[](https://www.dtmercury.com/)

A blue and white background

Description automatically generated

**Mercury API Specification**

**Processor – Payment Status Report**

**Outbound To Client**

Table of Contents

Confidentiality Notice: This document contains private information about Mercury (Pty) Ltd. You can't copy, reproduce, or share its product details, intellectual property, or design methods with anyone else unless you have our written consent.

[1) Document Purpose 3](#_Toc167795958)

[2) Process Flow 4](#_Toc167795959)

[3) Clearing Payment Status Report Message to Participant 6](#_Toc167795960)

[4) Embedded Participant Message Structures 17](#_Toc167795961)

# Document Purpose

The purpose of this document is to provide a detailed specification of the **Payment Status Report** call to be sent by the **Mercury Processor API** to the **Mercury Client** as part of the **Mercury Credit Transfer** Process.

This document will focus on the **Outbound Message** from the **Mercury Processor API** to the **Mercury Client** and will indicate the construction and processing of the **Payment Status Report** message.

**NOTE:** Some of the values contained in the Outgoing Message Structure will be retrieved from the Configuration Settings that are specific to the Mercury Processor API for the specified Entity.

# Process Flow

A screenshot of a diagram

Description automatically generated

The **Mercury Credit Transfer** process flow indicated above is initiated on the **Mercury Client** and the request will be processed from the **Mercury Client** to the **Mercury Processor API**. After decryption and validation of the message by the **Mercury Processor API**, a Transaction Ledger Entry will be created (Hold) for the Transaction (Linked to the Client Transaction and Internal Mercury Reference) and then the **Credit Transfer Request (Out to Host)** message will be constructed and sent to the **Host (Enterprise or External).**

The response to the **Mercury Client** will be based on the decryption and validation results and sent to the **Mercury Client** before proceeding with the Transaction Ledger Entry, Out to Host Message construction and sending of the message to the **Host (Enterprise or External).**

If the request to the **Host (Enterprise or External)** was successful (indicated by a **200 HTTP Response**), the request will be processed by the **Host (Enterprise or External).**

The **Host (Enterprise or External)** will send the response **(Payment Status Report),** as a separate message (Not as a response to the original Credit Transfer Request) based on the result of the processing of the Transaction. The **Payment Status Report (In from Host)** will be processed from the **Host (Enterprise or External)** to the **Mercury Host API**. After decryption and validation of the message by the **Mercury Host API**, the associated Transaction Ledger Entry will be Updated as **Settled** (Based on the Internal Mercury Reference) and then the **Payment Status Report (Out to Client)** message will be constructed and sent to the **Mercury Client**.

The response to the **Host (Enterprise or External)** will be based on the decryption and validation results and sent to the **Host (Enterprise or External)** before proceeding with the Transaction Ledger Entry Update, Out to Client Message construction and sending of the message to the **Mercury Client**.

Should the **Mercury Credit Transfer Request** to the **Host (Enterprise or External)** fail, the **Mercury Processor API** will, update the Transaction Ledger (set the Transaction as Failed) and then generated a **Payment Status Report (Out to Client)** indicating the reason and details for the failed transaction, to the **Mercury Client.**

# Clearing Payment Status Report Message to Participant

## Overview

This **Payment Status Report API** is invoked during the processing of the **Payment Status Report** received from the **Host (Enterprise or External).**

## Process Flow

A screenshot of a computer

Description automatically generated

As part of processing the **Mercury Payment Status Report** from the **Host (Enterprise or External),** the **Mercury Payment Status Report** to the **Mercury Client** need to be created as this message will contain less information and values that what is received in the message received from the **Host (Enterprise or External).**

As part of the basic API Processing the following steps need to be followed:

