WS-I RSP WG Usage Scenarios

Document Status: Working Group Draft

Version: 1.0

Date: November 06, 2006

This version:
http://www.ws-i.org/Profiles/RSP-Scenarios-1.0-2006-11-06.pdf

Latest version:
http://www.ws-i.org/Profiles/RSP-Scenarios-1.0.pdf

Errata for this version:
http://www.ws-i.org/Profiles/RSP-Scenarios-1.0-errata.pdf

Editors:

Jacques Durand, Fujitsu (jdurand@us.fujitsu.com)
Marc Goodner, Microsoft (mgoodner@microsoft.com)

Notices

MATERIAL, WHETHER OR NOT SUCH PARTY HAD ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

Feedback

The Web Services-Interoperability Organization (WS-I) would like to receive input, suggestions and other feedback ("Feedback") on this work from a wide variety of industry participants to improve its quality over time.

By sending email, or otherwise communicating with WS-I, you (on behalf of yourself if you are an individual, and your company if you are providing Feedback on behalf of the company) will be deemed to have granted to WS-I, the members of WS-I, and other parties that have access to your Feedback, a non-exclusive, non-transferable, worldwide, perpetual, irrevocable, royalty-free license to use, disclose, copy, license, modify, sublicense or otherwise distribute and exploit in any manner whatsoever the Feedback you provide regarding the work. You acknowledge that you have no expectation of confidentiality with respect to any Feedback you provide. You represent and warrant that you have rights to provide this Feedback, and if you are providing Feedback on behalf of a company, you represent and warrant that you have the rights to provide Feedback on behalf of your company. You also acknowledge that WS-I is not required to review, discuss, use, consider or in any way incorporate your Feedback into future versions of its work. If WS-I does incorporate some or all of your Feedback in a future version of the work, it may, but is not obligated to include your name (or, if you are identified as acting on behalf of your company, the name of your company) on a list of contributors to the work. If the foregoing is not acceptable to you and any company on whose behalf you are acting, please do not provide any Feedback.

Feedback on this document should be directed to:
wsi_rsp_comment@lists.ws-i.org.

Executive Overview

The RSP WG has decided to approach defining requirements for the RSP profile in terms of realistic and detailed use cases, called usage scenarios.

This document describes these usage scenarios. These scenarios will serve as detailed input for the profiling work, providing evidence of potential interoperability issues and/or need for best practice guidelines.

Table of Contents

Introduction..... 5

Status of this Document..... 5

Role of this Document..... 5

Properties of Usage Scenarios..... 5

Artifacts and Specifications Coverage..... 5

Definitions..... 7

Conventions in Defining Scenarios..... 9

Reliable One-way (ROW)..... 11

Description..... 11

Sequence Diagram..... 11

Scenario Constraints and Assumptions..... 12

Message Exchanges Details..... 13

Reliable One-way, anonymous client (ROW-anon)..... 15

Description..... 15

Sequence Diagram..... 15

Scenario Constraints and Assumptions..... 16

Message Exchanges Details..... 17

Reliable Request-Response (RRR)..... 19

Description..... 19

Sequence Diagram..... 19

Scenario Constraints and Assumptions..... 20

Message Exchanges Details..... 20

Reliable Request-Response, anonymous client (RRR-anon)..... 24

Description..... 24
Sequence Diagram..... 24
Scenario Constraints and Assumptions..... 25
Message Exchanges Details..... 25
Secure Conversation Establishment and Cancellation.....29
RequestSecurityToken, CreateSequence (RST-CS)..... 29
TerminateSequence, Cancel (TS-Cancel).....29
Revision History..... 31

Introduction

Status of this Document

This document is an Editors Draft; it has not yet been accepted by the Working Group as reflecting the current state of discussions. It is a work in progress, and should not be considered authoritative or final. Other documents may supersede this document.

This document will be updated from time to time to incorporate new usage scenarios as they are identified.

Role of this Document

The usage scenarios in this document do not represent exhaustive ways to combine the specifications targeted for the RSP profile, but only those ways that seem to exhibit interoperability issues or that need guidance.

The usage scenarios in this document represent input material candidate for profiling, and should not be interpreted as best practices for integrating the specifications targeted for the RSP profile. The RSP profile may actually restrict them, or propose better alternatives.

Other patterns of usage that do not fit in these scenarios are legitimate as long as the final RSP does not preclude them. Conversely, some of these scenarios or their options may later be precluded by RSP.

Properties of Usage Scenarios

A Usage Scenario is illustrative of real usage conditions, and of the rationale behind them. It describes assumed or possible environmental constraints, e.g. addressing, security, and reliability.

A Usage scenario details all contextual exchanges needed to enable it end-to-end (establishment of security context, or reliability sequences) and related options.

