



Using the AMQP Anonymous Terminus for Message Routing Version 1.0

Committee Specification 01

17 September 2018

Specification URIs

This version:

<http://docs.oasis-open.org/amqp/anonterm/v1.0/cs01/anonterm-v1.0-cs01.xml> (Authoritative)
<http://docs.oasis-open.org/amqp/anonterm/v1.0/cs01/anonterm-v1.0-cs01.html>
<http://docs.oasis-open.org/amqp/anonterm/v1.0/cs01/anonterm-v1.0-cs01.pdf>

Previous version:

N/A

Latest version:

<http://docs.oasis-open.org/amqp/anonterm/v1.0/anonterm-v1.0.xml> (Authoritative)
<http://docs.oasis-open.org/amqp/anonterm/v1.0/anonterm-v1.0.html>
<http://docs.oasis-open.org/amqp/anonterm/v1.0/anonterm-v1.0.pdf>

Technical Committee:

OASIS Advanced Message Queuing Protocol (AMQP) TC

Chairs:

Robert Godfrey (rgodfrey@redhat.com), Red Hat
Clemens Vasters (clemensv@microsoft.com), Microsoft

Editor:

Robert Godfrey (rgodfrey@redhat.com), Red Hat

Related work:

This specification is related to:

- OASIS Advanced Message Queuing Protocol (AMQP) Version 1.0 Part 0: Overview. Edited by Robert Godfrey, David Ingham, and Rafael Schloming. 29 October 2012. OASIS Standard.
<http://docs.oasis-open.org/amqp/core/v1.0/os/amqp-core-overview-v1.0-os.html>.

Abstract:

The Advanced Message Queuing Protocol (AMQP) is an open internet protocol for business messaging. AMQP defines links as a unidirectional transport for messages between a source and a target. The target of a link identifies the node to which messages are to be sent to. If a large number of distinct destinations are in use, or if the destinations to be sent to are not known ahead of time (for example, they are provided as a reply-to in incoming messages) then creating a link per destination can be burdensome. This document defines a mechanism whereby a single outgoing link can be used to transfer messages which are then routed using the address carried in their "to" field.

Status:

This document was last revised or approved by the OASIS Advanced Message Queuing Protocol (AMQP) TC on the above date. The level of approval is also listed above. Check the "Latest version" location noted above for possible later revisions of this document. Any other numbered Versions and other technical work produced by the Technical Committee (TC) are listed at https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=amqp#technical.

TC members should send comments on this specification to the TC's email list. Others should send comments to the TC's public comment list, after subscribing to it by following the instructions at the "Send A Comment" button on the TC's web page at <https://www.oasis-open.org/committees/amqp/>.

This specification is provided under the RF on RAND Terms Mode of the OASIS IPR Policy, the mode chosen when the Technical Committee was established. 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 TC's web page (<https://www.oasis-open.org/committees/amqp/ipr.php>).

Note that any machine-readable content (Computer Language Definitions) declared Normative for this Work Product is provided in separate plain text files. In the event of a discrepancy between any such plain text file and display content in the Work Product's prose narrative document(s), the content in the separate plain text file prevails.

Citation format:

When referencing this specification the following citation format should be used:

[anonterm-v1.0]

Using the AMQP Anonymous Terminus for Message Routing Version 1.0. Edited by Robert Godfrey. 17 September 2018. OASIS Committee Specification 01.

<http://docs.oasis-open.org/amqp/anonterm/v1.0/cs01/anonterm-v1.0-cs01.pdf>.

Latest version: <http://docs.oasis-open.org/amqp/anonterm/v1.0/anonterm-v1.0.pdf>.

Notices:

Copyright © OASIS Open 2018. All Rights Reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Committee Specification or OASIS Standard, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification.

OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this specification by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification. OASIS may include such claims on its website, but disclaims any obligation to do so.

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' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on 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 implementers or users of this OASIS Committee Specification or OASIS Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

The name "OASIS" is a trademark of OASIS, the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see <https://www.oasis-open.org/policies-guidelines/trademark> for above guidance.