|  |  |
| --- | --- |
| **STEP** | **DESCRIPTION** |
| 1. | Initialize the **Payment Status Report (Out to Client)** Message Class with the details of the **Payment Status Report (In from Host)** Message received from the **Host (Enterprise or External**).  Message Class: **procOTCPaymentStatusReport** |
| 2. | Retrieve the Subscriber ID, Primary Host Reference, Secondary Host Reference, and Mercury Client Endpoint from the **Client API Configuration (cac\_Client\_API\_Configuration)** table based on the followings fields contained in the Message from the **Host (Enterprise or External):**   * Subscriber Id   **IMPORTANT:** This is necessary to ensure that the message is sent to the correct Mercury Client. |
| 3. | Construct the **Secure Token** (that is required between the **Mercury Processor API** and **Mercury Client API)** to ensure that messages have not been tampered with in transit.  This Secure Token need to be constructed as an encrypted value based on the values in the constructed message to be sent.  Once constructed, populated the **secureToken** value in the **Payment Status Report Message (Out to Client)** Message Structure. |
| 4. | Store and Log the Message Request Details in the Message Request Detail table for auditing purposes.  **Table Name**: mrd\_Message\_Request\_Detail |
| 5. | Process the message to the **Mercury Client API** and log the response from the **Mercury Client API** against the Entry that was added to the Message Request Detail table.  **NOTE:** Should the response from the **Mercury Client API** indicate any validation failures or system errors, a specific notification should be sent to the relevant email and standby to ensure this is investigated and resolved. Possible Retry scenarios might be added later in the project.  The response from the **Mercury Client API** will be received as a [**Message Request Response**](#Response) structure. |
| 6. | Based on the **HTTP Response Code** (other than 200 Indicating failure) you will need to retry the sending of the request based on the **API Protocols Configuration** specified by the **Mercury**. See [**Response Codes**](#ResponseCodes).  After the retry instructions specified in the **API Protocols Configuration** have been met, and the processing of the request to the **Mercury Client** is still not successful, and notification must be generated to alert of the inability to send the request. |

## Request

|  |  |
| --- | --- |
| **REQUEST DETAIL** | |
| **ENDPOINT** | {Participant Endpoint}/procOTCPaymentStatusReport |
| **METHOD** | POST |

|  |  |
| --- | --- |
| **REQUEST BODY** | |
| **CLASS NAME** | clsOTCPaymentStatusReport |
| **FIELD** | **DESCRIPTION** |
| secureToken | **Required**  Unique encrypted value constructed from the field values contained in the message.  **Format:**  String |
| subscriberID | **Required**  Unique Subscriber ID that is issued by the Payment Hub to uniquely identify the source from where the call is to be sent.  **Format:**  String  **Example:**  902 |
| primaryHostReference | **Optional**  Primary Reference Value received in the Payment Status Report received from the Host (Enterprise, External or Payments Hub). This will be the same value populated in the original Credit Transfer Request linked to this Payment Status Report. This value can also be retrieve using the Original Message Identification to locate the value in the original Credit Transfer Request received.  **Format:**  String |
| secondaryHostReference | **Optional**  Secondary Reference Value received in the Payment Status Report received from the Host (Enterprise, External or Payments Hub). This will be the same value populated in the original Credit Transfer Request linked to this Payment Status Report. This value can also be retrieve using the Original Message Identification to locate the value in the original Credit Transfer Request received.  **Format:**  String |
| messageIdentification | **Required**  Unique Message Identification as received in the request received from the **Host (Enterprise, External or Payments Hub).** This value is determined by the **Operator (Bankserv).**  **Format:**  String: alphabetic, numeric, and underscores (\_), e.g. spaces are not allowed; max length 35 characters. This can also be a value that does not conform to the recommended format.  **Recommended Format:**  YYYYMMDD\_XXXX\_TTTTT\_999999999999999   * **YYYYMMDD** - Date the message is created * **XXXX** – First 4 characters of the senders **BICCode** * **TTTTT** – Service Code * **999999999999999** – 15-digit sequence number (for the current day – reset after day)   **Example:**  2021117\_BKSV\_PSRPT\_000000000000001 |
| creationDateTime | **Required**  Creation Date Time of the request from the Payee Participant. SAST to UTC time format in both incoming & outgoing messages in header across all the messages.  **Format:** String  yyyy-MM-ddThh:mm:ss.SSSSSS  **Example:**  2021-11-17T11:27:53.015708 |
| transactionIdentifier | **Required**  Unique Identification value assigned by the Mercury Processing System to unambiguously identify the Payments Status Report message processed by the Mercury System.  **Format:**  String  **Example:**  54000000000000001 |
| originalProcessingTransactionId | **Required**  Mercury Transaction Identifier that was assigned to the original Credit Transfer Request.  **Format:**  String  **Example:**  CAE4F68F-BFB1-4FB2-B2B0-4AA94096142E |
| originalTransactionIdentifier | **Required**  Transaction Identifier that was assigned by the Mercury Client to the original Credit Transfer Request.  **Format:**  String  **Example:**  1A66C164-0EDA-4170-8D13-FBF9E495B680 |
| originalUETR | **Required**  Transaction UETR that was assigned by the Mercury Client to the original Credit Transfer Request.  **Format:** String  [a-f0-9]{8}-[a-f0-9]{4}-4[a-f0-9]{3}-[89ab][a-f0-9]{3}-[a-f0-9]{12}  **Example:**  f22dd32f-07db-488b-b648-dab6b3ffc4ef |
| transactionStatus | **Required**  Status of the transaction reported by the Payee Participant.  **Format:**  String; max length 4 characters.  **Restrictions:**  It is one of the following values:   * **ACCP** (Accepted) * **RJCT** (Rejected) * **ACCC** (Accepted Settlement Completed) * **PDNG** (Pending) * **CANC** (Cancelled) – Payee Only |
| transactionStatusDescription | **Required**  Further details on the status reason. Can be a combination of the Status Reason Code and Status Reason Description received from the Host (Enterprise, External or Payments Hub).  **Usage**  Additional information can be used for several purposes such as the reporting of repaired information.  **Format:**  String; max length 105 characters. |

