PAYMENTS ASSOCIATION OF NAMIBIA

Namibian Open Banking Standards
Version 1.0

March 2025

Contents

1. Introduction

1.1 Document Overview

The Namibian Open Banking Standards come in two parts

- Namibian Open Banking Standards v1.0.docx, (“the standards document”) (this document)
- Namibian Open Banking Standards Data Dictionary v1.0.xlsx (“the data dictionary”).

The Standards Document

The Standards Document is set out as follows:

Project Context chapters

Chapters 1 and 2 (Introduction provides an overview of the document, the context to writing it changes to the document, an explanation of colour coding and the difference between normative and explanatory text, version control and how to give feedback.

Open Banking Context Chapters

Chapter 3 provides an overview of the ecosystem in the form of a “stakeholder map” and an operational overview of Open Banking. Chapters 3, 4 and 5 describe Participant Management, API flows and Ongoing Management, respectively. These chapters provide background and context to the standards.

Standards Chapters

- 7 Definitions and Terminology Standards
- 8 Participant Management Standards,
- 9 API Standards (along with the Data Dictionary)
- 10 Ongoing Management Standards

Annexes

Annexes provide additional reference material and links.

- Chapter 11 provides links to external standards.
- Chapter 12 provides a list of industry abbreviations
- Chapter 13 provides a glossary of industry terms
- Chapter 14 provides account definitions
- Chapter 15 provides suggestions for future changes

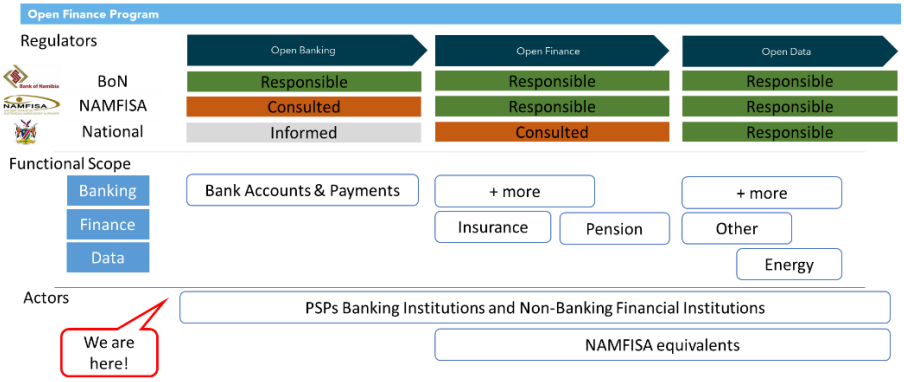
The Data Dictionary

The data dictionary provides formal tables on technical objects, datasets and fields: Elements, API Use Cases, API Endpoints, Consent Objects, Banking Objects, JSON Objects, Params, Fields, HTTP Status Codes, Codes.

1.2 Context for Writing this Document

1.2.1 The Position Paper

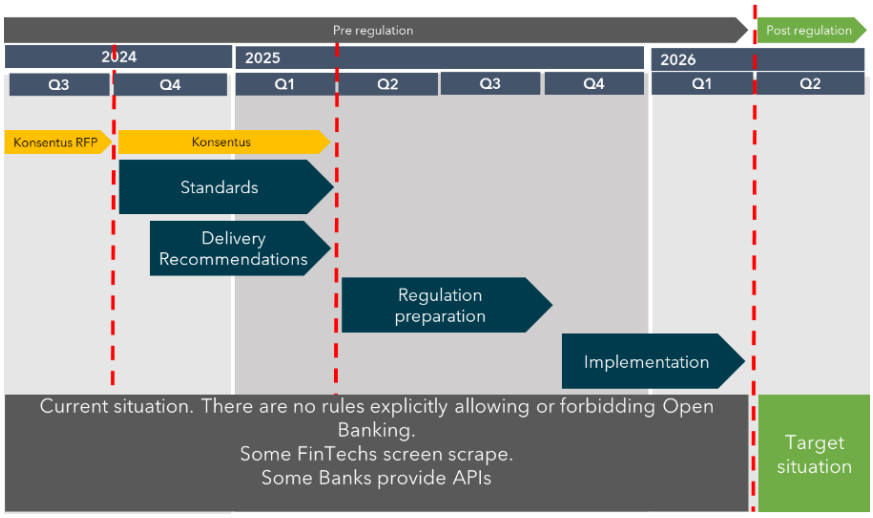
On October 31, 2022, the Bank of Namibia (BoN) issued a Position Paper expressing its intention to implement Open Banking, through a comprehensive legal framework. The Paper also described an approach that would lead from Open Banking to Open Finance to Open Data.



