

# **What Type of Reliable Messaging will be needed for Grid Computer Networks?**

Alan J Weissberger  
Data Communications Technology  
aweissberger@sbcglobal.net

## **Introduction**

The advantages of Web Services reliable message delivery for eCommerce and eBusiness have been well documented by IBM, Microsoft and others. An excellent whitepaper on this topic can be downloaded from:

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnglobspec/html/ws-rm-exec-summary.asp>

However, there has been almost nothing published on how grid data, control, and management messages will be reliably delivered to one or more end points (over a grid computer network). A gridtoday article by this author noted that WS Reliability was being used in the Japanese government sponsored Business Grid Project, for event notifications. Please refer to:

<http://www.gridtoday.com/04/1115/104252.html>

No other announcements have been forthcoming on use of WS reliable message delivery for Grids. We do know that grid networks are different from conventional Intranets and Extranets, but what's so special about grids? How do they differ from previous distributed systems? Franco Travostino of Nortel- chair of the GGF Grid High Performance Network Research Group (GHPN RG) - provided some answers at last October's OIF-GGF workshop:

-Grid user knows of a resource pool; the pool (but not its constituents) knows about the user. However, there is no a priori knowledge of specific resources by the user.

-Capability to "gang- schedule" a subset of a resource pool (CPU, storage, sensors, etc)

-Grids may straddle across administrative boundaries, trust boundaries, and large distances (Editors Note: early deployment of enterprise grids is usually contained within a single administration and trust cloud. This is because grid security has not been standardized yet and is dependent on WS Security roadmap of IBM-MSFT)

-Dynamic matchmaking in time space between "virtual organizations" and resource pools

<http://news.taborcommunications.com/msgget.jsp?mid=298965&xsl=story.xsl>

## Are Web Services Relevant for Grids?

Some grid developers think that web services are not needed at all and that TCP/IP will be sufficient for reliable message delivery, perhaps augmented by compression, acceleration, or some type of proprietary “throughput boosting” network solution. In sharp contrast, the GGF OGSA WG long ago defined a grid service as a web service (implying SOAP/XML encoded messages) with certain additional properties. The GGF recommends use of IPv6 for its expanded address space, but has not yet taken a position on the Transport protocol. While most assume it will be TCP, there are issues and concerns about that protocol in a grid networking environment, due to low throughput when different packet sizes are mixed over a single network interface. Indeed, the GGF High Performance Network Research Group is exploring alternative Transport layers. The GGF OGSA WG should be defining all Web Services specifications that are part of the grid infrastructure, but progress on this front has been very slow. Consider that all Web Services messages are SOAP/XML encoded, which significantly expands the original (un-encoded) message size and further degrades end to end throughput. Also note that certain application data (e.g. real time video/graphics, floating point numbers) can not be well represented as SOAP/XML streams. So the jury is still deliberating how and where web services will be used for grids.

Hence, we are left with many unanswered questions as to what role web services and reliable messaging play in a grid environment:

-Should grid application data messages use SOAP/XML encoding and be represented as a web service? If so, should some or all of those messages use WS reliable messaging?

-What WS format should application data messages take, e.g. as user data in the SOAP message body or as a minimal SOAP with (un encoded) attachments message?

-What about Grid control, management and security messages? [Here, most GGF participants assume the use of Web Services and emerging web services standards, e.g. **WS Security, WS Distributed Management, WS Resource Framework, and WS Notification**. All of these WS protocols will need to be enveloped in some type of reliable messaging SOAP header and “composed” with one of two competing WS Reliable Messaging specifications- see below].

-Consider that there are two very mature reliable message delivery specifications for web services (**WS Reliability** and **WS Reliable Messaging**) and two OASIS TCs working to progress the standardization of each one of these separately (**WSRM TC** has completed standardization of WS Reliability, while the new **WSRX TC** will standardize WS Reliable Messaging and WS Policy Assertions). So which version/ set of specs should be supported when there is a need for WS reliable message delivery in a grid environment? Can both of these specifications co-exist or will WS-Reliability be usurped and obsoleted by the WSRX specs?