## Response

API responses will return HTTP status code in the 2xx range. The response body will contain the response of the API as described in this specification.

|  |  |
| --- | --- |
| **RESPONSE** | |
| **CLASS NAME** | clsMessageRequestResponse |
| **FIELD** | **DESCRIPTION** |
| messageIdentification | **Required**  Unique Message Identification for the Message Request for which this response structure is created.  **Format:**  String: alphabetic, numeric, and underscores (\_), e.g. spaces are not allowed; max length 35 characters.  YYYYMMDD\_XXXX\_TTTTT\_999999999999999   * **YYYYMMDD** - Date the message is created * **XXXX** – First 4 characters of the senders **BICCode** * **TTTTT** – Service Code * **999999999999999** – 15-digit sequence number (for the current day – reset after day)   **Example:**  2021117\_BKSV\_PSRPT\_000000000000001 |
| messageResultDetail | **Required**  A [**Message Result**](#MessageResult) structure. |

## Response Codes

Internal Response Codes available for the Message Request Response are listed below. The Response Codes will have a format of **ECLABBCCCC** and is compiled as follows:

**ECL** Fixed value for the Response Code Prefix

**A** Service Indicator where **8** will indicate the **Processor API**, **9** will indicate the **Host (Enterprise, External or Payments Hub)** and **7** will indicate the **Mercury Client** as the origin of the response.

**BB** Message Type Indicator (2 digits) where **08** will indicate the **Payment Status Report** Message Type.

**CCCC** Unique value per Response indicating a unique Response.

|  |  |  |
| --- | --- | --- |
| **RESPONSE CODES** | | |
| **HTTP STATUS** | **RESPONSE CODE** | **RESPONSE DESCRIPTION** |
| 200 | ECL7080000 | **SUCCESS** |
| 400 | ECL7080001 | **BAD REQUEST (NO RETRY)** |
| 401 | ECL7080002 | **UNAUTHORIZED (RETRY)** |
| 403 | ECL7080003 | **FORBIDDEN (RETRY)** |
| 404 | ECL7080004 | **NOT FOUND (RETRY)** |
| 408 | ECL7080005 | **REQUEST TIME OUT (RETRY)** |
| 429 | ECL7080006 | **TOO MANY REQUESTS (NO RETRY)** |
| 500 | ECL7080007 | **INTERNAL SERVER ERROR (NO RETRY)** |
| 502 | ECL7080008 | **BAD GATEWAY (NO RETRY)** |
| 503 | ECL7080009 | **SERVICE UNAVAILABLE (RETRY)** |
| 504 | ECL7080010 | **GATEWAY TIMEOUT (RETRY)** |
| 511 | ECL7080011 | **NETWORK AUTHENTICATION (RETRY)** |
| 200 | ECL7080012 | ERROR\_INVALID\_FIELD\_TYPE |
| 200 | ECL7080013 | ERROR\_INVALID\_FIELD\_LENGTH |
| 200 | ECL7080014 | ERROR\_INVALID\_FIELD\_VALUE |
| 200 | ECL7080015 | ERROR\_REQUIRED\_DATA\_MISSING |
| 200 | ECL7080016 | ERROR\_INTERNAL\_SYSTEM\_ERROR |
| 200 | ECL7080017 | ERROR\_CRYPTOGRAPHY |
| 200 | ECL7080021 | ERROR\_DUPLICATE\_TRANSACTION |