1.2.2 Open Banking Delivery

The Open Banking delivery is happening in a series of stages: “Standards Definition” involves external consultants Konsentus, who are helping the industry write these standards v1.0. In parallel with the standards, Konsentus are making Delivery Recommendations for the Bank of Namibia. The Standards definition phase ends in March 2025.

The timelines after that are not defined, as the recommendations are still ongoing, but based on other projects, there will be phase where the regulation is prepared, and then a phase of implementation, with a live date in 2026.



1.2.3 Standards Phase

This document captures and consolidates the aims and context of an Open Banking program in Namibia, along with requirements and positions for

i. Standards required to achieve Open Banking

ii. Delivery approach for required to operationalise Open Banking.

The document serves as a vehicle to agree on the Standards and delivery approach between stakeholders.

This document is provided by Konsentus to the Namibian industry under the control of the Bank of Namibia. The content within this version is not necessarily endorsed by the Bank of Namibia.

While the standards are initially for Open Banking, it is also intended that they are extendable to other industries.

1.3 Change log

Version	Date	Comments
0.1.	23/10/2024	First version for industry distribution
0.2 0.3	26/11/2024 24/01/2025	Second version for industry distribution Third version for industry distribution
1.0	28.02.25	Final version for industry distribution

1.3.1 Changes Since Version 0.3

Section	Change on Standards v0.3
5.3.1.4	Revisions to Possession and Knowledge definitions
9.2.4.5	Domestic / EFT: amended to NRTC (not NRCT) Cross Border: NISS removed as it's provisioned by CMA
9.4	Table Row on TLS Certificates
10.4	Standards for Change Management text added
10.4.2	Change Management Standards for Scheme Manager updated
Annex 13	Updated with 5th Party definition
Annex 15	Suggestions for Future Changes added

1.4 Normative text, explanatory text and commentary

Within this document, there are different types of text. Each type Is highlighted by colour codes.

Normative text on rules and standards are contained in orange text boxes and tables. They MUST be followed to achieve compliance.
Normative text follows strict change control and uses precise language. Together they are known as "the arrangements" and are binding.

Descriptive or explanatory text is contained in blue text. These may bring out points of attention or examples to clarify the intent.
Descriptive text is NOT binding.

1.5 Document Versioning

The standards are versioned using three version parts <major>.<minor>.<bug fix>.

This version will be used to describe updates in the Change Log.

Each of the three components will be independently incrementing integers and are described as follows:

- **major**: Major version of the Standards. Reserved for increment only when a set of changes are applied that are large enough to make co-existence in the same implementation environment with previous versions untenable. This would include major changes to the information security profile, major changes to the high-level Standards or a change in basic protocols.
- **minor**: Significant changes to the Standards. This would include changes that require approval by the owner of the standards, e.g. the Scheme or a Scheme data Standards Body. Such changes include new endpoints or new versions of existing endpoints.
- **bug fix**: Minor documentation changes that clarify or correct the standards but do not meaningfully alter the standards.

1.5.1 Versioning: Major Versions

The rules and standards MUST have a two-level versioning strategy.

1. The high-level standards are versioned together, where the rules, standards and processes (along with the adherence agreement) are interrelated and form a single version.
2. Specific API endpoints, forms, annexes or other material **MAY** have an additional version specific to the annex.

2. Principles and Assumptions

2.1 Principles taken from the Open Banking Position Paper

Writing these standards is part of a larger program of work to introduce Open Banking, Open Finance and Open Data to Namibia.

