



ASAP over WSRM

Document identifier: wd-asap-asapwsrmguide-01
Location: <http://www.oasis-open.org/committees/asap/docs/>
Author: John Fuller <jfuller@wernervas.com>
Subject: Position paper on the use of ASAP with WSRM
Abstract:

ASAP provides a protocol for managing asynchronous instances of service using SOAP messages and WSRM provides a lower layer protocol for reliable delivery of SOAP messages. An ASAP operation or service can have specified reliability quality of service features, implemented by a reliable message processing layer using WSRM headers. Demonstration WSDL fragments and message element cardinalities are provided for a simple SOAP Request-Response use of ASAP over WSRM.

Status: Working Draft associated with ASAP 0.9 and WSRM 1.1.

ASAP Overview

ASAP provides a protocol for managing asynchronous instances of service using SOAP. To create a service instance, an Observer sends a request to a Factory containing an extensible context data element to provide data for the creation and execution of the service. Requests may be made of instances, and Instances may produce messages for Observers when states change. When service completes, a completion request message for the instance is sent to Observers containing an extensible data element for the result. Interactions between Observer and Factory, Observer and Instance, and Instance and Observer are all request-response operations, and information model data is conveyed in the extensible context and result elements.

WSRM Overview

WSRM provides a protocol supporting at-least-once, at-most-once and exactly-once message delivery for SOAP messages. The protocol specifies the behavior of message producing and message consuming Reliable Message Processors which use data provided in WS-Reliability headers. Reliable Message Processors encapsulate a service layer for providing reliability in the web service protocol stack. For request-response operations, the message producer submits a request to its message processor, the request crosses the wire and is delivered by the message consuming message processor to the message consumer. The message consumer responds to its message processor with the reply, the reply crosses the wire, and the message producer is notified by its message processor.

WSRM supports three reply patterns: Response, which uses a single SOAP request-response, Callback, where the reply is made as a SOAP request, and Poll, where the reply is made as a SOAP request given a Poll SOAP request. Callback and Poll reply patterns allow for high-throughput and fire-and-forget scenarios. While the service of delivery may be thought of as asynchronous in these cases, an application sends a SOAP message to invoke a application layer service operation: as far as the application is concerned that operation isn't the quality of service feature that the SOAP message arrive at the service.

Example Use of WSRM for ASAP

Creation and completion of asynchronous service are good examples where reliable delivery would be desirable and the WSRM protocol could be applied for these messages.

WSDL

ASAP WSDL provides bindings that define operations for Factory, Observer and Instance services. WSRM WSDL provides a way to extend WSDL elements with compositors to specify features and properties of the reliability feature for the element.

The factory and observer WSDL could specify their reliability feature support with a syntax like that given below.

The factory's WSDL may specify

```
<operation name="CreateInstance">
  <fnp:compositor uri="http://docs.oasis-open.org/wsrn/2004/06/fnp-1.1.xsd/compositor/all">
    <fnp:feature uri="http://docs.oasis-open.org/wsrn/2004/06/wsrmp-1.1.xsd">
      <fnp:compositor uri="http://docs.oasis-open.org/wsrn/2004/06/fnp-1.1.xsd/compositor/zero-or-more"
name="DeliverySemantics">
        <fnp:property name="wsrmp:NoDuplicateDelivery"><fnp:value>true</fnp:value></fnp:property>
        <fnp:property name="wsrmp:OrderedDelivery"><fnp:value>true</fnp:value></fnp:property>
        <fnp:property name="wsrmp:GuaranteedDelivery"><fnp:value>true</fnp:value></fnp:property>
      </fnp:compositor>
      <fnp:compositor uri="http://docs.oasis-open.org/wsrn/2004/06/fnp-1.1.xsd/compositor/choice"
name="ReplyPatternChoice">
        <fnp:property name="wsrmp:ReplyPattern"><fnp:value>Response</fnp:value></fnp:property>
        <fnp:property name="wsrmp:ReplyPattern"><fnp:value>Callback</fnp:value></fnp:property>
        <fnp:property name="wsrmp:ReplyPattern"><fnp:value>Poll</fnp:value></fnp:property>
      </fnp:compositor>
    </fnp:feature>
  </fnp:compositor>
  <soap:operation soapAction="http://www.oasis-open.org/asap/0.9/asap/factory/CreateInstance"
style="document" />
  <input>
    <soap:body use="literal" />
    <soap:header message="s1:CreateInstanceRequest" part="Request" use="literal" />
  </input>
  <output>
    <soap:body use="literal" />
    <soap:header message="s1:CreateInstanceResponse" part="Response" use="literal" />
  </output>
</operation>
```

The observer's WSDL may specify

```
<operation name="Completed">
  <fnp:compositor uri="http://docs.oasis-open.org/wsrn/2004/06/fnp-1.1.xsd/compositor/all">
    <fnp:feature uri="http://docs.oasis-open.org/wsrn/2004/06/wsrmp-1.1.xsd">
      <fnp:compositor uri="http://docs.oasis-open.org/wsrn/2004/06/fnp-1.1.xsd/compositor/zero-or-more"
name="DeliverySemantics">
        <fnp:property name="wsrmp:NoDuplicateDelivery"><fnp:value>true</fnp:value></fnp:property>
        <fnp:property name="wsrmp:OrderedDelivery"><fnp:value>true</fnp:value></fnp:property>
        <fnp:property name="wsrmp:GuaranteedDelivery"><fnp:value>true</fnp:value></fnp:property>
      </fnp:compositor>
      <fnp:compositor uri="http://docs.oasis-open.org/wsrn/2004/06/fnp-1.1.xsd/compositor/choice"
name="ReplyPatternChoice">
        <fnp:property name="wsrmp:ReplyPattern"><fnp:value>Response</fnp:value></fnp:property>
        <fnp:property name="wsrmp:ReplyPattern"><fnp:value>Callback</fnp:value></fnp:property>
        <fnp:property name="wsrmp:ReplyPattern"><fnp:value>Poll</fnp:value></fnp:property>
      </fnp:compositor>
    </fnp:feature>
  </fnp:compositor>
  <soap:operation soapAction="http://www.oasis-open.org/asap/0.9/asap/observer/Completed" style="document"
/>
  <input>
    <soap:body use="literal" />
    <soap:header message="s1:CompletedRequest" part="Request" use="literal" />
  </input>
  <output>
    <soap:body use="literal" />
    <soap:header message="s1:CompletedResponse" part="Response" use="literal" />
  </output>
</operation>
```

