Application Notes for WS-Reliability 1.1 Version 1.0

Committee Draft 01

30 March 2007

Specification URIs:
This Version:
>http://docs.oasis-open.org/wsrm/ws-reliability/v1.1/cd01/application-notes-cd-01.pdf
>http://docs.oasis-open.org/wsrm/ws-reliability/v1.1/cd01/application-notes-cd-01.doc
>http://docs.oasis-open.org/wsrm/ws-reliability/v1.1/cd01/application-notes-cd-01.html

Latest Approved Version:
>http://docs.oasis-open.org/wsrm/ws-reliability/v1.1/application-notes.pdf
>http://docs.oasis-open.org/wsrm/ws-reliability/v1.1/application-notes.doc
>http://docs.oasis-open.org/wsrm/ws-reliability/v1.1/application-notes.html

Technical Committee:
>OASIS WS Reliable Messaging TC

Chair(s):
>Tom Rutt

Editor(s):
>Jacques Durand

Related work:
>This specification is related to:
>
>- WS-Reliability 1.1

Declared XML Namespace(s):
>None

Abstract:
>This document describes how applications might use the WS-Reliability specification.

Status:
>This document was last revised or approved by the WS Reliable Message Technical Committee on the above date. The level of approval is also listed above. Check the "Latest Version" or "Latest Approved Version" location noted above for possible later revisions of this document.
>
>Technical Committee members should send comments on this specification to the Technical Committee's email list. Others should send comments to the Technical Committee by using the "Send A Comment" button on the Technical Committee's web page at http://www.oasis-open.org/committees/wsrm/.
>
>For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the Technical Committee web page (http://www.oasis-open.org/committees/wsrm/ipr.php.

The non-normative errata page for this specification is located at http://www.oasis-open.org/committees/wsrm/.

Notices

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS's procedures with respect to rights in OASIS specifications can be found at the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementors or users of this specification, can be obtained from the OASIS Executive Director.

OASIS invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to implement this specification. Please address the information to the OASIS Executive Director.

Table of Contents

1 Introduction

This document presents non-normative information that may be of benefit to WS-Reliability [WS-Reliability] application developers. It provides a guide for WS-Reliability application developers who already understand the fundamentals of developing a WS-Reliability application.

The purpose of this document is to answer common questions that might arise during WS- Reliability application development by means of non-normative scenarios and examples. Additionally, this document may also serve as an aid to clarify potential usage scenarios of WS-Reliability. The intended audience of this document are WS-Reliability application designers

Each section of this document is addressing a different aspect of WS-Reliability application development. Each section contains information on best practices, along with examples, or reference to examples elsewhere.

This document does not present a full end-to-end WS-Reliability example and is not meant as an entry point to WS-Reliability.

For normative descriptions of WS-Reliability, readers should refer to the WS-Reliability specification [WS-Reliability].

2 Application Notes relating to Reliable Sending of Response Messages

2.1 Problem Statement

Business payloads are often carried over protocol responses. This is the case in request-response message exchange patterns or when message pulling is being used (e.g. by e-Business SME partners in order to overcome their connectivity constraints). The WS-Reliability 1.1 specification does not distinguish between reliability of a request and reliability of a response. It assumes that a message that is not acknowledged - whether request or response - may always be resent by an implementation. In practice, the resending of such response messages is posing challenges that require a proper resolution – and preferably the same resolution across implementations in order to interoperate.

This guideline document makes a recommendation on how to ensure At-Least-Once reliability (GuaranteedDelivery agreement) to the response leg of a two-way protocol (see definition in next section) that is underlying to SOAP. It also describes what features that are stated as optional in the specification, must be supported by the implementation.

2.2 Terminology

Two-way protocol: An underlying transport protocol is qualified here as "two-way" if: (a) it guarantees a channel for transferring the response of every message (request) initiated by an MSH, back to this MSH without need for explicit addressing information in SOAP headers and regardless of connectivity restrictions such as inability to accept incoming new connections, and (b) it provides some means to the MSH initiator of the exchange for correlating the response with the request, without relying on the SOAP header. For example, HTTP qualifies as two-way, but SMTP and FTP do not (although FTP has a notion of session, it does not inherently support the coupling of (b)).

Back-channel: The channel offered by the request message in a two-way protocol is also called back-channel in this specification.

Response message: a response message is implicitly understood as a message sent over the back-channel offered by another message (the request message).

NOTE: the reliability model in WS-Reliability is described for messages submitted to an implementation via "Submit" operation, and delivered via "Deliver" operation. A response message is supposed to be submitted via the "Respond" operation. It is assumed in this document that the reliability semantics extends to messages submitted via "Respond".

2.3 Implementation Guidelines

In 3.2.2 (Duplicate Elimination section) of [WS-Reliability], the following requirement is stated:

When the Response RM-Reply Pattern is requested with Duplicate Elimination for a Reliable
Message, the Receiving RMP cannot deliver that message to the Consumer again (because it is
a duplicate of a previously delivered message), and a Consumer response payload is expected,

58 *the response of the SOAP MEP instance MUST contain one (but not both) of the following:*
59 *(a)- a copy of the original response payload returned for that Message (in the SOAP Body)*
60 *in addition to the Acknowledgment Indication (in the SOAP Header) or*
61 *(b)- a SOAP server Fault.*

62 The option (a) requires the implementation to have the ability to cache a response message that had
63 already been sent, and to be able to resend it later in case a duplicate of the request is received.

64   The solution recommended here for ensuring the reliability of a response message, requires that
65 an implementation supports option (a). The initial response produced for the first request message
66 received and delivered, must be cached until expiration or until it is acknowledged, whichever comes first.
67

2.4 Usage Recommendation

69

70 <u>How to set the reliability for the request message:</u>

71

72 Intuitively, the reliability of a response message in a request-response MEP depends on the reliability of
73 the request message.