2.1.1 The Open Banking Position Paper

The position paper talks about aims and benefits, enablers, risks

- The consumer must explicitly give consent to share their information
- No sharing of credentials
- The type of data includes, but not limited, to Account Holder transaction data; aggregated data (which refers to sets of averaged data of balance information, Account Holders, or open data sources); customer reference data; and open data (which refers to data that anyone can access such as product information and ATM locations).
- Open Banking (and therefore the standards) should be: Secure. Efficient and lead to Innovation. Competition. Fairness. openness, reusability, interoperability, transparency, safety, and security.

2.2 Principles taken from OBF discussions

Open Banking is complex, and implementation should be thought of in terms of phases, where the first phase is always the hardest.

All new functionality brings additional risk, cost and complexity and so requests to extend the scope should be considered considering their benefits e.g. how many people will use this function, how core are they to the key aims of Open Banking.

During the recent OBF meetings, when discussing the scope of functionality and data the following principles have emerged.

- Simplicity and cost-effectiveness.
- Proportionality.

The standards must balance the desire to define everything and increase the scope to the maximum possible, with the need for simplicity, which will allow a working implementation, with limited risk.

Where functionality is suggested but not included, these suggestions are kept for a future version.

2.3 Principle of Reuse

Some questions have asked for specific details about implementation. Here are three examples. In most cases, the answer is **"Data Providers should do whatever they do today through online or mobile banking channels."** Open Banking APIs are a new channel to provide a service, not ways of creating new functionality. Below are some sample questions and answers following this principle.

Question	Answer
When I give a balance, do the standards define how up-to-date that balance must be?	No. The API should deliver a balance that is the same as the balance that would be seen on the customer portal
When I give a piece of data, what standards affecting the accuracy of that data apply? Address data, for example may not always be accurate.	We presume that the balances, and transaction lists you give today are already accurate. We recognise that an address may no longer be up to date, but if the business doesn't worry about putting it on an account statement, why would you worry about putting it on an API?
When there is a joint account, do I need consent from both parties to make a payment or provide a transaction history?	No! Unless you need two people to consent to look at their transaction history.

2.4 Compliance with Namibian Regulations

Open Banking and these standards must be compliant with all national and sector-specific regulations. During comments, we have had requests to consider National Cyber Security Framework (draft) and the National Data / Information Security Framework (draft).

In creating these standards, we have reviewed most BoN determinations and many pieces of national legislation. We are not aware of anything that breaches what is written in Namibian law.

2.4.1 Draft Legislation That May Become Relevant

2.4.1.1 National Cyber Security Framework (Draft).

Attempts have been made by the Namibian government in the last few years to promulgate a Computer Security and Cybercrime Bill (the Bill). However, the Bill received widespread criticism from various stakeholders, which forced the Ministry of Information and Communication Technology to withdraw the Bill from the public consultation process.

2.4.1.2 Data Protection Regulation (Draft)

While this is a draft we have reviewed it carefully. It is in line with data protection regulations in other jurisdictions and causes no problems if the principles of transparency and account holder consent are followed.

2.5 Use of International Standards and Frameworks

Within the feedback received, there were questions about frameworks:

- Will a concept like Zero Trust Architecture be implemented or considered? Since sensitive data will be processed, we adopt a security framework that eliminates implicit trust and ensures we continuously authorize at every stage of digital interaction
- We also need to vet this against PCI DSS v4 requirements.

- Should include a standard such as OWASP or similar for digital channel developments

The OBF had previously highlighted 7 security principles based on the NIST CRSC Security principles. They are Layered security, Separation of duties, Least Privilege, Zero Trust, Dual Control, and Privacy.

NIST provides 33 security principles, and these principles overlap with other frameworks. Also, many principles will clash with other principles.

Rather than listing out many principles, we suggest that if there is any clause or provision in these standards that is concerning, then they should be raised.