Message Elements

In WSRM, Reliable message replies maybe included with requests to allow piggybacking.

Elements	Cardinality
soap:Envelope	{1}
soap:Header	{1}
wsm:Request	{0,1}
xsd:any	{0,1,..n}
wsm:MessageId	{1}
wsm:SequenceNum	{0,1}
wsm:ExpiryTime	{1}
wsm:ReplyPattern	{1}
wsm:Value	{1}
wsm:ReplyTo	{0,1}
wsm:AckRequested	{0,1}
wsm:DuplicateElimination	{0,1}
wsm:MessageOrder	{0,1}
wsm:Response	{0,1}
xsd:any	{0,1,..n}
wsm:NonSequenceReply	{0,1,..n}
xsd:any	{0,1,..n}
wsm:SequenceReplies	{0,1,..n}
xsd:any	{0,1,..n}
wsm:ReplyRange	{0,1,..n}
xsd:any	{0,1,..n}
xsd:any	{0,1,..n}
soap:Body	{1}

The simplest implementation maybe the SOAP request-response message exchange pattern and Response reliability reply pattern where the reliable message reply for a message rides on the SOAP response, so really only the ASAP requests need be WSRM requests and ASAP responses WSRM responses.

CreateInstance SOAP Request-Response Example

SOAP Request

Elements	Cardinality
soap:Envelope	{1}
soap:Header	{1}
wsmr:Request	{0,1}
xsd:any	{0,1,..n}
wsmr:MessageId	{1}
wsmr:SequenceNum	{0,1}
wsmr:ExpiryTime	{1}
wsmr:ReplyPattern	{1}
wsmr:Value	{1}
wsmr:ReplyTo	{0,1}
wsmr:AckRequested	{0,1}
wsmr:DuplicateElimination	{0,1}
wsmr:MessageOrder	{0,1}
xsd:any	{0,1,..n}
asap:Request	{1}
asap:SenderKey	{0,1}
asap:ReceiverKey	{1}
asap:ResponseRequired	{0,1}
asap:RequestID	{0,1}
soap:Body	{1}
asap:CreateInstanceRq	{1}
asap:StartImmediately	{1}
asap:ObserverKey	{0,1}
asap:Name	{0,1}
asap:Subject	{0,1}
asap:Description	{0,1}
asap:ContextData	{1}

SOAP Response

Elements	Cardinality
soap:Envelope	{1}
soap:Header	{1}
wsmr:Response	{0,1}
xsd:any	{0,1,..n}
wsmr:NonSequenceReply	{0,1,..n}
xsd:any	{0,1,..n}
wsmr:SequenceReplies	{0,1,..n}
xsd:any	{0,1,..n}
wsmr:ReplyRange	{0,1,..n}
xsd:any	{0,1,..n}
xsd:any	{0,1,..n}
asap:Response	{1}
asap:SenderKey	{1}
asap:ReceiverKey	{0,1}
asap:RequestID	{0,1}
soap:Body	{1}
asap:CreateInstanceRs	{1}
asap:InstanceKey	{1}

Completed SOAP Request-Response Example

SOAP Request

Elements	Cardinality
soap:Envelope	{1}
soap:Header	{1}
wsm:Request	{0,1}
xsd:any	{0,1,..n}
wsm:MessageId	{1}
wsm:SequenceNum	{0,1}
wsm:ExpiryTime	{1}
wsm:ReplyPattern	{1}
wsm:Value	{1}
wsm:ReplyTo	{0,1}
wsm:AckRequested	{0,1}
wsm:DuplicateElimination	{0,1}
wsm:MessageOrder	{0,1}
xsd:any	{0,1,..n}
asap:Request	{1}
asap:SenderKey	{0,1}
asap:ReceiverKey	{1}
asap:ResponseRequired	{0,1}
asap:RequestID	{0,1}
soap:Body	{1}
asap:CompletedRq	{1}
asap:InstanceKey	{1}
asap:ResultData	{1}

SOAP Response

Elements	Cardinality
soap:Envelope	{1}
soap:Header	{1}
wsm:Response	{0,1}
xsd:any	{0,1,..n}
wsm:NonSequenceReply	{0,1,..n}
xsd:any	{0,1,..n}
wsm:SequenceReplies	{0,1,..n}
xsd:any	{0,1,..n}
wsm:ReplyRange	{0,1,..n}
xsd:any	{0,1,..n}
xsd:any	{0,1,..n}
asap:Response	{1}
asap:SenderKey	{1}
asap:ReceiverKey	{0,1}
asap:RequestID	{0,1}
soap:Body	{1}
asap:CompletedRs	{1}

Acknowledgements

Thanks to the ASAP and WSRM technical committees for review, Keith Swenson for ASAP early conceptual clarifications and Doug Bunting for clarifying representation of WSRM element extensibility in the pseudoschema.