Portlets reloaded

Julien Viet
JBoss Inc.

Agenda

- Introduction
- Why Portlet 2.0 ?
- Portlet 2.0 features
- Getting involved

- Julien Viet
  - ✓ JBoss Portal founder and project lead
  - ✓ JBoss Inc. representative on the JSR 286 Portlet 2.0 Expert Group

Portals

- Portals offer an integration layer between several heterogeneous applications
  - ✓ Provide the user the capability to use several applications across organizational boundaries
- Integration occurs at the presentation layer
  - ✓ Single Sign On
  - ✓ Aggregate syndicated content into a single web page
- Other features
  - ✓ Personalization
  - ✓ Internationalization
  - ✓ Entitlement
  - ✓ Collaboration

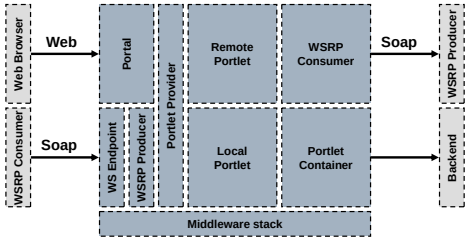
Portlets

- A portlet is a pluggable user interface component model living in the portal ecosystem
  - ✓ Its minimum behavior is to able to produce a markup fragment
- A fragmented market
  - ✓ Historically JEE has seen a variety of different component models
  - ✓ Beyond JEE, other middleware stack have defined their own component models (Zope portlet for instance)
- A portlet container is a service that takes care of managing a portlet
  - ✓ Can run in a portal environment or standalone

WSRP

- Web Services for Remote Portlets is an OASIS standard that defines a contract between the Portal and the Portlet Container
  - ✓ It mandates a set of services that handle the various interactions such as
    - Discovery
    - Runtime execution
    - Management
  - ✓ Based on the SOAP transport layer to guarantee the interoperability between different middleware systems
  - ✓ Introduce the notion of remoteness
  - ✓ Defines the notion of Producer and Consumer, usually
    - the Producer is the Portal
    - the Consumer is the Portlet Container

The full picture



The Portlet 1.0 specification

- The Portlet 1.0 specification defines the component model for the JEE stack
  - ✓ Final in October 2003
  - ✓ Wide market adoption
  - ✓ Integrated with JEE stack but not part of it
  - ✓ Developed along with WSRP 1.0

The Portlet 1.0 specification

- The portlet application defines the packaging unit as a standard war file that contains a portlet.xml in its WEB-INF directory
- Enable the portlet to participate in the global portal request/response cycle
- Provide advanced personalization features
  - ✓ Portlet preferences
  - ✓ Portlet meta data
    - Description
    - Categorization
    - Capabilities
    - etc…

The Portlet 1.0 specification

- Portlet 1.0 defines the bare minimum
  - ✓ Provide a good integration of the portlet within the aggregated portal page
  - ✓ But lacks of coordination between portlets
- Portlets are not Web 2.0 ready
- Difficulties to integrate existing servlet based frameworks, the creation of bridge is not easy
- WSRP 2.0 introduces new features that needs API support

It's time to reload !!!

The Portlet 2.0 specification

- Portlet 2.0 specification
  - ✓ Started 29th of November 2005
  - ✓ IBM leads the specification
  - ✓ More information at http://jcp.org/en/jsr/detail?id=286
  - ✓ Aggressive schedule
    - July 2006 : Early public draft covering partly the spec
    - Oct 2006 : Early public draft
    - Dec 2006 : Public Draft
    - Feb 2007 : Final Public Draft
    - Apr 2007 : Submit Spec+RI+TCK
    - May 2007 : Final draft

Portlet 2.0 key features

- The major complaint is lack of coordination between portlets
  - ✓ Portlet 1.0 only offers the portlet session application to share a state among several portlets of the same application
  - ✓ Portlet 2.0 will add more capabilities
    - Eventing
    - Sharing of render parameters

