

# WSRP Cross-Portlet Coordination SC

---

WSRP 8th Face to Face meeting  
March 1<sup>st</sup> – 4<sup>th</sup>, 2004

# Status

---

- Have worked from the original question set to develop a semantic view of the needed coordination model – see <http://www.oasis-open.org/archives/wsrp/200402/bin00001.bin>

# Goals

---

1. Enable portlets to become aware of changes at other portlets regardless of the Producer.
  2. Attempt to piggy-back on existing operations whenever possible:
  3. Leverage existing standards whenever possible.
- Underlying use case: [http://www.oasis-open.org/apps/org/workgroup/wsrp/download.php/1280/wsrp\\_business\\_scenario\\_multimedia\\_sports\\_portal.doc](http://www.oasis-open.org/apps/org/workgroup/wsrp/download.php/1280/wsrp_business_scenario_multimedia_sports_portal.doc)
  - Additional use case: <http://lists.oasis-open.org/archives/wsrp-coord/200402/msg00000.html>
    - Requires an additional goal for the Consumer/End-User being able to affect the initial runtime state of portlets.

# Proposed additional use case

---

- Legacy Systems that:
  - Defines a set of “Page Parameters”
  - Portlet metadata declares what parameters it would like to receive
  - Page designer (or auto-wire) can cause parameters to be passed to the portlet at render time  
=> effectively additional navState
- Note: Such systems could include ability for portlets to update the parameters.
- Note: Requires an additional goal for Consumer/End-User impacting initial portlet runtime state.

# 2 -> 3 step processing model

---

- WSRP v1 processing model
  1. (Optional) User Interaction causes state changes via *performBlockingInteraction()*
  2. Current portlet state rendered via *getMarkup()*
- WSRP v2 processing model
  1. (Optional) User Interaction causes state changes via *performBlockingInteraction()*
  2. (Optional) Consumer mediates distribution of coordination information to the portlets. Current intent is to use events, preferably by leveraging other standards, to carry this information.
  3. Current portlet state rendered via *getMarkup()*

# Needed changes – Metadata

---

- Add metadata (WSDL?) about events a portlet can generate.
  - Allows page designer to decide whether/how to wire each event to other portlets.
- Add metadata (WSDL?) about events a portlet can process.
  - Allows page designer to decide whether/how to wire events from other portlets to this one.

# Needed changes – Signatures

---

- *performBlockingInteraction()* needs to be able to return events generated during its processing.
- Current intent is to always pass arrays of events (reduces overhead). Semantics:
  - Returned array is in the order events were generated.
  - Array passed to Producer/Portlet is in the order the events are to be processed
  - => Provides Consumer with the same control afforded by individual invocations.

# Needed changes – New operation

---

- Operation that can receive the array of events along with sufficient information to invoke the portlet with the right state. Still discussing the **semantics**, currently we have:

[PortletContext, sessionId, events[], newMode?  
newWindowState?, newNavState?]  
= **handleEvents** (Registration Context,  
PortletContext, RuntimeContext, UserContext,  
navigationalState, portletStateChange, mode,  
windowState, events[]);



# State changes

---

- Processing events can impact any portion of a Portlet's state:
  - **portletState**: Cloning may occur => handling the portletStateChange flag is the same as that detailed for *performBlockingInteraction()*.
  - **session state**: Impacts raise a question regarding cache invalidation (later).
  - **navigationalState**: Has same navigational impacts as *performBlockingInteraction()*.

# Questions: Event Generation

---

- Can events occur outside the scope of a user interaction? **(yes)**
  - Is page load a (logical) user interaction?
  - Can the Consumer generate arbitrary events?
- Should the spec declare that Consumer decorations have to generate specific events? **(possibly as a may generate specific events)**
- Any predefined events? **(yes ... level depends value of the definition to the Consumer/Portlet)**
  - modeChanged
  - windowStateChanged
  - ...

# Questions: Event Processing

---

- Are generations of events synchronized?
  - Must all Portlets have responded to the events distributed to them before another generation of event distribution starts? **(no)**
  - If not, can additional events be targeted to a Portlet still processing the previous set of events? **(no)**
- Limit the number of generations to one for v2.0? **(no)**
- Is there a specific processing model for distributing events? For example, DOM has distinct capture and bubbling phases. Transaction systems often have beginning and ending markers.

# Questions: Event metadata

---

- Any value to capturing possibility for event chains? **(not for v2)**
- Value of event "topic" hierarchical structure?
  - Topic trees (e.g. WS-Notification) may be more than we need, but listing all possible events could be costly. **(Try using for the purpose a portlet stating what events it is capable of processing)**
  - Just a "root/parent/child" style? **(no)**

# Questions: Event metadata

---

- Is it an error for a portlet to generate an event it hasn't described? **(no)**

# Questions: Processing notification

---

- Would a light-weight mechanism for indicating success/failure for the processing of a user interaction make sense? (see use cases Andre proposed)

## PortletDescription

boolean notifyFailure

boolean notifySuccess (don't expect this to be used much)

States whether the portlet must be notified about status of last interaction processing step:

- “event processing failed” notification REQUIRED if notifyFailure==true
- “event completed normally” notification REQUIRED if notifySuccess==true

**(Consider, but reserve judgement for full strawman)**

# Questions: Security

---

- Include means by which an event can be readable only by its recipient (e.g. encrypted data)? **(Should not preclude)**
- Include means for policy statement requires all processing of an event's payload respect some policy regarding confidentiality, privacy, etc? **(Include with security considerations)**

# Questions: Consumer support

---

- Should we introduce additional layers of Consumer functionality; noEvents vs simpleEvents vs complexEvents? **(likely a good idea)**
- Consider Consumers that will be tying into existing event propagation infrastructure that may only handle simple types.
- **Consider value of portlet's declaring more about the events they process.**



# Questions: Granularity

---

- Seems a bit strange to do per-portlet event arrays rather than per-producer event/portlet arrays from a protocol point of view (Producer is the web service endpoint). Is this for:
  - Convenience?
  - Addressing the normal case (only one or two events to a portlet at the Producer)?
  - Reflecting real need for directly addressing portlets?
- Note: Makes failure semantics be per-portlet rather than per-producer!
- **Note motivation within the description (i.e. logically targets a portlet much like the other operations within the Markup portType)**

# Questions: Miscellaneous

---

- **Andre:** propose  
PortletDescription.(boolean)alwaysCallHandleEvent  
On any performBlockingAction on any Portlet in  
page/scope call HandleEvent for this portlet, even if the  
action processing (i.e. pba) did not raise any events.  
The event list passed would be empty (or just event  
target), but the EventHandlerContext would indicate  
"user action".  
Gives portlets the chance to update or coordinate with  
other local portlets on ALL user interactions.  
**(possibility as a predefined event)**

# Questions: Miscellaneous

---

- Can performBlockingInteraction return markup and events in a single response?  
**(Leave optimization in place, but make sure the semantics of what is implied are clear)**

# Questions: Miscellaneous

---

- Processing an event can cause local state changes which the Consumer can not detect.
- How can the Portlet cause cached markup to be invalidated?
  - For its current set of MarkupParams?
  - For other portlets with the same groupID?
  - For other portlets at this Producer?