Artifacts and Specifications Coverage

The usage scenarios in this document involve the following Web services artifacts and specifications, subject to profiling, either individually or in composition:

Specifications:

- WS-I Basic Profile 1.2
- WS-I Basic Profile 2.0
- WS-I Basic Security Profile 1.0
- WS-I Basic Security Profile 1.1
- WS-ReliableMessaging 1.1
- WS-SecureConversation

Definitions

The following terms will be used throughout this document to refer to the various factors that make up individual scenarios.

Addressable client: A client that is capable of accepting connections on a network endpoint.

Anonymous client: A client that does not accept incoming connections.

Asynchronous request-response message exchange: A SOAP message exchange in which a requester sends a SOAP message to a service and receives a response message. "Asynchronous" in this context refers to the manner in which the underlying transport protocol is used to carry the request and response messages. The response message is sent over a separate connection that is initiated by the service to the client (a "callback").

Message Exchange Unit: A unit representing a coherent atomic exchange of elements (and related messages).

One-way message: An application SOAP message for which no application SOAP response is expected.

Reliable messaging: The act of sending SOAP messages using the WS-ReliableMessaging 1.1 protocol.

Reliable message: A message sent reliably using the WS-ReliableMessaging 1.1 protocol.

Request message: An application SOAP message for which an application SOAP response is expected.

Response message: An application SOAP message triggered by a request message.

Secure messaging: In the general sense this term refers to the act of sending a message with one or more of the following security qualities: integrity, confidentiality, and authenticity. For the purposes of this document it is assumed that these attributes will be provided through the use of either SSL/TLS or WS-SecureConversation 1.3.

Sequence Lifecycle Message: A message that contains one of: CreateSequence, CreateSequenceResponse, CloseSequence, CloseSequenceResponse, TerminateSequence, TerminateSequenceResponse as the child element of the SOAP body element.

Sequence Traffic Message: A message containing a Sequence header block.

Synchronous request-response message exchange: A SOAP message exchange in which a requester sends a SOAP message to a service and receives a response message. "Synchronous" in this context refers to the way in which the underlying transport protocol used to carry the request and response messages. The response message is returned on the back channel of the request message.

Conventions in Defining Scenarios

A scenario may be viewed under different perspectives, which will be captured and represented differently in this document.

These main perspective lines are:

- Overall description and usage rationale.
- Sequence diagram describing the messages choreographies. These will show flow diagrams, where solid lines represent **requests** over an underlying protocol, and dashed lines represent **responses** sent back over the back-channel offered by the request.
- Constraints and assumptions underlying to the entire scenario (e.g. addressing constraints of one of the endpoints)

In addition, the message choreography as reported in the activity diagram can be decomposed as a sequence of *message exchange units*, a unit representing a coherent atomic exchange of elements (and related messages) such as CreateSequence/ CreateSequenceResponse, or AckRequested /SequenceAcknowledgement, or yet an exchange of a SecurityContextToken element.

The scenario definition introduces a description of how each one of these units of message exchanges, is carried out. This is done in form of a table that shows various dimensions or aspects of the execution of such a unit. The general layout for each instance of such a table is as follows:

Scenario Message Exchange Unit(s)	Aspects of the Message Exchange Unit	Message Details
(example: RM protocol CreateSequence/ CreateSequenceResponse)	Addressing and correlation	The following are examples of addressing information whose values may be called out or be specified for specific legs of an exchange. wsa:ReplyTo wsa:RelatesTo wsa:To wsa:Action
	Underlying protocol binding and connection establishment	Underlying MEP being used and how (HTTP) Any reliance on connection establishment (e.g. MakeConnection)
	Piggybacking	(patterns allowed by the scenario)
	Security	(may be relevant or not depending on the scenario)
	Error handling	(content details and addressing aspects)

Reliable One-way (ROW)

Description

Scenario summary: Reliable One-way Exchange, where the client endpoint is addressable. The initiator (requestor) is called the Client, the other endpoint the Service.

Use Case: The most common use case is of a client that initiates a request to a service for which no response is expected. The message is sent reliably. The client is addressable, and both parties decide to NOT make use of the underlying protocol back channel for any response to the client. Secure conversation may be used.

Sequence Diagram

The complete scenario includes the following exchanges. The following diagram does not illustrate any optional underlying protocol back-channel use:

- [optional] Secure Conversation Establishment and Cancellation
- Reliable Sequence establishment (CS/CSR)
- Application reliable exchange (1 instance of One-way message)
- Acknowledgment exchanges (either after this message, or later a consolidated Ack)
- [optional] Sequence Closing
- Sequence Termination