Contents

1 Introduction	4
1.1 IPR Policy	4
1.2 Terminology	4
1.3 Normative References	4
1.4 Non Normative References	4
2 Message Routing	5
2.1 Sending A Message	5
2.2 Routing Nodes	5
2.2.1 Link Properties And Target Capabilities	5
2.2.2 Routing Errors	6
2.3 Anonymous Terminus	6
3 Conformance	7
Appendix A. Acknowledgements	8
Appendix B. Revision History	9

1 Introduction

1.1 IPR Policy

This specification is provided under the RF on RAND Terms Mode of the OASIS IPR Policy, the mode chosen when the Technical Committee was established. 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 TC's web page (<https://www.oasis-open.org/committees/amqp/ipr.php>).

1.2 Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this specification are to be interpreted as described in [RFC2119] and [RFC8174] when, and only when, they appear in all capitals, as shown here.

The authoritative form of this specification consists of a set of XML source documents. These documents are transformed into PDF and HTML representations for readability. The machine readable version of the AMQP DTD describes the XML used for the authoritative source documents. This DTD includes the definition of the syntax used any excerpts of XML presented in the PDF and HTML representations.

1.3 Normative References

[AMQP]

Godfrey, Robert; Ingham, David; Schloming, Rafael, *Advanced Message Queuing Protocol (AMQP) Version 1.0*. October 2012, OASIS Standard.

<<http://docs.oasis-open.org/amqp/core/v1.0/os/amqp-core-overview-v1.0-os.html>>

[RFC2119]

Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP14, RFC2119, DOI 10.17487/RFC2119, March 1997,

<<http://www.rfc-editor.org/info/rfc2119>>

[RFC8174]

Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC2119 Key Words", BCP14, RFC8174, DOI 10.17487/RFC8174, May 2017,

<<http://www.rfc-editor.org/info/rfc8174>>

1.4 Non Normative References

[AMQPCONNCAP]

AMQP Capabilities Registry: Connection Capabilities

<http://www.amqp.org/specification/1.0/connection-capabilities>

[AMQPCONNPROP]

AMQP Capabilities Registry: Connection Properties

<http://www.amqp.org/specification/1.0/connection-properties>

2 Message Routing

2.1 Sending A Message

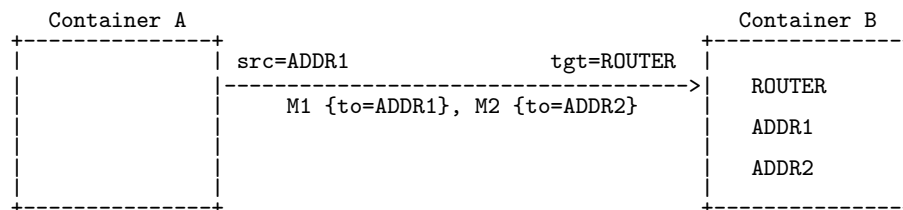
When creating and sending a message AMQP defines two fields which are to be used to determine how the message should be routed:

- the `to` field of `properties` as part of the message
- the `address` field of `target` as part of the link attach

The `to` field specifies the intended ultimate recipient of the message. The `address` field determines the node to which the messages will be transferred to on a given link. The behavior of the node identified by this address determines how the message will be processed. A class of nodes which stores messages for onward distribution between consumers is identified in 3.3 Distribution Nodes of [AMQP], here we define a second class of nodes which do not store messages, but simply route incoming messages to other nodes based on the `to` field of the message.

2.2 Routing Nodes

A *routing node* acts as a proxy for links to other nodes. Messages sent over links into a routing node will be forwarded to the node referenced in the `to` field of `properties` of the message just as if a direct link has been established to that node.



Messages M1 and M2 travel along a single link to the ROUTER node. Message M1 is delivered to address ADDR1, Message M2 is delivered to address ADDR2.

Figure 2.1: Router Example

2.2.1 Link Properties And Target Capabilities

When creating a sending link, the `properties` field of `attach` and the `capabilities` field of `target` can be used to influence the behavior of the receiving node, or other aspects of the interaction between the source and target. The receiving peer will then respond by setting the `properties` field of `attach` and the `capabilities` field of `target` to reflect values that the target node supports.

Since a *routing node* acts only as a proxy between the source and intended target nodes, it itself does not directly support the `properties` and `capabilities` but rather the intended target nodes do. Unless every target node that the *routing node* can route to supports the same set of `properties` and `capabilities`, it is impossible for it to accurately reply with the correct set of supported behaviors.