Sharing state using session

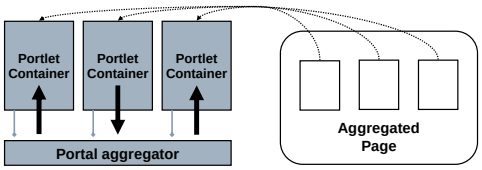
- Shared session
  - ✓ Portlet deployed in the same web application can share objects
  - ✓ Objects are stored in the user session scope

```
public void render(RenderRequest req, RenderResponse resp)
{

  // First we obtain the session
  PortletSession session = req.getPortletSession();

  // This foo attribute is visible in all portlets
  session.setAttribute("foo",
                       "3",
                       PortletSession.APPLICATION_SCOPE);
  …
}
```


Portlet events

- Portlets can use a system of events to coordinate
  - ✓ Really UI event, not business events
  - ✓ Non reliable, i.e no guarantee of delivery
  - ✓ The portlet declares the events it can produce or consume
  - ✓ Portlets are wired at runtime to create a communication bus
  - ✓ API modifications
    - Action phase can generate events
    - A new phase is introduced between the action phase and the render phase that allow the portlet to consume events
    - The event phase can also generate events

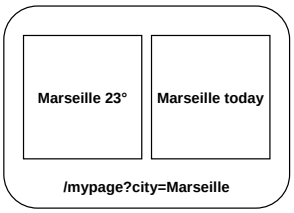
Portlet events



- During an action the porlet container produces an event
- The Portal receives the event and route it to the target portlet containers
- The target containers process the event
  - ✓ Eventually the containers can in response generate other events

Shared render parameters

- Shared render parameters
  - ✓ Allow portlets to share render parameters
    - The scope of sharing is determined by the portal : page, site, …
    - The zip code of a selected city may be a shared render parameter that would allow different portlets to display information for this city



WSRP 2.0 alignement

- WSRP 2.0 alignment
  - ✓ New features
    - Coordination
    - Resource serving
    - Runtime ids like portlet instance id
    - Portlet management : import/export

WSRP resource serving

- Typical use case
  - ✓ download a binary file retrieved from a repository
- The only way to achieve that currently with JSR 168 is to bundle a servlet with your portlet, but there is
  - ✓ No access to preferences
  - ✓ No portal security involved
- Resource serving offers to the portlet the opportunity to server a resource
  - ✓ Full access to the portlet context
  - ✓ Full control over the response

WSRP runtime ids

- Currently it is possible to *somewhat* get the instance id of the portlet
  - ✓ **`response.getNamespace()`** does not have this semantic
  - ✓ Complicated trick
    - Put an object in the session application scope using the key **`XYZ`**
    - It will be stored in the underlying http session with the key **`javax.portlet.<ID>?XYZ`**
    - Request dispatch to a servlet and lookup that object in the session
    - Extract the id from underlying session key
- The new spec now provides the id in the request
  - ✓ Can be used to perform per portlet instance caching

19

Web frameworks

- Better support of web frameworks
  - ✓ Talk to the lead of frameworks like JSF, Struts, Spring, WebWork in order to get feedback how to integrate better portlets with these frameworks
  - ✓ Goal : provide additional means in the specification to make such an integration easy and portable across portals

20

Portlet AJAX Support

- It is possible today to provide limited support for portlets
  - ✓ No access to the context of the portlet
    - Render parameters cannot be updated
    - No access to preferences
    - Cannot generate portlet URLs
  - ✓ Modifying the application session state can potentially affect the state of other portlets on the same page that would not be rendered
- For Portlet 2.0 the new synchronization features make it worse
  - ✓ Eventing
  - ✓ Shared render parameters
- Should work in the WSRP case

21

Portlet 2.0 Agenda

- The first early public draft intends to cover most of the WSRP 2.0 features
  - ✓ July 2006
  - ✓ Create public mailing list in order to get feedback and discuss 1st draft
- Create 2nd early public draft covering all features and feedback of 1st draft
  - ✓ Oct 2006
- Create 1st public draft based on feedback of the 2nd early public draft
  - ✓ Dec 2006

22

Getting involved

- You will have access to the early public draft to give feedback early on
- After the public draft there is a defined review period where you can give feedback
- Once the spec is final you can still send feedback for the next version

23

Q&A

Q&A

24