

## Store and Forward

[</> See also API reference for Credit Transfer](#)

[Check API >](#)

DuitNow is designed with Store and Forward (SAF) capability taking into consideration there will be a huge amount of messages being exchanged between a large number of participants, many of whom may not be online at the time of transmission. It removes the uncertainty and inconvenience due to an outage or connectivity being unreliable from a recipient.

---

### Benefits of Store and Forward

- To avoid interruptions especially when a participant is offline due to an unplanned outage. For example, customer is making payment to a recipient where the receiving institution is offline. In this case, the customer needs not to inquire the status of the recipient prior to making the transfer.
  - To avoid the need of having manual reconciliation especially when the last leg of a payment request failed due to network error. For example, when a customer initiates a debit request, the fund has been debited successfully but the crediting request has failed due to network timeout.
  - To control the volume of traffic being sent and received, and the overall response time to ensure any slow response from any participant would not cause a catastrophic failure in our services.
- 

### Store and Forward Queue

Messages are stored in a Store and Forward Queue (SAF) based on transaction code that the processing session assigns to the message. A status reason U002 (SAF acknowledgement) is returned to acknowledge a message has been routed through SAF. If SAF is disabled, messages are sent immediately online. If there is no response to a request, it is repeated up to the maximum retry limit and placed in an error state if there is still no response.

---

## SAF Throttle Parameters

Depending on the situation, PayNet could configure with different throttling parameters without prior notice to maintain best possible performance for our services.

Parameter	Description
SAF Maximum Distribution Rate	Controlling the maximum number of messages permitted to be sent from SAF per second to avoid participant from being flooded with too many SAF requests that could lead to performance degradation, especially after a long period of outage.
SAF Message Queue Limit	Controlling the maximum number of SAF messages can be accepted by RPP. This prevents the system to be flooded with excessive SAF requests that would deteriorate the overall system performance.
SAF Message Retry Limit	SAF could resend a request up to a maximum number of retry limit. This is to prevent system from retrying indefinitely. And in the event participant is having a long outage and not being signed off from the network, RPP could re-mobilise its system resources.
SAF Open Session Limit	The maximum number of concurrent messages permitted to be sent to participant but waiting for a response. This is used to prevent RPP from

Parameter	Description
	being overloaded with too many open SAF sessions due to participant having high latency that could jeopardize the overall response time.
SAF Pause Period	System could pause the transmission of any message after exceeding the number of consecutive SAF timeout threshold. In such case, participant could be unavailable and it's best to stop all processing.

## How It Works

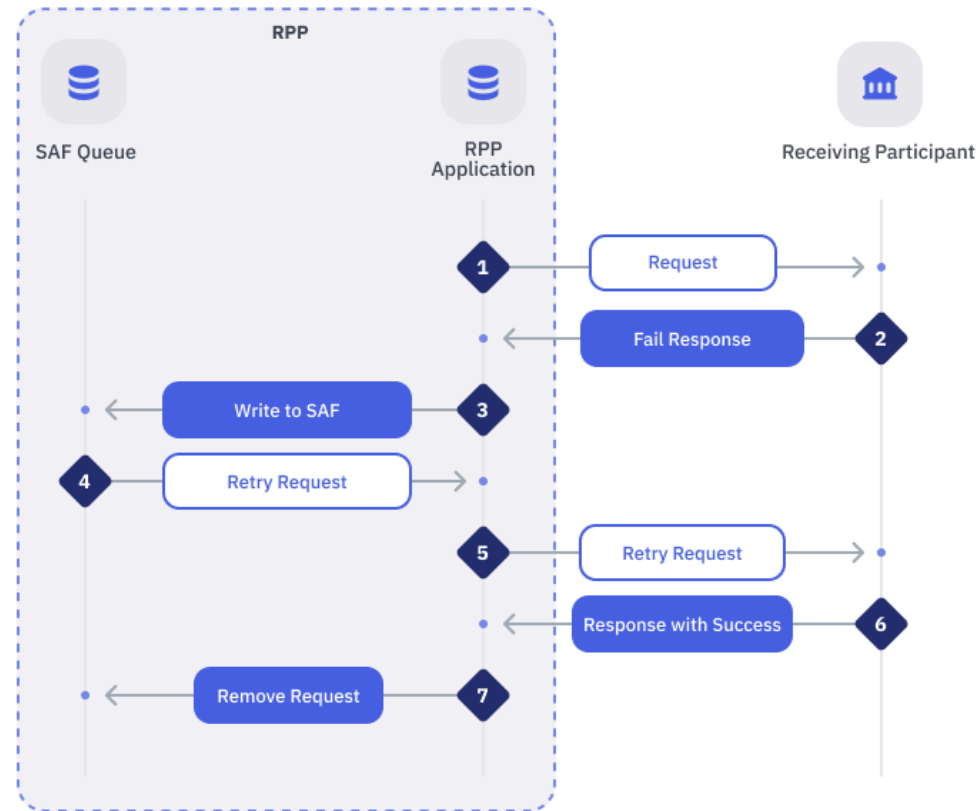
SAF works as a retry mechanism that monitors a request and on the detection of failure, it shall re-attempt the request by placing it into a destined queue. Attempting to retry on any and every transaction is not advisable due to the adverse impact to system resources. Due to that, only selected transactions are set to have this capability