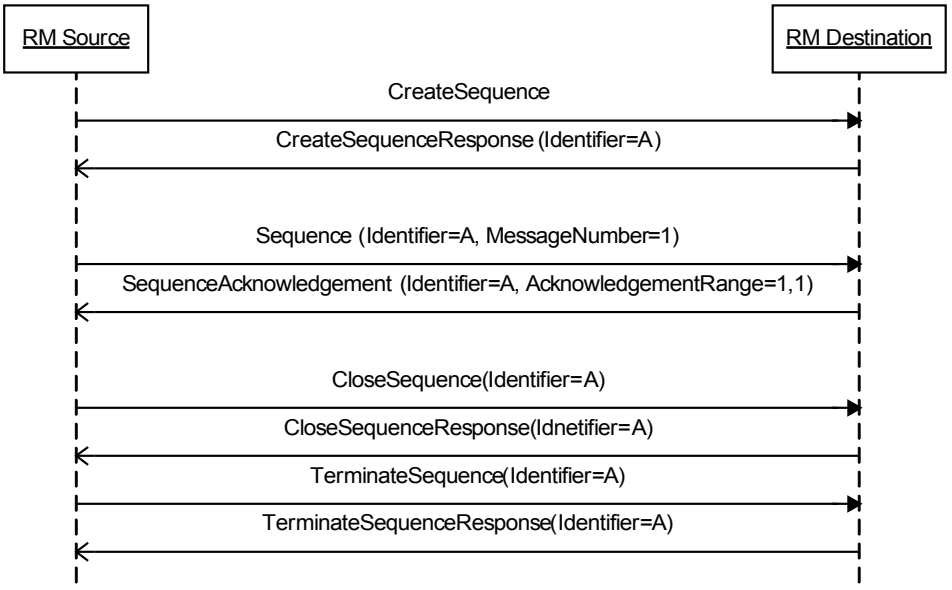


Figure 1 – Reliable One-way

Scenario Constraints and Assumptions

No addressing constraints for either client or service endpoints.

Assumptions:

- In this usage scenario the client assumes the service endpoint has a preference for issuing any responses as new requests over the underlying protocol.

Scenario Constraints:

- There are no specific constraints in this scenario. Both endpoints are addressable.

Description:

- If WSDL is used then there must be no out messages defined.

Message Exchanges Details

Sequence Lifecycle Messages

Scenario Message Exchange Unit(s)	**Aspects of the Message Exchange Unit**	**Message Details**
Sequence establishment (CS/CSR) Sequence closing (optional) (ClS/ClSR) Sequence termination (TS/TSR)	Addressing and correlation	Wsa:ReplyTo : (on CS / ClS / TS) **client endpoint reference** Wsa:RelatesTo: (expected on CSR / ClSR / TSR, relates to request) Wsa:To
	Underlying protocol binding and connection establishment	Two (HTTP) requests in opposite directions. Endpoints involved in exchange must be prepared for new HTTP connection
	Piggybacking	Not applicable. Additional SOAP headers may be present.
	Security	Message level security: Optional following guidelines from WS-RM sections 5 and 6.
	Error handling	WS-Addressing rules apply in handling faults.

Sequence Traffic Messages

Note that there are no differences in Sequence Traffic messages for an addressable and anonymous client.

Scenario Message Exchange Unit	**Aspect of the Message Exchange Unit**	**Message Details**
Application Message exchange : A **One-way message** (as defined in terminology)	Addressing and correlation	wsa:To
	Underlying protocol binding and connection establishment	Underlying request (HTTP) No application message on HTTP response, though possibly SOAP envelope with a Fault.
	Piggybacking	Not Applicable.
	Security	Message level security: Optional, RM headers must follow guidelines from WS-RM sections 5 and 6 if the sequence is protected.
	Error handling	WS-Addressing rules apply in handling faults.

Acknowledgment Messages

Scenario Message Exchange Unit	**Aspect of the Message Exchange Unit**	**Message Details**
Acknowledgements driven by either (a) spontaneous new requests as determined by Ack policy, or (b) in response to AckRequested messages	Addressing and correlation	wsrm:AcksTo EPR: client endpoint reference
	Underlying protocol binding and connection establishment	For AckRequested: Underlying protocol request (HTTP) or AcksTo EPR. For Acks: Sent to AcksTo EPR per WS-RM processing rules
	Piggybacking	For AckRequested: can be piggybacked on application one-ways, or sent separately. For Acks: possibly over SOAP requests containing application messages sent to client endpoint.
	Security	If the sequence is protected then acknowledgements must be secured per the rules in WS-RM sections 5 and 6.
	Error handling	WS-Addressing rules apply in handling faults.

Reliable One-way, anonymous client (ROW-anon)

Description

Scenario summary: Reliable One-way Exchange, with the use of an anonymous client endpoint. The initiator (requestor) is called the Client and is anonymous, the other endpoint the Service.

Use Case: The most common use case is of a client that initiates a request to a service for which no response is expected. The message is sent reliably. The client is addressable, and both parties decide to make use of the underlying protocol back-channel for all responses to client. Secure conversation may be used.

