



# **New Payments Platform**

## **API Framework**

**Publication Version 1.0**

28 September 2018

NPP Australia Limited and SWIFT SCRL

The authors NPP Australia Limited and SWIFT accept no liability to any third party in respect of the contents of this document.

©NPP Australia Limited and SWIFT© 2018. All rights reserved. No reproduction or translation of this document may be made without prior permission of the authors. Requests for permission may be directed to [secretary@nppa.com.au](mailto:secretary@nppa.com.au).

SWIFT is the trading name of S.W.I.F.T. SCRL

## Contents

|          |   |          |
|----------|---|----------|
| <b>1</b> | <b>Introduction</b> .....                   | <b>3</b> |
| 1.1      | Purpose & Scope .....                       | 3        |
| 1.1.1    | Initial sample APIs.....                    | 3        |
| 1.2      | Approach.....                               | 4        |
| 1.3      | Glossary & abbreviations .....              | 4        |
| <b>2</b> | <b>NPP Open API Design Principles</b> ..... | <b>6</b> |
| 2.1      | Basics.....                                 | 6        |
| 2.1.1    | RESTful APIs.....                           | 6        |
| 2.1.2    | ISO 20022 .....                             | 6        |
| 2.1.3    | Security Standards .....                    | 7        |
| 2.1.4    | NPP Overlay Service Identification .....    | 7        |
| 2.1.5    | Status Codes .....                          | 7        |
| <b>3</b> | <b>NPP Open API Use Cases</b> .....         | <b>8</b> |
| 3.1      | Payment Initiation .....                    | 8        |
| 3.1.1    | Payment Flow Processing Steps.....          | 8        |
| 3.1.2    | Sample API Documentation .....              | 9        |
| 3.1.3    | Usage Example .....                         | 10       |

## 1 Introduction

The New Payments Platform (NPP) is new national infrastructure for fast, flexible, data rich payments in Australia. NPP Australia Limited (NPPA) has engaged SWIFT to design, build and operate the basic infrastructure of NPP.

This document has been jointly developed by NPPA and SWIFT to promote development of the NPP ecosystem and to assist NPP Participants, Third Party Service Providers and software developers with the development of API solutions for NPP transactions.

In order to maximise standardisation and interoperability, and to provide a consistent NPP experience, NPP Participants, Third Party Service Providers and software developers are encouraged to use the NPP API Framework as a guide for the development of their own API solutions for NPP. This document provides a consolidated point of reference and provides an expanded view for NPP Participants, Third Party Service Providers and software developers to further their adoption of API technologies for NPP.

NPPA does not mandate use of this Framework. NPP Participants that utilise this Framework for the development of open APIs or partner-based APIs are free to use this Framework, and to publicise such use, as they see fit.

### 1.1 Purpose & Scope

The purpose of the NPP API Framework document is to provide guidance relevant to the design of APIs in the context of NPP. It is designed to support and facilitate NPP Participants' open APIs and partner-based APIs, to enable Third Party Service Providers and software developers to design NPP payment initiation services. The API framework is expected to encourage collaboration by creating a set of minimum standards primarily for the benefit of Third Party Service Providers and software developers, to remove the need to build multiple customised APIs for each NPP Participant and Overlay Service product.

Overlay Service Providers (**OSPs**) are encouraged to make available specific data usage or mandatory elements for use with the API framework. These elements will be defined and maintained by each OSP and published by NPPA together with this framework.

NPP Participants, Third Party Service Providers and software developers may identify different or additional considerations for their API design at a more detailed level which are not covered by this document.

The specific purpose of this document is to:

- Identify a common set of design principles and best practices that could be leveraged by the NPP community to reduce interoperability variation across core data exchanges when implementing API based technologies;
- At a high level, outline considerations to ensure the NPP community has a common reference of key API design considerations and how they may be approached by the industry; and
- Identify applicable standards for NPP data transfer, data constructs and security

#### 1.1.1 Initial sample APIs

**To provide Third Party Service Providers and software developers with an understanding of how to utilise the API Framework, three sample APIs have been included as part of the framework documentation. The sample APIs include usage guidelines in .pdf and excel format as well as a JSON example.**

The samples include the following:

- Look up of a PayID which will validate that the PayID exists and if valid return the associated short name and status - API: a11.001.01 (Get Alias Account);.
- Submit a payment initiation request to process a payment - API: pain.a09.001.01 (Post Customer Credit Transfer Initiation); and
- Confirm completion of payment - API: pain.a10.001.01 (Get Payment Status Report).

## 1.2 Approach

This document describes a development framework and refers to other related reference material.