This feature is used internally by RPP on selected transaction codes to minimize the failure rate of a critical transaction not being able to be processed by receiver due to various technical issues e.g. network error, timeout and etc

## How to handle SAF request

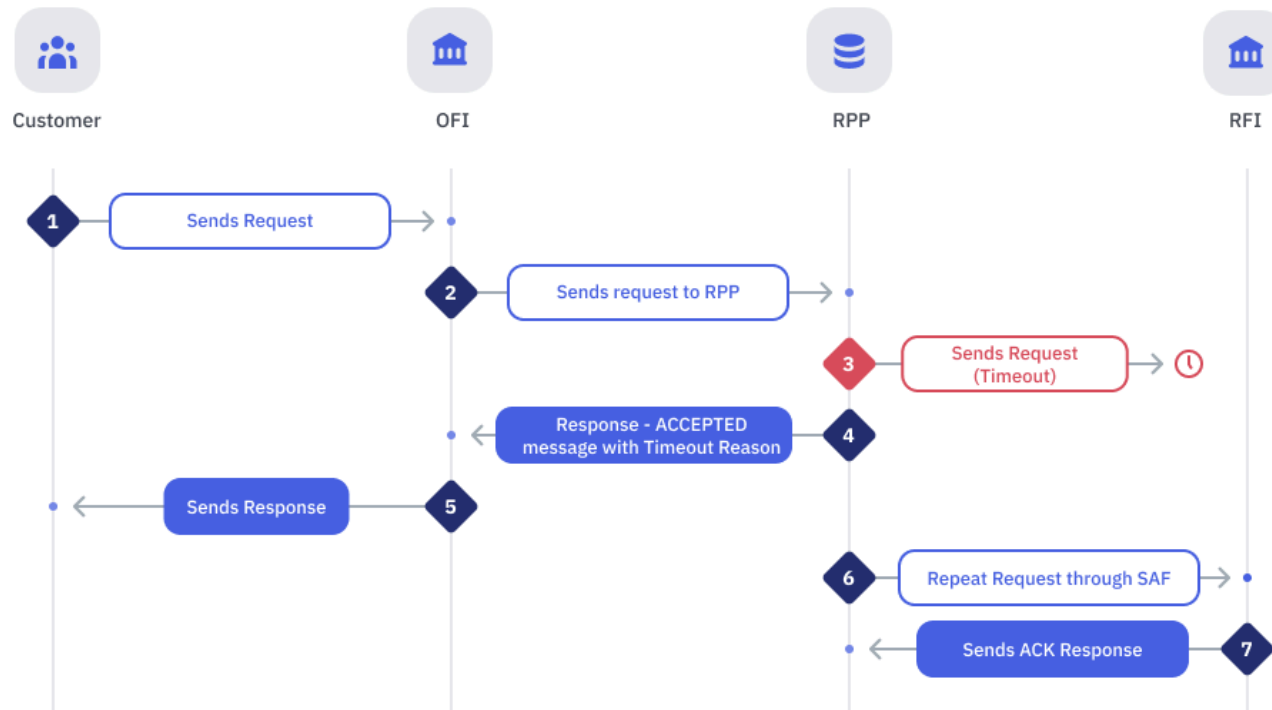
- The SAF transactions will be resent automatically to the RFI up to the maximum retries in case of timeout
- RFI cannot reject transactions coming from the SAF queue
- If the maximum number of retries has been exceeded and still no response received from RFI, the system will mark the transaction as timeout and the transaction will remain in SAF until it is cleared it manually

- Manual reconciliation might be required for extreme cases like response from RFI was negative, but OFI receives an ACCEPTED