Sequence Diagram

The complete scenario includes the following exchanges. Every response uses the underlying protocol back channel:

- [optional] Secure Conversation Establishment and Cancellation
- Reliable Sequence establishment (CS/CSR)
- Application reliable exchange (1 instance of One-way message)
- Acknowledgement exchanges (either after this message, or later a consolidated Ack)
- [optional] Sequence Closing
- Sequence Termination

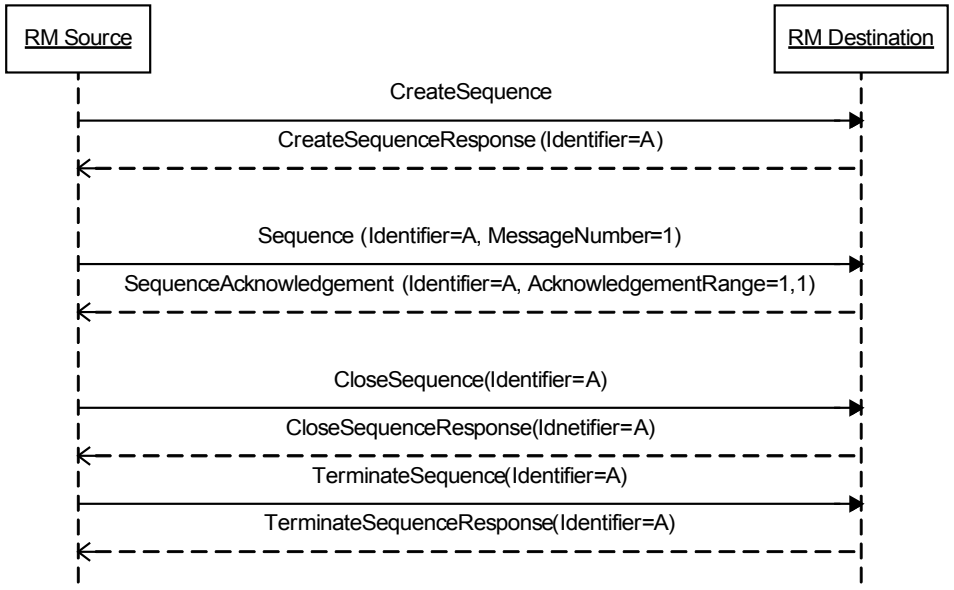


Figure 2 - Reliable One-way, anonymous client

Scenario Constraints and Assumptions

No addressing constraints for either client or service endpoints.

Assumptions:

- In this usage scenario, client assumes the service endpoint has a preference for not issuing requests back to it and will use the back channel for all its responses.

Scenario Constraints:

- There are no specific constraints in this scenario.

Description:

- If WSDL is used then there must be no out messages defined.

Message Exchanges Details

Sequence Lifecycle Messages

Scenario Message Exchange Unit(s)	**Aspects of the Message Exchange Unit**	**Message Details**
Sequence establishment (CS/CSR) Sequence closing (optional) (ClS/ClSR) Sequence termination (TS/TSR)	Addressing and correlation	[optional] Wsa:ReplyTo : (on CS / ClS / TS) **anonymous** wsa:RelatesTo: (expected on CSR / ClSR / TSR, relates to request) wsa:To wsa:Action
	Underlying protocol binding and connection establishment	Single (HTTP) request-reply MEP
	Piggybacking	Not applicable. Additional SOAP headers may be present.
	Security	Message level security: Optional following guidelines from WS-RM sections 5 and 6.
	Error handling	WS-Addressing rules apply in handling faults.

Sequence Traffic Messages

Note that there are no differences in Sequence Traffic messages for an addressable and anonymous client.

Scenario Message Exchange Unit	**Aspect of the Message Exchange Unit**	**Message Details**
Application Message exchange : A **One-way message** (as defined in terminology)	Addressing and correlation	wsa:To
	Underlying protocol binding and connection establishment	Underlying request (HTTP) No application message on HTTP response, though possibly SOAP envelope with RM headers, or a Fault.
	Piggybacking	Not Applicable.
	Security	Message level security: Optional, RM headers must follow guidelines from WS-RM sections 5 and 6 if the sequence is protected.
	Error handling	WS-Addressing rules apply in handling faults.

Acknowledgement Messages

Scenario Message Exchange Unit	**Aspect of the Message Exchange Unit**	**Message Details**
Acknowledgements driven by either (a) piggybacking over responses (as determined by Ack policy not represented here), or (b) AckRequested messages, or (c) MakeConnection messages.	Addressing and correlation	wsrm:AcksTo EPR: **anonymous**
	Underlying protocol binding and connection establishment	For AckRequested: Underlying request (HTTP) For Acks: back-channel of underlying protocol (response to application message, or response to MakeConnection.)
	Piggybacking	For AckRequested: can be piggybacked on application one-ways, or sent separately. For Acks: only SOAP responses of one-ways (empty SOAP body).
	Security	If the sequence is protected then acknowledgements must be secured per the rules in WS-RM sections 5 and 6.
	Error handling	WS-Addressing rules apply in handling faults.