To facilitate use of the document, each section has sub-sections to highlight key themes or processes NPP Participants and third parties should consider in their development of API solutions for NPP. It is intended to be of use to both existing and prospective NPP Participants, Third Party Service Providers and software developers.

The use of the API Framework document as a source of information does not affect or alter:

- (a) any rights or obligations of NPP Participants under the NPPA Regulations and NPP Procedures. For the avoidance of doubt, where there is any inconsistency between this document and the NPP Regulations and Procedures, or an NPPA-approved NPP design document, those documents prevail to the extent of the inconsistency.; or
- (b) the rights or obligations of NPP Participants as data controllers, to comply with privacy laws and to establish their own permission frameworks and requirements for secure data transfer.

## 1.3 Glossary & abbreviations in this document

| Term                     | Description  |
|--------------------------|--|
| Addressing Service       | Component of the Basic Infrastructure that enables registration of customer account information and PayID (account proxy) information      |
| API                      | Application Program Interface  |
| BI                       | NPP <u>B</u> asic <u>I</u> nfrast <u>r</u> ucture  |
| Connected Institutions   | Connect to the Basic Infrastructure solely for the purposes of sending and receiving Non-Value Messages.                                   |
| JSON                     | JavaScript Object Notation   |
| NPP Participant          | Connects to the Basic Infrastructure for the purposes of clearing and settling NPP Payments  |
| Overlay Service          | Refers to a payment service, or payment-related service, using the NPP Basic Infrastructure  |
| Overlay Service Provider | Overlay Service Providers provide Overlay Services to Overlay Service Subscribers. Overlay Service Providers may be Connected Institutions |

| Term                         | Description   |
|------------------------------|---|
| PayID                        | Refers to an alias record in the Addressing Service. The Addressing Service provides a mechanism that allows a registered PayID (acting as a unique identifier for a Customer Account) to be resolved to an Account. A PayID can be one of four alias types; phone number, email address, ABN or Organisational Identifier. |
| RESTful                      | Representational state transfer (REST) or RESTful web services  |
| Third Party Service Provider | Payment service providers that are 3rd parties (i.e. not any of the following: owner of the account, the account servicer or the account servicer's sponsoring Participant)   |

## 2 NPP Open API Design Principles

### 2.1 Basics

The following key design principles incorporate both RESTful concepts and ISO20022 as the data standard and take into consideration the flexibility of the NPP Overlay Service concept.

#### 2.1.1 RESTful APIs

Each NPP API should adhere to the RESTful API concepts as *the transfer standard of choice*.

A RESTful API is a set of Hypertext Transfer Protocol (HTTP) request messages, along with a definition of the structure of response messages, which is in a JavaScript Object Notation (JSON) format.

Overall, the priority should be to have an API that is simple to understand and easy to use. In instances where following RESTful principles would be convoluted and complex, the principles have not been followed.

References:

- The highest level Data Description Language used is the JSON Schema : <http://json-schema.org/>
- Best Practice has also been taken from the Data Description Language for APIs; JSON API : <http://jsonapi.org/>
- The Interface Description Language used is the Swagger Specification version 2.0 (also known as Open API) : <http://swagger.io/>

#### 2.1.2 ISO 20022

NPP API payloads should be designed based on existing ISO 20022 message elements and components, where available, as *the data standard*. Intended to decrease implementation time for developers to consume, these APIs support interoperability with the asynchronous NPP message formats currently used across the platform.

The principles applied to re-use of ISO message elements and components are:

- Where relevant - the API payloads should be flattened so that they are more developer friendly.
- Only elements that are required for the functioning of the API endpoint should be included in the API payload. API endpoints are defined for specific use-cases (not to be generically extensible for all use-cases). For example - only elements that are required for a single immediate payment initiation would be included in the Payment API payload.
- Support modification of ISO 20022 elements where the existing standard does not cater for an API context (such as filtering, pagination etc.). For example, latitude and longitude in decimal format - as this is how developers will work with latitude and longitude; or using simple types (e.g., a single date-time field) instead of a complex type (e.g., a choice field with a nesting of date and time).
- Consideration of *ISO 20022 and JSON: An Implementation Best Practices*. Includes pertinent modelling guides

### 2.1.3 Security Standards

Each NPP Participant will be responsible for setting the security standards for third party service providers connecting to the NPP Participant via APIs.

NPPA recommends the use of global best practice in security standards where possible.

