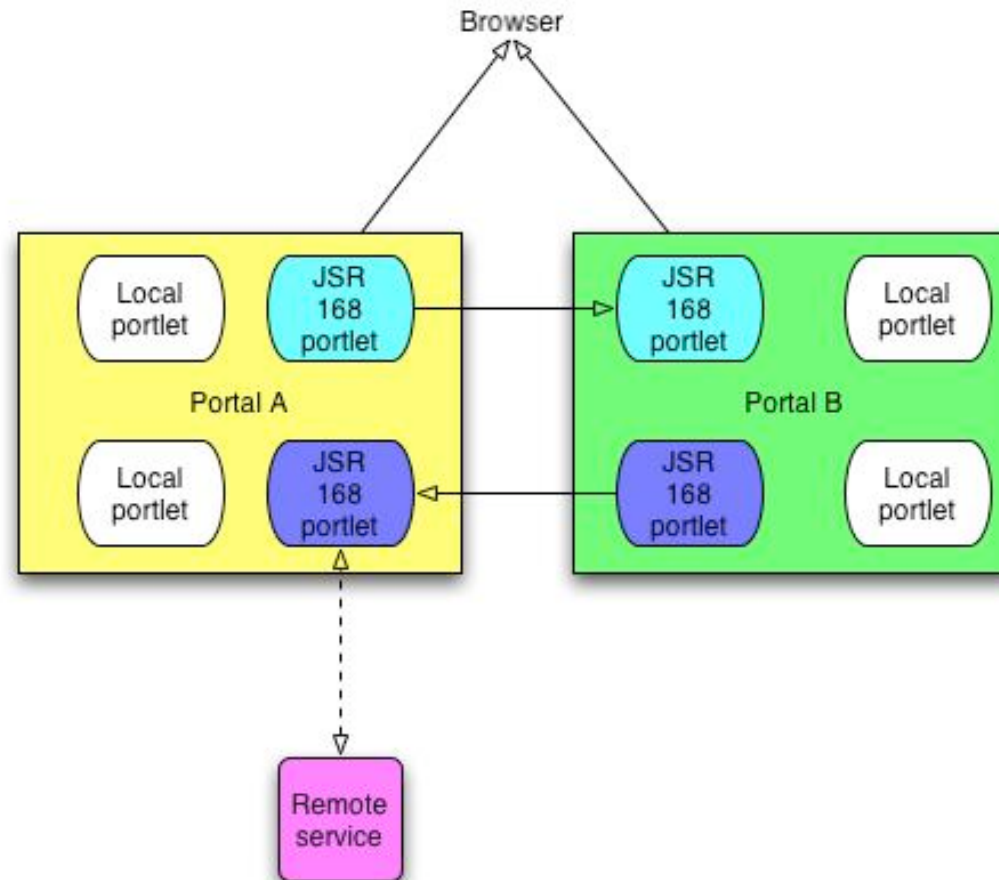


JSR 168

- Defines how a portlet should communicate with the portal framework/container
- For use with Java-based portal frameworks
 - Can move portlets between different portals
- Portlets usually adopt look and feel of portal
- Can access to remote services via HTTP, Z39.50 or other
 - Support/performance issues