Reliable Request-Response (RRR)

Description

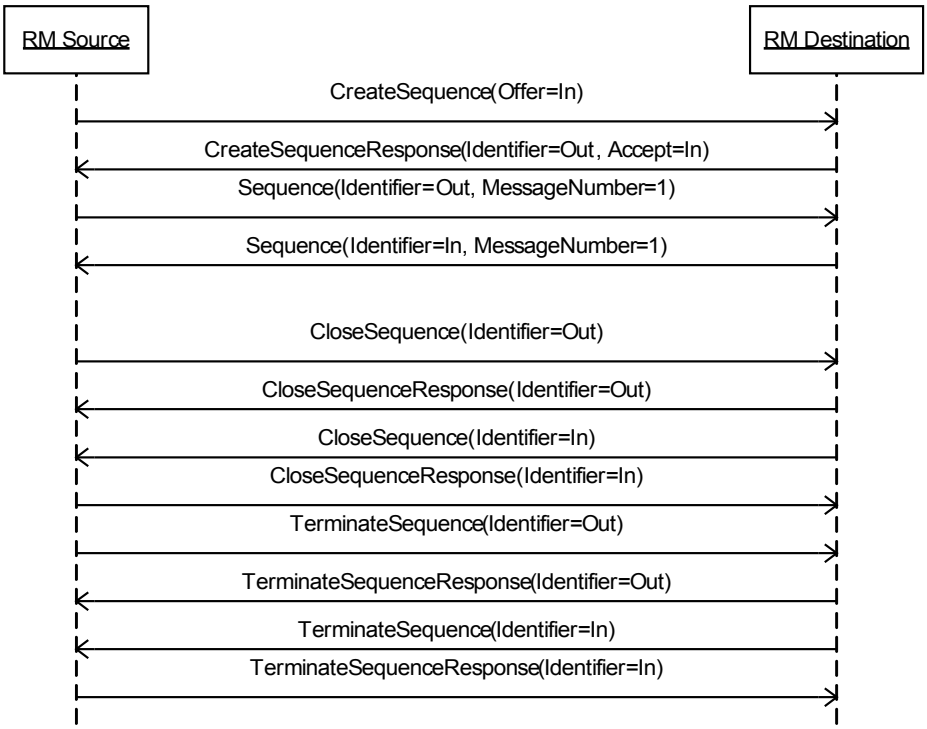
Scenario summary: Reliable asynchronous Two-way Exchange, NO use of anonymous endpoint: both endpoints are addressable. The initiator (requestor) is called the Client, the other endpoint the Service.

Use Case: A common use case is of a client that initiates a request to a service, for which a response is expected on a separate connection. The request message is sent reliably. The service responds with a separate service invocation reliably carrying the response to the client. Both endpoints are addressable, and both decide to NOT make use of the underlying protocol back-channel for any response. Secure conversation may be used.

Sequence Diagram

The complete scenario includes the following exchanges. None of them uses the underlying protocol back-channel:

- [optional] Secure Conversation Establishment and Cancellation
- Reliable Sequence establishment client-to-service (CS/CSR), with offered service–to-client sequence.
- Application reliable request client-to-service
- Application reliable response service–to-client
- Acknowledgement exchange client-to-service. (not shown)
- Acknowledgement exchange service–to-client. (not shown)
- [optional] Sequence Closing client-to-service.
- [optional] Sequence Closing service–to-client.
- Sequence Termination client-to-service.
- Sequence Termination service–to-client.



Scenario Constraints and Assumptions

No addressing constraints for either client or service endpoints.

Assumptions:

- In this usage scenario, both client and service assume the other endpoint has a preference for issuing any responses to their request messages, as new requests over the underlying protocol.

Scenario Constraints:

- No specific constraints in this scenario. Both endpoints are addressable.

Description:

- When WSDL is used then there will be either request-response operations or independent in and out messages defined. If WSDL is used then there must be no out messages defined.

Message Exchanges Details

Sequence Lifecycle Messages

Scenario Message Exchange Unit(s)	**Aspects of the Message Exchange Unit**	**Message Details**
Client-service Sequence establishment (CS/CSR) Client-service Sequence closing (optional) (ClS/ClSR) Client-service Sequence termination (TS/TSR) Service-client Sequence closing (optional) (ClS/ClSR) Service-client Sequence termination (TS/TSR)	Addressing and correlation	Wsa:ReplyTo : (on CS / ClS / TS) **client endpoint reference** Wsrm:Offer (on CS) Wsrm:Accept (on CSR) Wsa:RelatesTo: (expected on CSR / ClSR / TSR, relates to request) Wsa:To
	Underlying protocol binding and connection establishment	Two (HTTP) requests in opposite directions. Endpoints involved in exchange must be prepared for new HTTP connection
	Piggybacking	Not applicable.
	Security	Message level security: Optional following guidelines from WS-RM sections 5 and 6.
	Error handling	WS-Addressing rules apply in handling faults.