### 2.1.4 NPP Overlay Service Identification

In order to support in each common API request call, a mechanism to vary the API call attributes according to Overlay Service, the Service Level element (from <SvcLvl/Prtry> element in ISO 20022) is defined as the first element in the API request. The Service Level element is used to distinguish the specific messaging or overlay service under which an API (or XML message) is being used e.g.: npp.msg.01-x2p1.01 (example of an overlay service for basic messaging).

When available, NPPA intends to publish (with the Overlay Service Provider's permission) a list of Overlay Service specific attributes in conjunction with the API framework e.g. Osko Service X2P1 uses the 2x140 characters of unstructured remittance information.

### 2.1.5 Status Codes

Each API may need to consider three status codes that serve different purposes:

- The HTTP Status Code reflects the outcome of the API call (the HTTP operation on the resource).  
E.g. 200 - OK , 400 - Bad Request, 405 - Method Not Allowed
- In the API content, where a status could be returned to reflect the outcome of the request. For example, the Status field in a Payment API payload could reflect the status of a specific payment that makes use of the ISO 20022 PaymentStatusCode code-list enumeration (external code list) to report status.  
E.g. CH11 - Creditor Identifier Incorrect
- Participant specific status codes. Each Participant may have a specific set of status codes that it has defined for API connectivity to its own infrastructure.

### 3 NPP Open API Use Cases

#### 3.1 Payment Initiation

The figure below illustrates an NPP payment initiation process that could be realised as NPP real-time Credit Transfer through the use of a series of common open APIs made available by NPP Participants and associated institutions. APIs are foreseen as complementary to other channels such as asynchronous messaging or web forms / mobile phone apps.

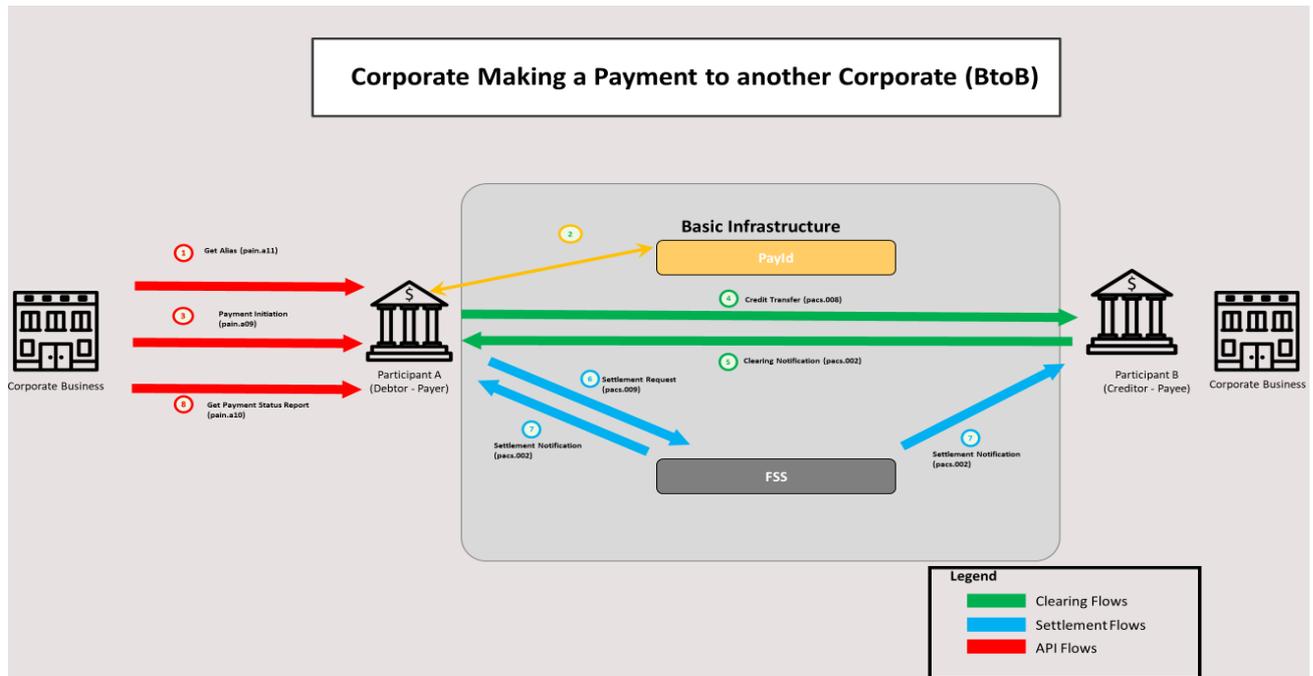


Figure 1. – Payment Initiation Use Case Overview

##### 3.1.1 Payment Flow Processing Steps

The sequence of processing steps in this section is provided as an example of a possible payment flow using APIs; it is a guide only, and is not meant to be prescriptive.