## Sample Request

{

    "secureToken": "string",

    "subscriberID": "string",

    "primaryHostReference": "string",

    "secondaryHostReference": "string",

    "messageIdentification": "string",

    "creationDateTime": "string",

    "transactionIdentifier": "string",

    "originalProcessingTransactionId": "string",

    "originalTransactionIdentifier": "string",

    "originalUETR": "string",

    "transactionStatus": "string",

    "transactionStatusDescription": "string"

}

## Sample Request Message Accepted

{

    "secureToken": "07ac3cf0add872d8af56e2ac62213b9add6a4fb61977e1fe109c54dda67c1c48",

    "subscriberID": "910",

    "primaryHostReference": "2023083100001",

    "secondaryHostReference": "78602467923",

    "messageIdentification": "20231003\_BKSV\_PSRPT\_000000000000002",

    "creationDateTime": "2023-10-03T11:27:55.015708",

    "transactionIdentifier": "54000000000000001",

    "originalProcessingTransactionId": "CAE4F68F-BFB1-4FB2-B2B0-4AA94096142E",

    "originalTransactionIdentifier": "1A66C164-0EDA-4170-8D13-FBF9E495B680",

    "originalUETR": "a1182b47-12ec-3655-8a94-8bb3d95bea55",

    "transactionStatus": "ACCC"

}

## Sample Request Message Rejected

{

    "secureToken": "07ac3cf0add872d8af56e2ac62213b9add6a4fb61977e1fe109c54dda67c1c48",

    "subscriberID": "910",

    "primaryHostReference": "2023083100001",

    "secondaryHostReference": "78602467923",

    "messageIdentification": "20231003\_BKSV\_PSRPT\_000000000000002",

    "creationDateTime": "2023-10-03T11:27:55.015708",

    "transactionIdentifier": "54000000000000001",

    "originalProcessingTransactionId": "CAE4F68F-BFB1-4FB2-B2B0-4AA94096142E",

    "originalTransactionIdentifier": "1A66C164-0EDA-4170-8D13-FBF9E495B680",

    "originalUETR": "a1182b47-12ec-3655-8a94-8bb3d95bea55",

    "transactionStatus": "RJCT",

    "transactionStatusDescription": "Transaction Rejected by Operator [AG10]"

}

## Sample Response

{

    "messageRequestID": "string",

    "messageResultDetail": {

        "resultCode": "string",

        "resultDescription": "string",

        "resultMessage": "string",

        "processingTransactionId": "string"

    }

}

## Sample Response Message Success

{

    "messageRequestID": "20231003\_BKSV\_PSRPT\_000000000000002",

    "messageResultDetail": {

        "resultCode": "ECL7080000",

        "resultDescription": "SUCCESS",

        "resultMessage": "",

        "processingTransactionId": "70adf9df-dedc-4d25-9af0-ff90cc722130"

    }

}

## Sample Response Message Error

{

    "messageRequestID": "20231003\_BKSV\_PSRPT\_000000000000002",

    "messageResultDetail": {

        "resultCode": "ECL7080015",

        "resultDescription": "ERROR\_REQUIRED\_DATA\_MISSING",

        "resultMessage": "The Message Identifier field is required.",

        "processingTransactionId": "70adf9df-dedc-4d25-9af0-ff90cc722130"

    }

}

# Embedded Participant Message Structures

## Message Result

**Note:** This structure is used in various Message Request Structures in the Processor and Participant API to and from the Participant API.

|  |  |
| --- | --- |
| **MESSAGE RESULT** | |
| **CLASS NAME** | clsMessageResult |
| **FIELD** | **DESCRIPTION** |
| resultCode | **Required**  Appropriate Response Code for the Message Type  **Format:** String; length 10 characters. |
| resultDescription | **Required**  Appropriate Response Description for the Message Type  **Format:** String; max length 50 characters. |
| resultMessage | **Optional**  Additional Response Detail associated with the Response Code and Description (Example: Field Name that failed the validation)  **Format:** String; max length 256 characters. |
| processingTransactionId | **Conditional**  Based on whether the Payment Status Report was accepted by the Mercury Client, return the Mercury Client Transaction Id that was assigned for the request on the Client System. |