## Exception Flows

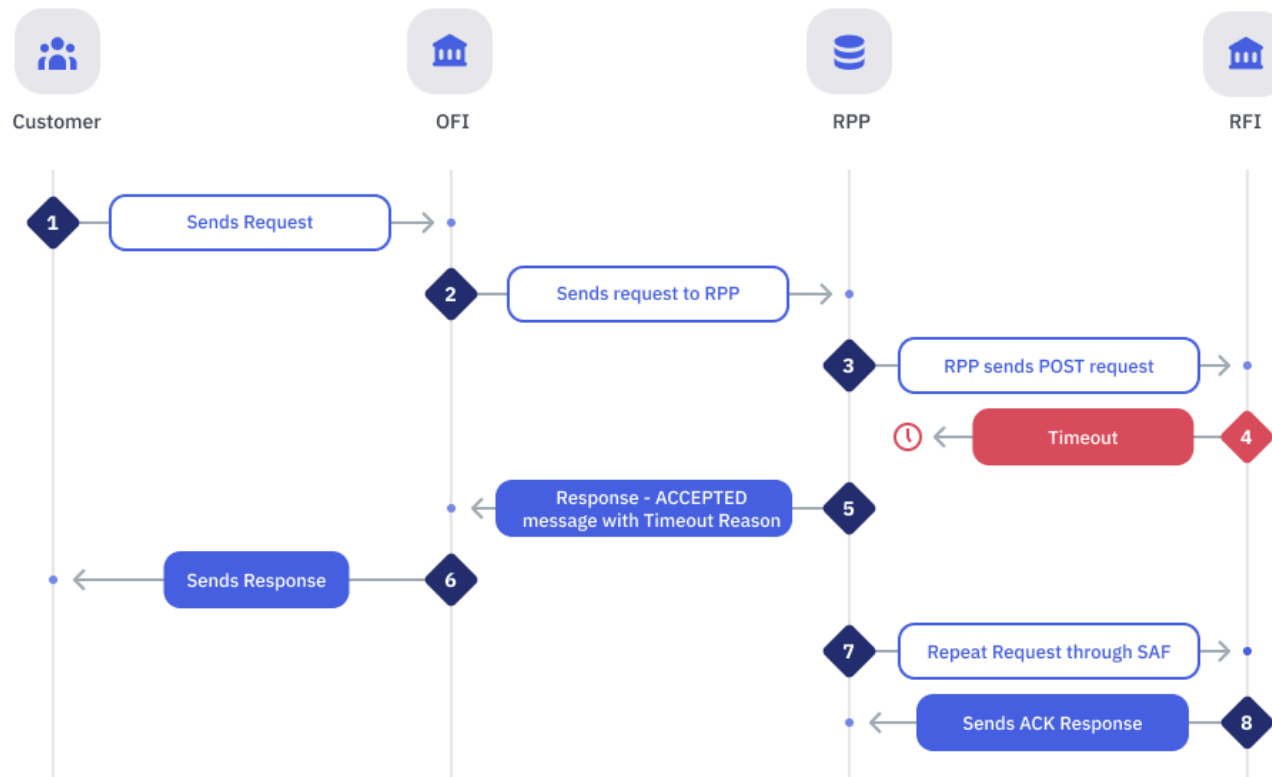
### RFI Failed to Receive Request from RPP



Condition	Action	Alternatives
<p>In Step 3, RPP sent a request to RFI. However, RFI did not receive the request from RPP</p> <p>As no response is received from RFI after x period of time, RPP eventually timeout</p>	<p>RPP shall:</p> <ul style="list-style-type: none"> <li>• Timeout</li> <li>• Return a NEGATIVE response with timeout reason</li> </ul> <p>OFI shall:</p> <ul style="list-style-type: none"> <li>• Display an appropriate error message to the Customer</li> <li>• Stop processing</li> </ul>	<p>If retry is configured, RPP shall:</p> <ul style="list-style-type: none"> <li>• Store the request to SAF</li> <li>• If retry count has not exceeded the maximum number of retries               <ul style="list-style-type: none"> <li>◦ Send a repeat request</li> <li>◦ Continue processing</li> </ul> </li> </ul>

Condition	Action	Alternatives
	<ul style="list-style-type: none"><li>• Initiate a Transaction Enquiry separately to confirm on the payment status</li></ul>	<ul style="list-style-type: none"><li>• If retry count has exceeded the maximum number of retries<ul style="list-style-type: none"><li>◦ Display an ACCEPTED status message to Customer</li><li>◦ Stop processing</li><li>◦ Confirm the payment manually via Back Office portal</li></ul></li></ul>

## RPP Failed to Receive Response from RFI



Condition	Action	Alternatives
<p>In Step 3, RPP sent a request to RFI, and RFI did receive the request. However, RFI response did not reach to RPP.</p> <p>As no response is received from RFI after x period of time, RPP eventually timeout</p>	<p>RPP shall:</p> <ul style="list-style-type: none"> <li>• Timeout</li> <li>• Return a NEGATIVE response with timeout reason</li> </ul> <p>OFI shall:</p> <ul style="list-style-type: none"> <li>• Display an appropriate error</li> </ul>	<p>If retry is configured, RPP shall:</p> <ul style="list-style-type: none"> <li>• Store the request to SAF</li> <li>• If retry count has not exceeded the maximum number of retries               <ul style="list-style-type: none"> <li>◦ Send a repeat request</li> <li>◦ Continue processing</li> </ul> </li> </ul>

Condition	Action	Alternatives
	<p>message to the Customer</p> <ul style="list-style-type: none"><li>• Stop processing</li></ul>	<ul style="list-style-type: none"><li>• If retry count has exceeded the maximum number of retries<ul style="list-style-type: none"><li>◦ Display an ACCEPTED status message to Customer</li><li>◦ Stop processing</li><li>◦ Confirm the payment manually via Back Office portal</li></ul></li></ul>