## **What Type of WS Reliable Messaging (assuming Web Services provide the basic grid infrastructure)?**

Let's look a bit deeper at this last conundrum. First, let's consider the **WS Reliable Messaging** spec. First published in March 2003 (coincidentally this is when the WSRM TC was created by a different set of vendors), the WS Reliable Messaging spec is co-authored by IBM, Microsoft, BEA, and TIBCO. It now has very wide industry support as evidenced by the three interop events, where implementations of this spec were tested. The interop events also involved "composability" testing of WS Reliable Messaging with WS-Addressing (now submitted to W3C), WS-Security (an OASIS standard) and most recently, WS-Secure Conversation and WS-Trust (these latter two specs have not yet been submitted to a standards body). The message exchange patterns to be tested were described in a Secure WS-Reliable Messaging Scenarios document prepared for the Workshop participants.

To read about the third of these interop events (April 2005), which tested implementations of WS Reliable Messaging composed with WS Secure Conversation, please refer to:

<http://news.taborcommunications.com/msgget.jsp?mid=363854&xsl=story.xsl>

Note that **WS Reliable Messaging** depends on **WS Policy**, which has not yet been submitted to a standards body and has not yet been tested for interoperability.

In February 2005, the WS Reliable Messaging spec was revised. The spec, along with an abstract, XML Schema and associated WSDL, may be downloaded free from:

<http://www-128.ibm.com/developerworks/webservices/library/specification/ws-rm/>

or from a vendor neutral site (no abstract):

<http://schemas.xmlsoap.org/ws/2005/02/rm>

On May 3, 2005, a call for participation for the new WSRX TC was announced by OASIS. That document along with the TC Charter, may be referenced from:

<http://xml.coverpages.org/WS-RX-CFP.html>

A generic description of the WSRX TC along with background information may be referenced from:

<http://xml.coverpages.org/ni2005-05-04-a.html>

Forty-one (41) individuals identified as "WSRX TC co-proposers" have already agreed to support the work of the new TC, representing at least twenty-eight (28) corporate institutions. These include: ACORD, Actional, Adobe, Arjuna, BEA Systems, Blue Titan, Choreology, Entrust, Ericsson, +Hitachi, IBM, IONA, Microsoft, +NEC, Nortel,

Novell, OAGi, +Oracle, Reactivity, SAP, +SeeBeyond, +Sonic Software, +Sun Microsystems, Systinet, TIBCO, United Kingdom e-Government Unit, The University of North Carolina at Chapel Hill, and webMethods.

+ These companies were also proponents of the WSRM TC and the WS-Reliability standard.

From the “coverpages” link above:

The WS-RX TC will continue development of the (BEA Systems, IBM, Microsoft, and TIBCO Software) **Web Services Reliable Messaging** specification (WS-Reliable Messaging) submitted to the TC. The defined mechanism by which Web services express support for reliable messaging and related useful parameters "will be based upon the **Web Services Reliable Messaging Policy Assertion (WS-RM Policy Assertion\*)** specification," also to be submitted to the TC. The WS-RX work will be designed to compose with the WSS (Security) TC specifications and will utilize the WS-Addressing (W3C version) functions where appropriate, and avoiding the creation of overlapping functions.

\* assumed to be a subset of WS-Policy spec:

<http://schemas.xmlsoap.org/ws/2005/02/rm/policy>

Many approaches have been taken to reliable message transfer. The WS-RX TC Call for Participation includes a discussion of related work, which recognizes the OASIS Web Services Reliable Messaging TC and the OASIS ebXML Business Process TC (which developed the ebMS spec that uses WS Reliability and WS Security). The **WSRM TC, although it has a goal similar to that of WS-RX, "has a fundamentally different view with regard to scope of the specification, required functions, policy, and Web services architecture composability."**

*Some members of the WSRM TC might disagree with the quote above.* They would suggest that the only key difference here is that policy is embedded in WS-Reliability protocol (or the associated WSDL file), and does not require a separate WS Policy specification. In particular, WS-Reliability defines an abstract contract that can be bound to different representations. The QoS required for reliable message delivery (exactly once, duplicate elimination, or sequential/ordered delivery) may be specified via the WS-Reliability protocol, when sent to the recipient (or consumer) of the WS reliable message. This would allow the policy or agreement to be deployed on the client side only, while the Web Service endpoint would advertise reliable message QoS capabilities (e.g. through the associated WSDL file). The QoS can also be specified on a per message basis.

