
2 The Reliability agreement

2.1 Definition

A Reliability agreement for messaging - or RM Agreement - describes which reliability features a sending party and a receiving party have agreed to use when exchanging a set of messages. The RM Agreement can be seen as a contract at two levels:

- A Quality of Service (QoS) contract between the messaging layer (including RMPs involved) and the application layer. The contract describes the delivery conditions (from receiving RMP to receiver application) of each message submitted (from sender application to sending RMP).
- A Protocol contract between sending RMP and receiving RMP. At a lower level, this contract is about the cooperation between RMPs. For example, with Guaranteed Delivery, the receiving RMP must acknowledge to the sending RMP the messages that it delivered.

These two aspects of an agreement are intertwined: QoS requirements will mandate the use of particular protocol features.

2.2 Representation and Deployment

The concrete representation of an RM Agreement is beyond the scope of this specification, as this may be part of a more general agreement that exceeds the reliability aspect. However, the RM Agreement determines the use of the reliability protocol and the behavior of RMPs. For these reasons, this specification describes the RM Agreement in an abstract way, simply as a list of (name, value) pairs, called Agreement Items. This allows for describing the concrete effect of each Agreement Item on the message content and flow, without imposing a particular representation to the RM Agreement. Once there is a broad enough consensus for using a particular representation for agreements, a future version of this specification will define a corresponding binding for RM Agreements.

The way an RM Agreement is established or communicated to each party is out of scope. However, one of the principles of this specification – at least for the reliability features described here – is that it should not be necessary to deploy an RM Agreement on both sending and receiving RMPs prior to executing business transactions. Only the sending RMP needs to initially have knowledge of the RM Agreement. No prior communication of the agreement to the receiving party (RMP and its application) is required. The only input that the receiving RMP will need in order to enforce the reliability requirements will be obtained from the header of received messages (or by the absence of such messages).

As a way to support the creation of RM Agreements, it may be useful for Web services providers to advertise somehow the reliability features (or RM Agreement Item values) that are supported by a deployed Web service. Such capabilities (that we could call RM Capabilities, in contrast to agreements that involve both parties) may conveniently be associated with WSDL definitions. In support of this option, this specification proposes a concrete representation for these capabilities (see Appendix ...)

2.3 RM Agreement Items

An RM Agreement is a list of Agreement Items. An RMP implementation **MUST** be capable of:

(1) taking knowledge (e.g. either via configuration, or via an API call, or via a message, or via the result of an algorithm) of a set of values that represent the RM Agreement Items described in this specification,

(2) processing them according to the semantics described in this specification.

The following list of Agreement Items is considered by this specification. Each item is listed with its possible values:

- GuaranteedDelivery (enabled/disabled): for setting Guaranteed Delivery. (See Section 3.1 for details)
- NoDuplicateDelivery (enabled/disabled): for setting message delivery without duplicates, or Duplicate Elimination. (See Section 3.2 for details)
- OrderedDelivery (enabled/disabled): for setting Guaranteed Message Ordering. (See Section 3.3 for details)
- GroupMaxIdleDuration (number of seconds): For setting the elapsed time limit from the last message sent or received in a group, after which the group can be terminated.
- GroupExpiryTime (date/time): For setting the date and time after which the group can be terminated.
- ExpiryTime (date/time): For setting the date and time after which a message must not be delivered to the receiving application.
- ReplyPattern ("response", "callback", "poll") For setting the mode of response for Acknowledgments or Faults.

2.4 Messaging Scope of Agreement Items

The messaging scope of these agreement items may vary, as messages may be associated with a group. There are three scopes to consider:

- (s1) All messages sent over a connection between a sending RMP and a receiving RMP (default).
- (s2) All messages sent within a group.
- (s3) A single message, within a group of several messages (non-singleton group).

Some agreement items obviously relate to a particular scope, e.g. ExpiryTime is affecting each message separately, while GroupExpiryTime is an agreement item about groups.

The smallest scope of applicability for each RM Agreement item is:

Message scope (s3):

- ExpiryTime
- ReplyPattern

Group scope (s2):

- OrderedDelivery
- GuaranteedDelivery
- NoDuplicateDelivery

112 • GroupExpiryTime

113 • GroupMaxIdleDuration

114 An RMP MAY support RM Agreement items at a scope that is higher than the smallest scope
115 allowed. For example, an RMP implementation may decide to provide a way to specify the same
116 ExpiryTime value for all messages of a group.

117 An RMP MUST NOT support RM Agreement items at a scope that is lower than the smallest
118 scope allowed. For example, an RMP implementation MUST NOT use different guaranteed
119 delivery modes for different messages of a group.

120 **2.5 Rules about Agreement Items**

121 When defining an RM Agreement instance, there are some dependencies between the items of
122 the agreement that must be respected:

123 • If GuaranteedOrdering is enabled for a messaging scope, then GuaranteedDelivery
124 and NoDuplicateDelivery MUST also be enabled for that messaging scope.

125 • If GroupExpiryTime is enabled for a messaging scope, then the item
126 GroupMaxIdleTime MUST NOT be enabled, and vice versa.

127

128

129