Sequence Traffic Messages

(Only varies from table in scenario 6 by ReplyTo value.)

Scenario Message Exchange Unit	**Aspect of the Message Exchange Unit**	**Message Details**
Application Message request : A **One-way, request, or response message**	Addressing and correlation	wsa:ReplyTo : client endpoint reference wsa:RelatesTo: For a response message, URI / message ID of the request.
	Underlying protocol binding and connection establishment	Underlying request (HTTP) No application message on HTTP response, though possibly SOAP envelope with a Fault.
	piggybacking	Not applicable.
	Security	Message level security: Optional, RM headers must follow guidelines from WS-RM sections 5 and 6 if the sequence is protected.
	Error handling	WS-Addressing rules apply in handling faults.

Acknowledgment Messages

Scenario Message Exchange Unit	Aspect of the Message Exchange Unit	Message Details
Acknowledgements from Service, driven by (a) spontaneous new requests as determined by Ack policy, or (b) in response to AckRequested messages Acknowledgements from Client, driven by either (a) spontaneous new requests as determined by Ack policy, or (b) in response to AckRequested messages	Addressing and correlation	AcksTo (for sequence sent to Service): client endpoint reference, or other (NOT anonymous) AckRequested (for sequence sent to Service): sent with wsa:ReplyTo aligned with AcksTo element. AcksTo (for sequence sent to Client): service endpoint reference, or other (NOT anonymous) AckRequested (for sequence sent to Client): sent with wsa:ReplyTo aligned with AcksTo element.
	Underlying protocol binding and connection establishment	For AckRequested: Underlying request (HTTP) For Acks: new request of underlying protocol
	Piggybacking	For AckRequested: can be piggybacked on application one-ways, or sent separately. For Acks: possibly over SOAP requests containing application messages sent to client endpoint.
	Security	If the sequence is protected then acknowledgements must be secured per the rules in WS-RM sections 5 and 6.
	Error handling	WS-Addressing rules apply in handling faults.

Reliable Request-Response, anonymous client (RRR-anon)

Description

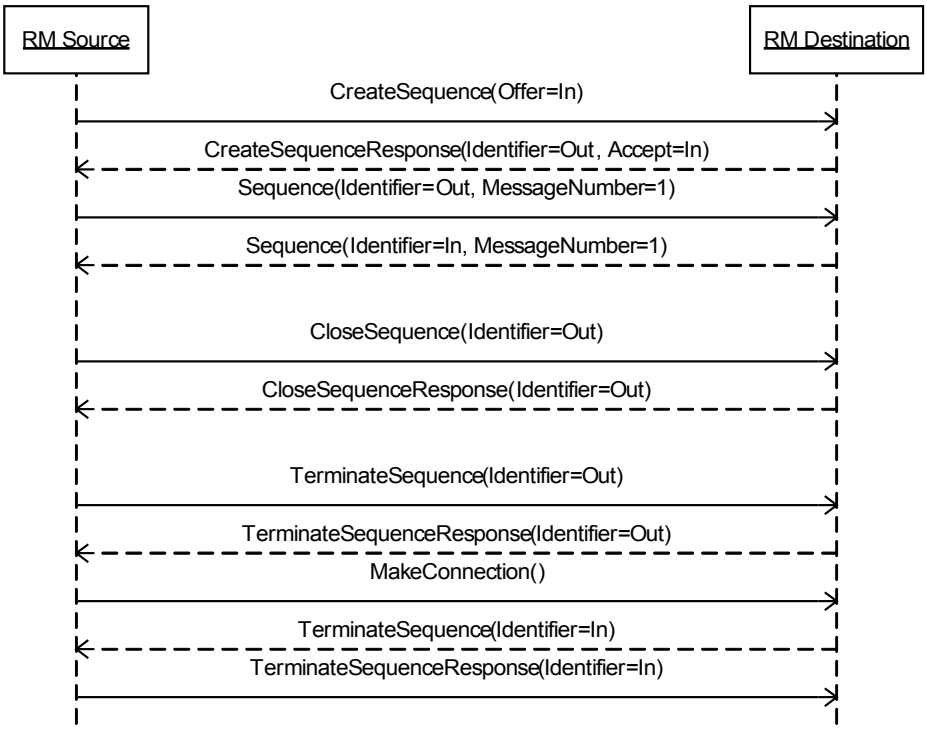
Scenario summary: Reliable asynchronous Two-way Exchange, with one anonymous endpoint (or behaving as such). The initiator (requestor) is called the Client, the other endpoint the Service.

Use Case: A common use case is of a client that initiates a request to a service, for which a response is expected on the same connection. The request message is sent reliably. The Service responds reliably on the back channel which carries the response to the client. Both endpoints may be addressable, but the Client for some reason has connectivity issues (e.g. firewall) and cannot receive incoming requests, therefore behaves as an anonymous endpoint. Any message from Service to Client will need to make use of the underlying protocol back channel created by a previous request. Secure conversation may be used.