3. Environment & Stakeholder map

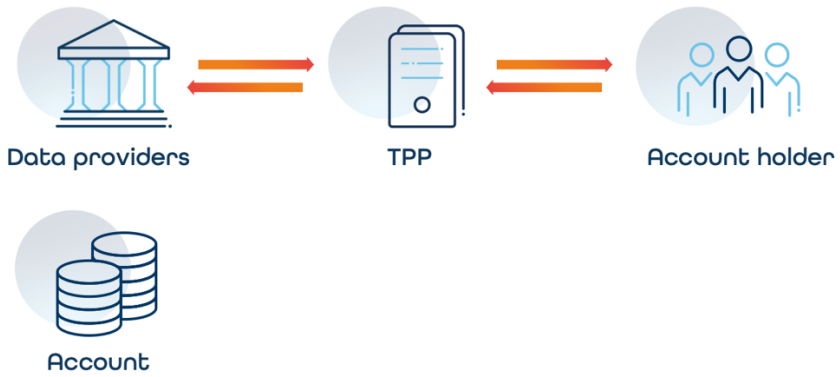
3.1 The Open Banking Ecosystem

3.1.1 Open Data, Open Banking and Accounts

Open Banking is the act of allowing **Account Holders** to instruct their **Payment Service Providers (PSPs) (Banks)** to provide their data to **Third Party Providers (TPPs),** so that the data can be used to benefit the Account Holder who owns the data.

The data or money is stored in an "**account**" held by the Payment Service Provider.

In Open Banking, an **account** refers to a store of value held by a BoN-licensed Payment Service Provider under that license. In this document, the term "account" may include wallets or cards, even if this is not the legal definition. The scope of accounts is defined in the Standards.



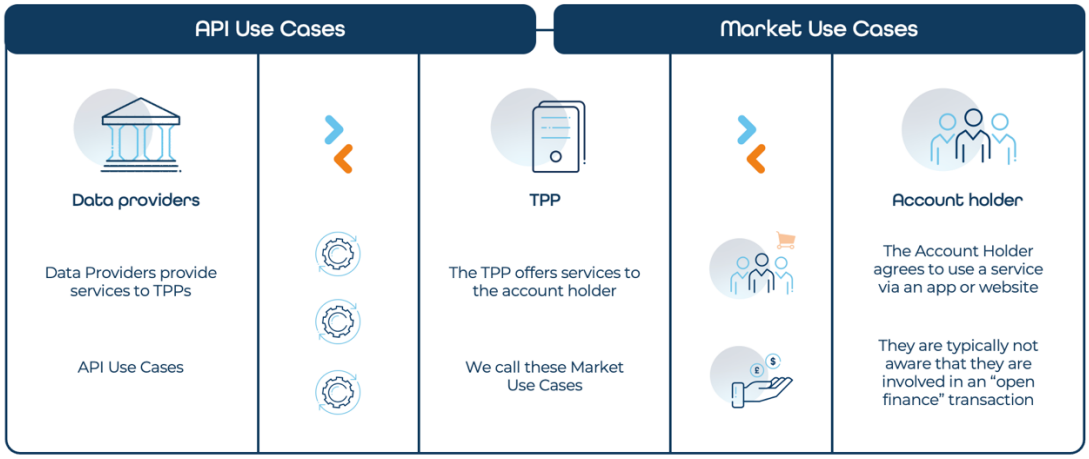
There are questions arising from this discussion, which are brought together in a discussion in a separate chapter at the end of this document (Annex 14: Account Definition).

3.1.2 Use Cases

The functionality provided by Open Banking is known as a "Use Case".

The term "use cases" causes confusion as they can be seen from the perspective of the TPP to the Account Holder (end customer), or the perspective of the Data Provider and TPP. We therefore make a difference between the two:

- **API Use Cases** are functionality provided within Open Banking, generally through APIs, e.g. Get a list of Accounts. Make a Payment.
- **Market Use Cases** are the services that TPPs offer Account Holders. They are built from data provided by API Use Cases, but may combine other services, e.g. credit checking, know your customer checking, accounting services, reconciliation, tax filing.



Regulators can define the API Use Cases that must be provided by the Data Providers but cannot define the purpose or product that is offered to the Account Holder by the TPP, although they can encourage and educate the market.

- Use Cases are further developed in this document: Chapter 9.2. Use Case standards.
- Document: API use cases and Market Use case examples.

