ELEXON

DIP Manager Replay and Requeue Guidance Note

Date
April 2026

Document owner
DIP Manager

Paper number
V1.0

Synopsis

This guidance note explains when a message should be:

- **replayed**
- **requeued**
- **resent**

In summary:

- **Replay** or **requeue** should normally be used when the DIP successfully received the message, but the recipient cannot access it after delivery, or the DIP was unable to complete delivery after retries were exhausted.
- **Parties should resend** when the message failed before or during DIP processing

1. Introduction

1.1. This guidance note helps DIP Participants determine the correct recovery action where a message is missing, unavailable or has failed.

1.1.1. The correct action depends on where the failure occurred.

1.1.2. As a general principle:

- use replay or requeue where the message was successfully received by the DIP
- resend the message when it did not successfully enter or complete processing in the DIP

2. Replay and Requeue

2.1. Replay or requeue should normally be used where:

- The message was successfully received by the DIP
- The recipient did not retain or process the message correctly after delivery, or
- The DIP was unable to complete delivery after retry attempts were exhausted

2.1.1. This applies where the message exists within DIP processing or delivery records.

2.1.2. Replay and Requeue can be done either via the DIP Portal or the preferred method the Replay/Requeue API

2.2. Replay or requeue should not be used where the message never entered the DIP successfully.

2.2.1. In these cases, the sender will need to resend the message.

2.3. Technical Considerations for Replay API use

2.3.1. The Replay API returns a synchronous response to the request.

2.3.2. DIP Manager has observed cases where the calling party's system times out when using the Replay API. This is typically because the DIP appears non-responsive while a large volume of message payloads is being returned as part of the response.

2.3.3. To reduce the risk of timeout, recipients are advised not to use the maximum batch size of 100 Transaction IDs in a single request. Instead, Participants should use a lower number of Transaction IDs and make multiple API calls where needed.

3. Message Resend

3.1. Messages should be resent when

- The message failed at ingress
- The message failed during DIP processing and could not be progressed
- The sender's own systems did not submit the message successfully

3.1.1. In these cases, the sender is responsible for resubmitting the message once the issue has been resolved.

3.1.2. Depending on the failure the sender may need to generate a new SUR, these will need to be managed on an error by error basis.