Sequence Diagram

The complete scenario includes the following exchanges. All communication must be initiated by the Client. All of the messages sent from the Client to the service are over new connections. All of the messages sent from the Service to Client use the underlying protocol back-channel of a previous request.

- [optional] Secure Conversation Establishment and Cancellation
- Reliable Sequence establishment client-to-service (CS/CSR), with offered service–to-client sequence (accepted if reliable responses).
- Application reliable request client-to-service (1 instance of One-way message)
- Application reliable response service–to-client (as response in 1 instance of Synchronous request-response exchange, or as response to MakeConnection)
- Acknowledgement exchange client-to-service.
- Acknowledgement exchange service–to-client (using back-channel).
- [optional] Sequence Closing client-to-service.
- [optional] Sequence Closing service–to-client (using back-channel).
- Sequence Termination client-to-service.
- Sequence Termination service–to-client (using back-channel).



Scenario Constraints and Assumptions

No addressing constraints for either client or service endpoints.

Assumptions:

- In this usage scenario, the client only is behaving as non-addressable. All transfers from Service to Client use the back-channel of underlying protocol.

Scenario Constraints:

- Both endpoints may be addressable, but the Client may have connectivity issues that make it behave as non-addressable.

Description:

- When WSDL is used then there will be either request-response operations or independent in and out messages defined. If WSDL is used then there must be no out messages defined.

Message Exchanges Details

Sequence Lifecycle Messages

The difference from the RRR usage scenario is that the Client's ReplyTo is anonymous.

Scenario Message Exchange Unit(s)	**Aspects of the Message Exchange Unit**	**Message Details**
Client-service Sequence establishment (CS/CSR) Client-service Sequence closing (optional) (ClS/ClSR) Client-service Sequence termination (TS/TSR) Service-client Sequence closing (optional) (ClS/ClSR) Service-client Sequence termination (TS/TSR)	Addressing and correlation	wsa:ReplyTo (from Client): (on CS / ClS / TS) **anonymous** wsrm:Offer (on CS from Client) wsrm:Accept (on CSR to Client) wsa:RelatesTo: (expected on CSR / ClSR / TSR, relates to their request messages) wsa:To
	Underlying protocol binding and connection establishment	For Client-service exchanges: a single (HTTP) request-response. For Service-client exchanges: the ClS / TS message is over an HTTP response, back-channel offered by MakeConnection. The ClSR / TSR message is over an HTTP request.
	Piggybacking	Not applicable.
	Security	Message level security: Optional following guidelines from WS-RM sections 5 and 6.
	Error handling	WS-Addressing rules apply in handling faults.

Sequence Traffic Messages

Scenario Message Exchange Unit	**Aspect of the Message Exchange Unit**	**Message Details**
Application Message request : A **One-way message** or a response of a **Synchronous request-response exchange** from Client (unrelated to the initial request), or as response to **MakeConnection**	Addressing and correlation	wsa:ReplyTo (in Client request) : anonymous wsa:RelatesTo: For a response message, URI / message ID of the request.
	Underlying protocol binding and connection establishment	Underlying request (HTTP) No application message on HTTP response to the Request, though possibly SOAP envelope with a Fault. Service to client messages over an HTTP response, back-channel offered by MakeConnection (or in case of variant, reuse of back-channel of any other subsequent request)
	Piggybacking	Possible piggybacking of RM headers or other headers on this message.
	Security	Message level security: Optional, RM headers must follow guidelines from WS-RM sections 5 and 6 if the sequence is protected.
	Error handling	WS-Addressing rules apply in handling faults.

Acknowledgment Messages

The difference from the RRR Usage scenario is that the Client's AcksTo EPR is anonymous.

Scenario Message Exchange Unit	**Aspect of the Message Exchange Unit**	**Manifestation / Control**
Acknowledgements from Service, driven by (a) piggybacking over responses (as determined by Ack policy not represented here), or (b) in response to AckRequested messages, or (c) in response to MakeConnection message. Acknowledgements from Client, driven by either (a) spontaneous new requests as determined by Ack policy, or (b) in new request as response to AckRequested messages	Addressing and correlation	AcksTo (for sequence sent to Service): anonymous AcksTo (for sequence sent to Client): service endpoint reference, or other (NOT anonymous)
	Underlying protocol binding and connection establishment	For AckRequested (from Client): Underlying request (HTTP) For Acks (from Service): response of underlying protocol (HTTP) For AckRequested (from Service): Underlying response (HTTP). For Acks (from Client): new request of underlying protocol (HTTP)
	Piggybacking	For AckRequested or Acks from Client: can be piggybacked on application one-ways. For AckRequested or Acks from Service: can be piggybacked on application responses.
	Security	If the sequence is protected then acknowledgements must be secured per the rules in WS-RM sections 5 and 6.
	Error handling	WS-Addressing rules apply in handling faults.