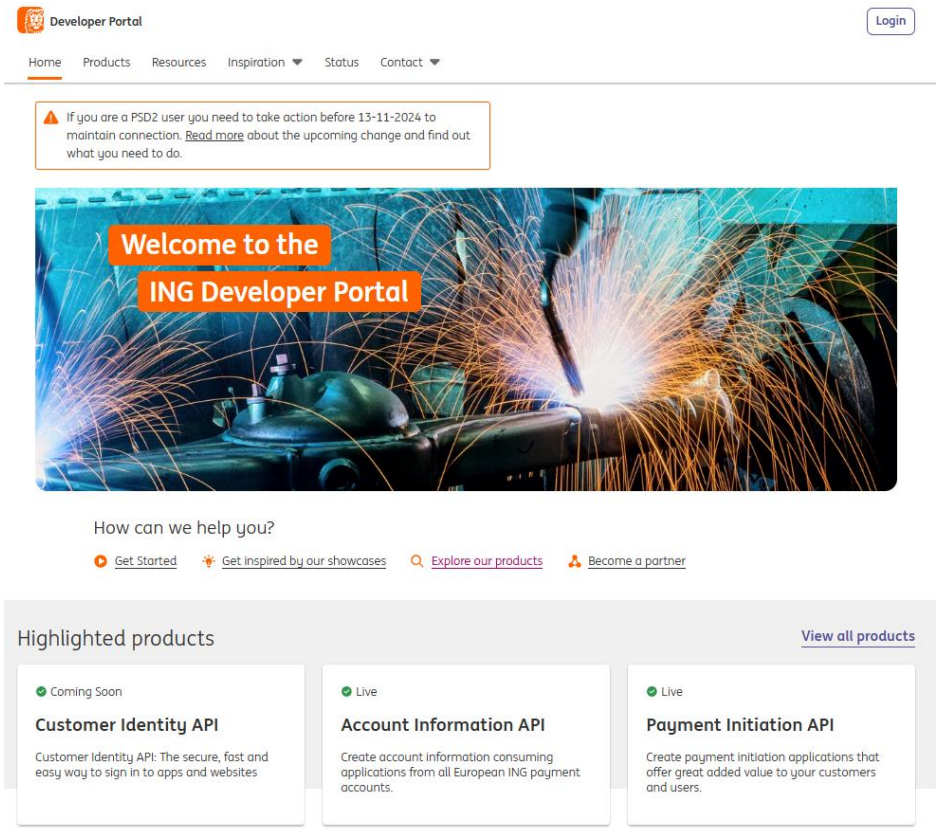
3.1.3 APIs

Data is exchanged using Application Programming Interfaces (APIs). An API is an interface that provides access to data based on agreed upon standards, which may include data formats, security, and consent.

In a general technology context, APIs may be open to the public and free; may be open to the public and paid for; restricted to a class of users or only made available based on a bilateral agreement between two parties.

In an Open Banking context, APIs are generally available to a restricted set of organisations that have an appropriate level of trust. Arguably, they are NOT "Open APIs" from a technology perspective, as they are not open to everybody, although they are more Open than other channels that Financial Institutions (FI's) are used to providing. Therefore, the term "Open APIs" Is best avoided.

APIs and API documentation are published on "Developer Portals". The diagram below shows a sample developer portal from ING In Europe. ING Developer Portal