74 As a prerequisite to supporting GuaranteedDelivery for a response message, a user must ensure that the
75 following reliability agreements are in use for the request message:

76 - GuaranteedDelivery. The reply pattern must be "Response".
77 - DuplicateElimination. In addition, the implementation must exhibit the behaviour (a) stated in the
78   previous section (3.3), when a request message duplicate is received.

79 <u>Considerations about notification of delivery failure</u>:

80 When sending a request message, the RMP is not necessarily aware of the MEP that this request
81 belongs to. It usually has no way to know whether a response message was expected or if this response
82 was sent reliably. The Producer application of the request is assumed to have access to this knowledge.

83 Consequently, when a request message does not get acknowledged, a delivery failure notified to its
84 Producer could mean either (1) the request failed to be delivered to the other party, or (2) the request was
85 delivered, but the response – along with the request acknowledgement – failed to be delivered, in case a
86 response was expected. In both cases, before concluding that a delivery failure has occurred, the request
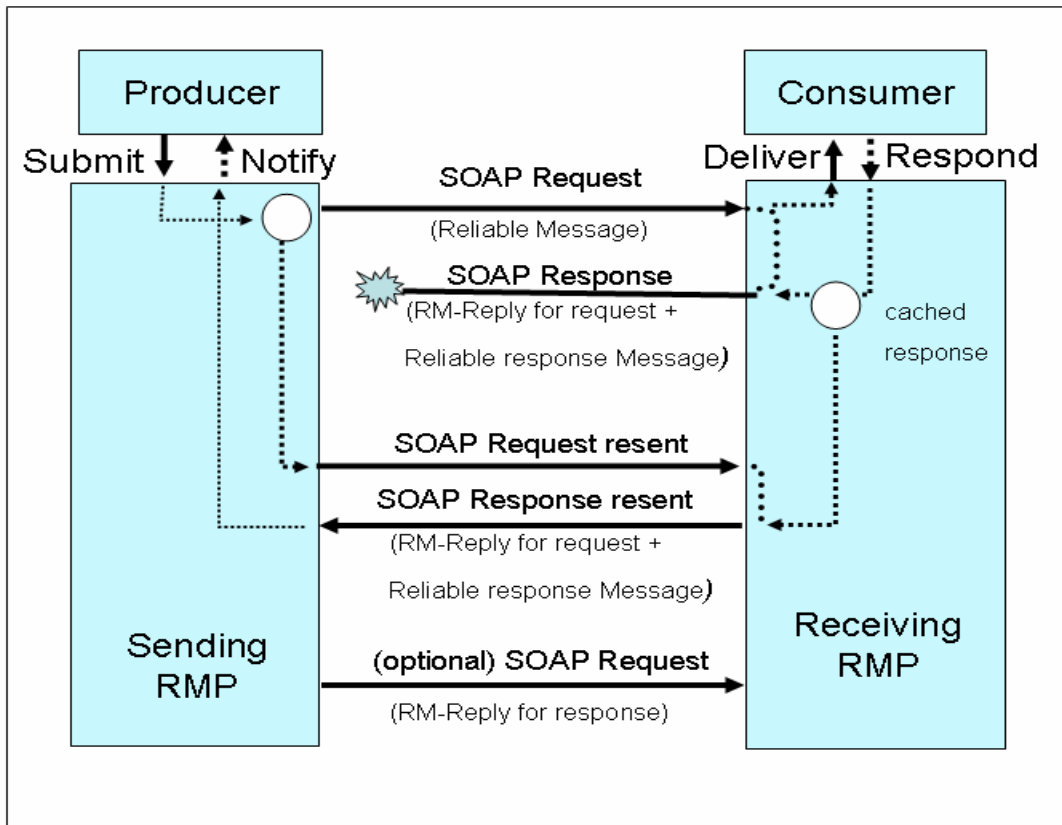87 will be resent, providing an opportunity to resend the cached response, had it failed before.

88 Given the above, request acknowledgement (or failure to get one) is sufficient to ensure the following: (1)
89 resending of the response message as a consequence of resending the request message, (2) delivery
90 failure notification for both request and response, to the request Producer party. However, without
91 acknowledgment of the response, the response producer must be willing to cache each response until its
92 request message expires since it is would not received delivery failure notifications for its responses. In
93 some cases, a producer may choose to request an acknowledgment of receipt of its response
94 messages.. This response acknowledgment will serve the following objectives:

95 (a) More efficient cache management, since the response may be removed from the cache when its
96 acknowledgment is received.

97 (b) Notification the request Consumer party of failure to deliver its response message (even though the
98 request Producer was also notified).

99 In case an implementation adds the wsrm:Request element with an AckRequested element to the
100 response message, the RM-Reply Pattern value must be "Callback".

101   The following figure illustrates a reliable exchange request-response, with both messages being
102 acknowledged. Both request message and response message are cached for resending purpose.
103 Duplicate elimination for the request prevents the request message from being delivered twice in case the
104 request is resent.

105
106
107
108
109
110

3 References

3.1 Normative References

[WS-Reliability]   Web Services Reliability 1.1,

http://docs.oasis-open.org/wsrm/ws-reliability/v1.1

OASIS Standard, November 2004..

3.2 Non-Normative References

118 A. Revision History

119

120

Revision	Date	Editor	Changes Made
0.1	01/22/2007	Jacques Durand	Initial Creation based on previous drafts of reliable sending of synchronous response messages
0.4	02/06/2007	Jacques Durand	Editorial changes due to feedback from Tom Rutt, Paul Knight
1.0	03/25/2007	Jacques Durand	Formatting using OASIS templates.

121

122