| Step | Description   |
|------|---|
| 1    | Payer business initiates a check for a valid PayID through its NPP bank before submitting a payment initiation  |
| 2    | Payer business submits a payment initiation request to its NPP bank   |
| 3    | Payers NPP bank uses details in initiation message and PayID to create a NPP clearing request to the payees NPP bank.   |
| 4    | Payers NPP bank creates a clearing request with details from the payment initiation request and PayID which is routed via the Basic Infrastructure to the payees NPP Bank |
| 5    | Payee NPP Bank accepts clearing request and responds with a clearing notification   |

| Step | Description  |
|------|--|
| 6-7  | Basic Infrastructure and FSS complete settlement processing and sends confirmations to NPP banks |
| 8    | Payer Bank responds to Payment Status (successful / unsuccessful) to Payer                       |

### 3.1.2 Sample API Documentation

The associated zip file “NPP API Sample Schema and Documents” contains the following for each of the 3 sample APIs:

- **API: pain.a09.001.01 (Post Customer Credit Transfer Initiation)**  
pain.a09.001.01.pdf  
pain.a09.001.01.xls  
pain.a09.001.01.schema.json
- **API: pain.a10.001.01 (Get Payment Status Report)**  
pain.a10.001.01.pdf  
pain.a10.001.01.xls  
pain.a10.001.01.schema.json
- **API: a11.001.01 (Get Alias Account)**  
pain.a11.001.01.pdf  
pain.a11.001.01.xls  
pain.a11.001.01.schema.json

### 3.1.3 Usage Example

The figure below illustrates through the purchase of an airline ticket, the interactions and work flow that might take place:

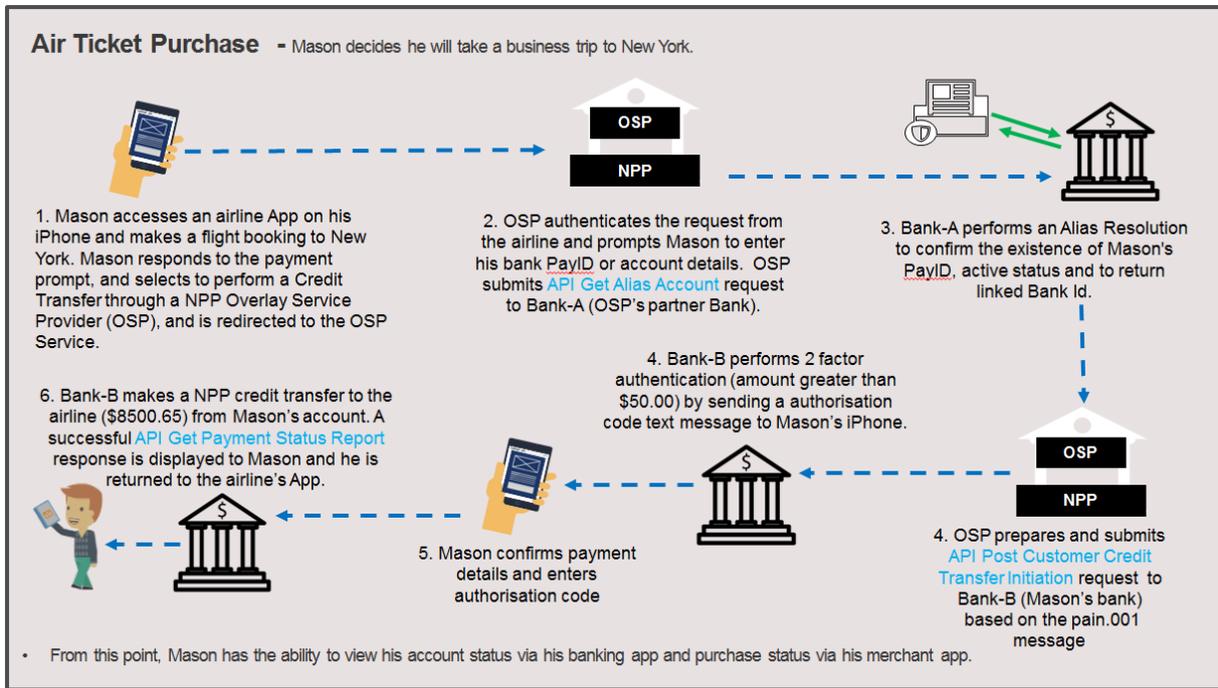


Figure 3. – API Usage Example