If a *routing node* receives an *attach* which carries one or more *properties* or for which the *capabilities* field of *target* has one or more entries then the corresponding *attach* SHOULD contain the those *capabilities* and *properties* which are supported by at least some subset of the nodes the *routing node* can route to. If a message is subsequently sent along the link with an address in the *to* field of *properties* which resolves to a node that does not support these behaviors then the *routing node* MUST signal an error, either by rejecting the message (if the source supports the *rejected* outcome and the incoming message was *unsettled*) or by detaching the link with a *not-implemented* error.

2.2.2 Routing Errors

It is possible that a message sent to a *routing node* has an address in the *to* field of *properties* which, if used in the *address* field of *target* of an *attach*, would result in an unsuccessful link establishment (for example, if the address cannot be resolved to a node). In this case the *routing node* MUST communicate the error back to the sender of the message.

If the *source* of the link supports the *rejected* outcome, and the message has not already been settled by the sender, then the *routing node* MUST reject the message. In this case the *error* field of *rejected* MUST contain the error which would have been communicated in the *detach* which would have be sent if a link to the same address had been attempted.

If the *source* of the link does not support the *rejected* outcome, or the message has already been settled by the sender, then an error MUST be reported. If the message was sent as part of a transaction then the rules defined in 4.4.4 Interaction Of Settlement With Transactions of [AMQP] for Transactional Posting apply - that is the error can be reported by immediately destroying the controlling link on which the transaction was declared, or by rejecting any attempt to discharge the transaction where the *fail* flag is not set to true. If the message was not transactionally posted, then the *routing node* MUST detach the link over which the message was sent with an error. The *error* sent by the *routing node* MUST contain the error which would have been communicated in the *detach* sent on attempting to link directly to the address in the message's *to* field. Additionally the *info* field of *error* MUST contain an entry with symbolic key *delivery-tag* and binary value of the *delivery-tag* of the message which caused the failure.

2.3 Anonymous Terminus

A *target* with a *null* address is referred to as the *anonymous terminus*. The *anonymous terminus* is reserved for use as a routing node by containers that support such capabilities. Support for the *anonymous terminus* by a container is determined by the use of a connection capability.

Name	Description
ANONYMOUS-RELAY	<p>If present in the <i>offered-capabilities</i> field of <i>open</i> then the sender of <i>open</i> performative supports anonymous terminus with the semantics defined in this document. If not present then the receiver of <i>open</i> MUST assume that its partner does not support the anonymous terminus.</p> <p>If present in the <i>desired-capabilities</i> field of <i>open</i> then the sender of <i>open</i> MAY use the anonymous if its partner offers its support. If it is not present then the sender of <i>open</i> MUST NOT attempt to use the anonymous terminus supports even if its partner offered the capability in the <i>open</i> performative which it sent.</p>

3 Conformance

This specification implicitly defines two roles which an AMQP container may take: a *provider* of the *anonymous terminus*, and a *user* of the *anonymous terminus*. An AMQP container may choose to implement none, one or both of these roles.

- A container acting as a *provider* of the *anonymous terminus* MUST indicate this by supplying the relevant capability in the *offered-capabilities* field of *open* as defined in 2.3 Anonymous Terminus. Further the container MUST support a target with null address which conforms with the definitions given in 2.2 Routing Nodes.
- A container acting as a *user* of the *anonymous terminus* MUST indicate this by supplying the relevant capability in the *desired-capabilities* field of *open* as defined in 2.3 Anonymous Terminus.

Appendix A. Acknowledgements

The following individuals have participated in the creation of this specification and are gratefully acknowledged:

- Robert Gemmell (Red Hat)
- Robert Godfrey (Red Hat)
- David Ingham (Red Hat)
- Jakub Scholz (Red Hat)
- Gordon Sim (Red Hat)
- Clemens Vasters (Microsoft)
- Keith Wall (Red Hat)

Appendix B. Revision History

Revision	Date	Editor	Changes Made
WD01	23-Mar-2017	Rob Godfrey	Initial Revision
WD02	13-Apr-2017	Rob Godfrey	AMQP-111 : Typos and repeated words AMQP-112 : Detail behaviour when addressed node doesn't exist
WD03	6-Oct-2017	Rob Godfrey	AMQP-124 : Add a (minimal) conformance section
WD04	23-Mar-2018	Rob Godfrey	AMQP-140 : Make the behaviour of the anonymous terminus clear when faced with transactional pre-settled messages
WD05	13-Jun-2018	Rob Godfrey	AMQP-124 : Update conformance section to define roles
WD06	15-Jun-2018	Rob Godfrey	Fix typos