JSR 168

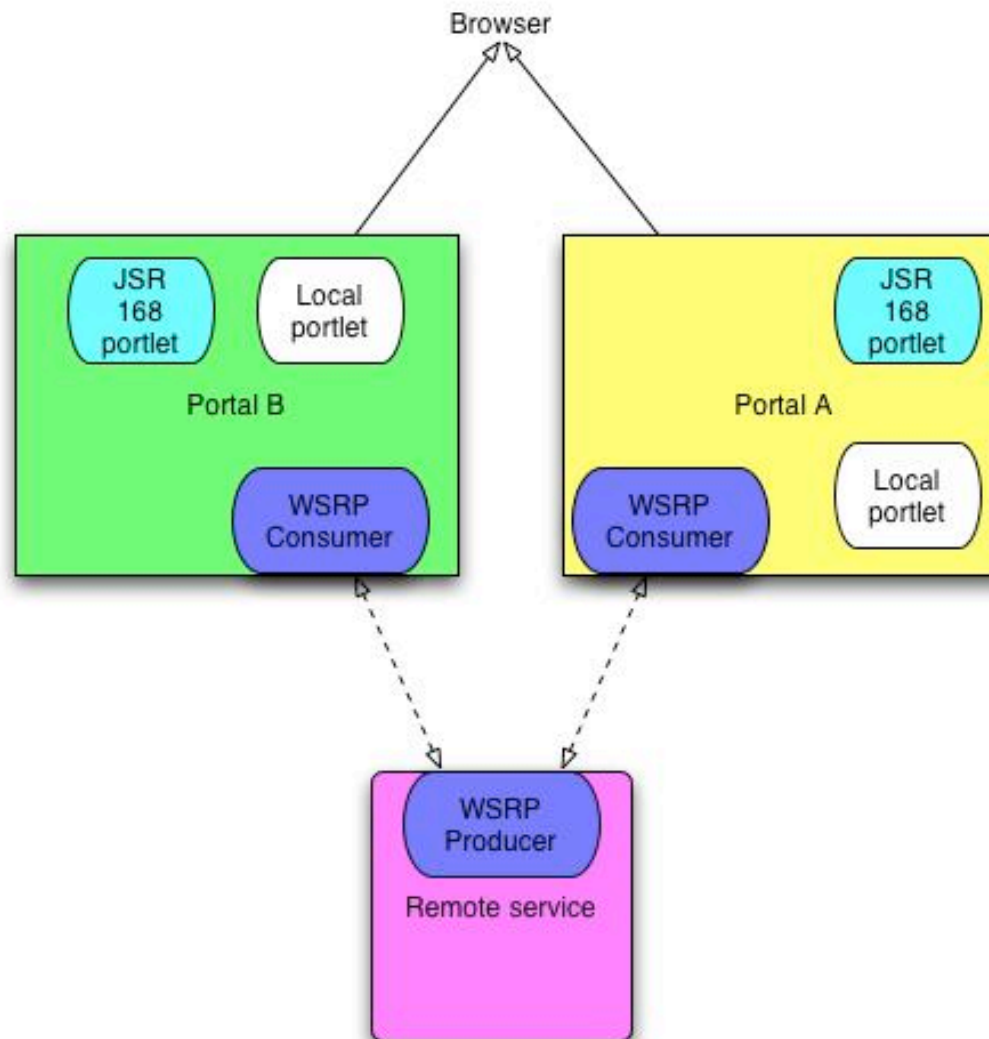


WSRP

- Defines how a portal should communicate with a remote portlet provider
- Language and platform independent
 - Can make portlets available to conformant portals
- Remote service can dictate look and feel of portlet
- Uses Web service protocol for communication



WSRP

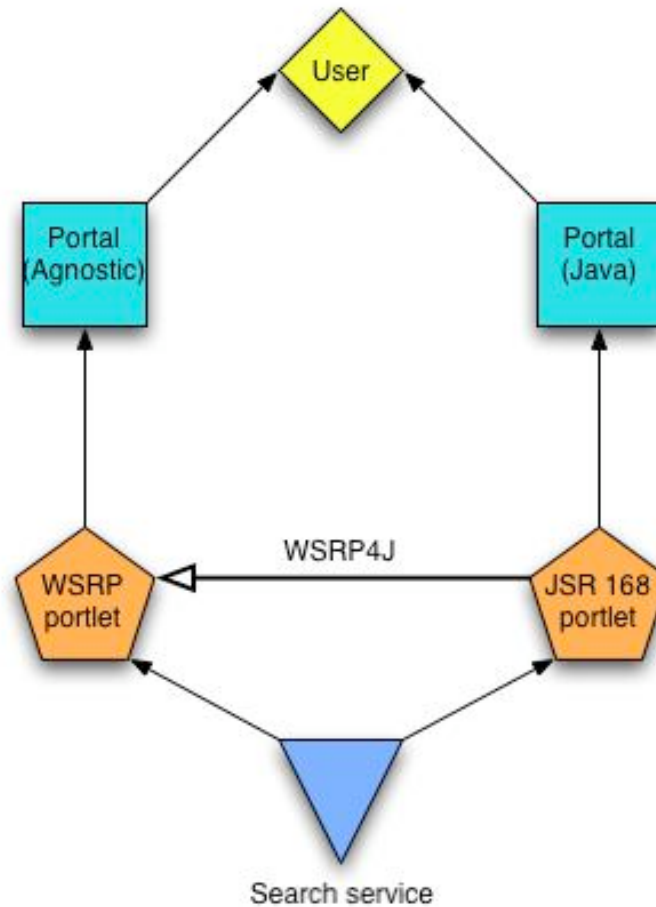


Using standards all round

- JSR 168 portlets can be used alongside WSRP portlets
 - User sees no difference unless WSRP look and feel are different
- JSR 168 portlets can be used as the basis for WSRP portlets
 - Enables both local and remote use



JSR 168 to WSRP



Re-purposing existing tools

- JAFER - Z39.50 engine for searching library catalogues (and other) - Oxford
- GetRef - Metasearch engine for searching bibliographic databases - EDINA
 - Including GetCopy - OpenURL resolver - EDINA
- Archaeology Data Service Z39.50 search
- Google



Outstanding issues

- Inter-portlet communication
 - Will allow information to be moved between portlets and increase functionality
- Security
 - Authentication/authorisation not really covered
 - Communication links not secure



Future steps - WSRP

- WSRP 2.0 due out soon (JSR 168 2.0 in preparation)!
- Allows inter-portlet communication
- Advocates use of WS-Security to secure Web service messages
- Plans for future look at enabling WSRP Consumer within non-portal applications
 - Return to original WSIA intention



Conclusions

- JSR 168 proved as relatively simple method for exposing existing services through portals
- JSR 168 offers path to exposing services as WSRP
- WSRP offers service providers the opportunity to serve their services remotely whilst maintaining support



Conclusions 2

- XSLT proved as a powerful tool for transforming look and feel of portlets
- Search service functionality can be re-produced within a portlet
 - Move from shallow to deep portal
- Enables search to be offered alongside other services within a portal
- Looking to include CREE outcomes in our institutional portal



Questions?

CREE Project (including full results)

<http://www.hull.ac.uk/esig/cree/>



THE UNIVERSITY OF HULL