One key difference in spec functionality is that **WS Reliability includes a Polling capability**, while WS Reliable Messaging does not. Polling enables reliable messaging protocol to reach WS endpoints behind a firewall (that would otherwise block Request/Response message exchanges).

[Note: Chris Ferris, a very knowledgeable and experienced WS architect from IBM, takes the view that while polling is necessary, it is a generic function that is orthogonal to reliable message delivery. *We wonder if any user behind a firewall would be willing to send and receive unreliable messages.*]

With respect to composability, the views seem to be consistent and not at all different. NEC and Fujitsu had previously proposed an interop test of **WS Reliability composed with WS Security**. This has the same basic goal as the interop testing of WS Reliable Messaging with WS Secure Conversation (that spec is not part of the WSRX TC charter). Also, Oracle will provide examples of SOAP header dumps with both WS-Reliability and WS-Security headers in use, as per their implementation for a previous interop event.

### **WSRM TC Liaison with WSRX TC**

Several member companies of the WSRM TC (Oracle, Fujitsu, Hitachi, and NEC) have recently formed a task force to work on a "gap analysis" contribution to the WSRX TC. The gap analysis will be based on a comparison and mapping of the two WSRX TC specs with the previously generated WSRM TC reliable messaging functional requirements document. This work is expected to be completed prior to the first WSRX TC meeting June 23-24 in Palo Alto, CA.

### **Quick Take on WS Reliability**

WS Reliability defines three ways for the receiver to send back an Acknowledgment message or a Fault message to the sender. These are referred to as the "RM Reply patterns," which are defined as follows:

- **Response RM-Reply Pattern**

We say that a Response RM-Reply pattern is in use if the outbound Reliable Message is sent in the underlying protocol request, and the resultant Acknowledgment message (or Fault message) is contained in the underlying protocol response message which corresponds to the original request. In essence, the Acknowledgement is "piggybacked" onto the business response message.

- **Callback RM-Reply Pattern**

We say that a Callback RM-Reply pattern is in use if the Acknowledgment message (or Fault message) is contained in an underlying protocol request of a second request/response exchange (or a second one-way message), operating in the opposite direction to the message containing the outbound Reliable Message.

- **Polling RM-Reply Pattern** (not included in WS Reliable Messaging)

We say that the Polling RM-Reply pattern is being used if a second underlying protocol request is generated, in the same direction as the one

containing the outbound Reliable Message, to act as a "request for acknowledgment." The Acknowledgment message (or Fault message) is contained in the underlying protocol response to this request. This polling pattern can be used in instances where it is inappropriate for the sender of reliable messages to receive underlying protocol requests e.g. the sender behind a firewall.

These three reply patterns provide "the users" with flexibility to send reliable request/ response or one-way SOAP messages (Callback and Polling patterns). Callback is important for one-way request message patterns and for batching of acknowledgements and fault messages. We have already mentioned the importance of Polling to reach a WS endpoint behind a firewall. Otherwise, the two reliable message delivery specs have very similar functionality (WS Reliable Messaging has a few additional features).

### **What about implementations of these specs?**

WS Reliability has an open source implementation, developed by the three Japanese companies (Fujitsu, Hitachi, NEC) involved in the Business Grid Project. Referred to as "Reliable Messaging for Grid Services or RM4GS," it was supported by the Japanese Ministry of Economy, Trade, and Industry (MITI). Please refer to:

<http://www.adtmag.com/article.asp?id=10291>

[There is no current plan in the Japanese business grid project to replace WS-Reliability with WS-Reliable Messaging or any other WSRX TC deliverable spec]

In addition to the RM4GS, several vendors have developed their own implementations of WS-Reliability. Two of these are believed to be Fujitsu Software USA and Oracle Corp.

WS Reliable Messaging also has an open source version. It is being developed by an Apache group based in Sri Lanka. To be sure, WS Reliable Messaging has been implemented by many more vendors than WS-Reliability, as evidenced by double digit participation in the interop events. There are also Internet end points available from IBM, Microsoft, BEA, Systinet and others for more informal interop testing with emerging implementations.

**But which one of the two specifications will be more relevant for grids? Even more important, what role will WS reliable message delivery play in the grid infrastructure/ ecosystem? We have no answers here and look to GGF for guidance.**

**Acknowledgements:** The author thanks Jacques Durand of Fujitsu and Chris Ferris of IBM for their very valuable contributions and clarifications made in the review and editing of this article.