Secure Conversation Establishment and Cancellation

Every scenario in this document may include additional exchanges for establishing and canceling a secure conversation. The establishment and cancellation of secure conversations will be done according to one of the following sub-scenarios.

RequestSecurityToken, CreateSequence (RST-CS)

A reliable sequence is assumed to be transferred from start to end within a single secure conversation. The conversation is started with the intent of securing this sequence. The conversation may include more than one sequence.

This sub-scenario assumes that the STS / RM Destination is addressable.

Client sends RST (RequestSecurityToken) to the Service endpoint's STS to establish SecurityContextToken. Service endpoint responds with RSTR and new SecurityContextToken.

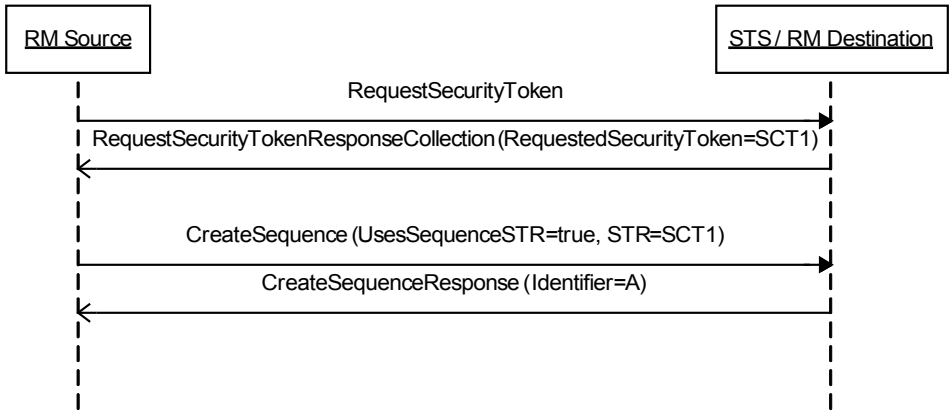


Figure 3 - SCT Establishment

TerminateSequence, Cancel (TS-Cancel)

In this sub-scenario, the secure conversation was established for an RM sequence. This sub-scenario assumes that the STS / RM Destination is addressable.

The secure conversation that includes a reliable sequence, will be cancelled after the sequence is terminated. Client sends RST (RequestSecurityToken) with a CancelTarget element identifying the SecurityContextToken of the conversation to be terminated. Service endpoint responds with RSTRC confirming the cancellation.

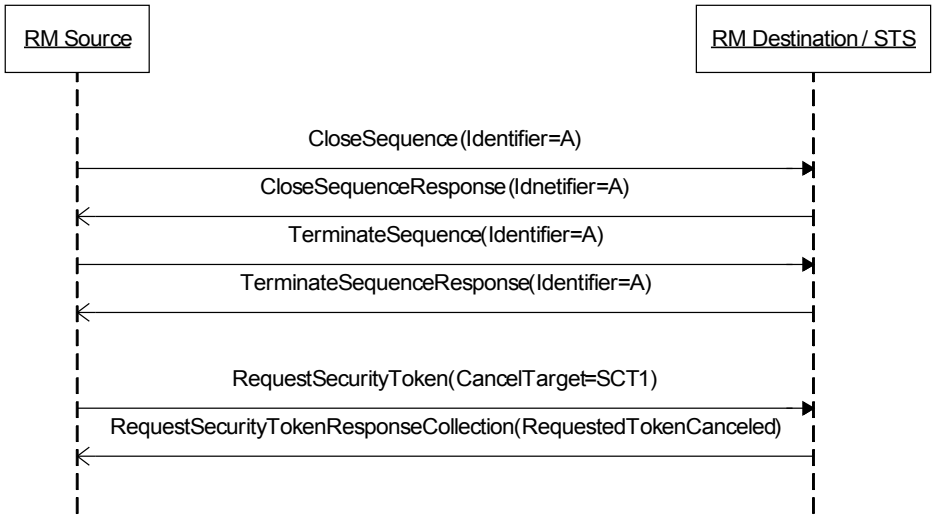


Figure 4 - SCT Cancellation

Revision History

Rev	Date	By Whom	What
0.1	2006-09-1	Jacques Durand	Initial draft.
0.2	2006-09-12	Marc Goodner and Jacques Durand	Review / various edits,
0.3	2006-09-28	Jacques Durand	Updated scenario 1 (-> ROW-anon), added ROW-addressed, RA2W-addressed, RS2W-all.
0.4	2006-09-30	Marc Goodner & Jacques Durand	Added flow diagrams, for SecureConversation exchanges and for ROW scenario.
0.5	2006-10-13	Jacques Durand	Various edits, Added RA2W-1anon scenario after discussion with Marc.
0.6	2006-10-30	Marc Goodner	Edits from Plenary discussion.
1.0	2006-11-06	Bob Freund	WDG 1.0