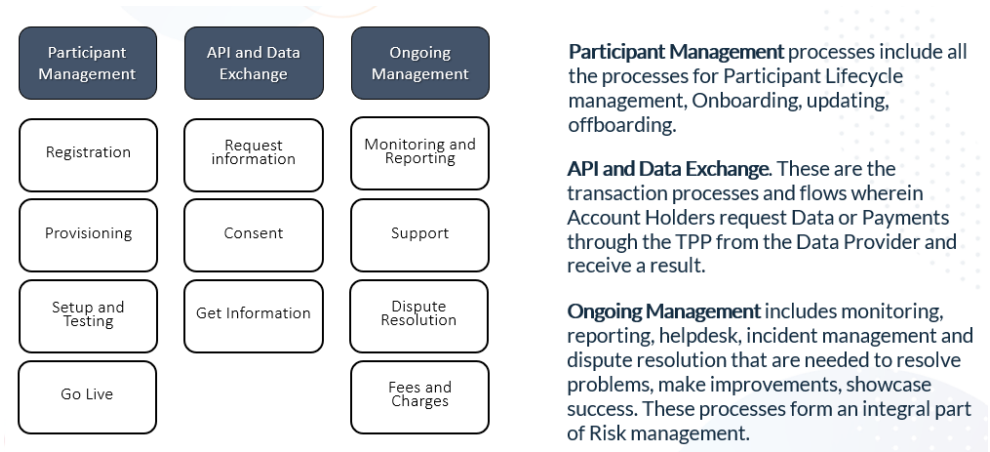


3.1.4 A Trust Framework

3.1.4.1 Trust Framework Pillars

Open Banking, Open Finance and Open Data relies heavily on trust. This trust runs across three pillars

- Trusted Participants: Only trusted players can access data, and they can be trusted to look after that data once they have it.
- Trusted API and Data Exchange: Data is exchanged in a secure manner.
- Trusted Management Processes: If problems occur, they are resolved, if Account Holders have problems they know where to turn, if standards are not being followed measures can be taken.



3.1.4.2 Components of a Trust Framework

Trust is enforced across the three pillars by a combination of rules, standards, processes and tools or infrastructure.

This document focusses on the standards, but if there are API standards it is implied that there are rules, standards, processes and tools or infrastructure, that enforce or support the standards.

Examples of rules, standards, processes and Infrastructure components.

Arrangement	Example
Rules	Participants must follow the standards
Standard	API Standard
Processes	Checking that an API follows the standards Support processes, when the API seems not to work Complaints when a party believes standards are not being followed Change Management processes
Infrastructure	Testing tools, sandboxes, helpdesk infrastructure.

Examples of Open Banking Infrastructure components.

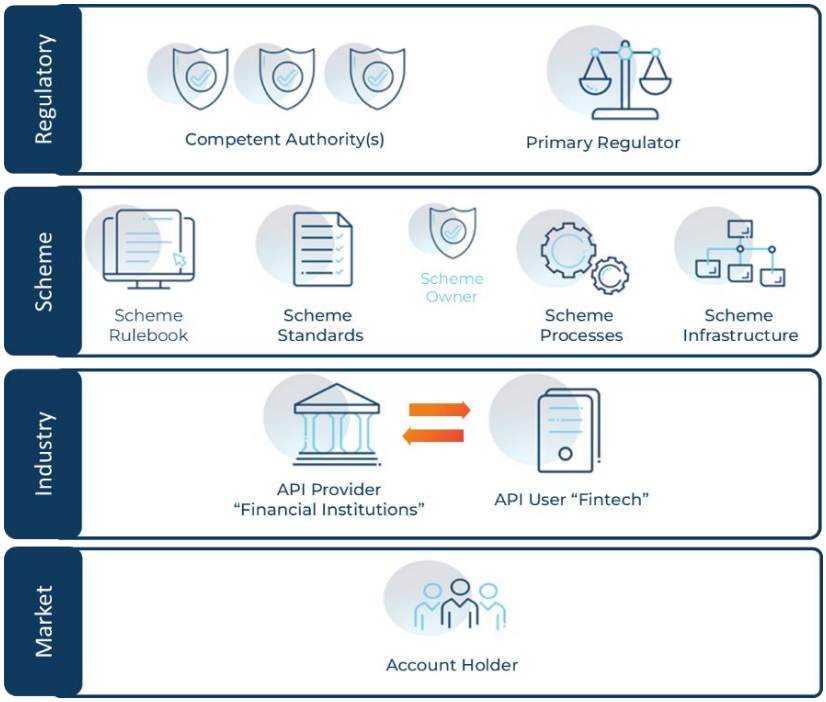
#	Term	Definition	Also known as
1	Developer Portal	A Developer Portal is the interface between a set of APIs and/or other digital tools created by a Data Provider and API User.	
2	API Catalogue	An API Catalogue is a library of APIs created by Data Providers, organised by organisation, subject, purpose, and/or type. TPPs can browse or search API Catalogues to find the APIs in which they are interested.	
3	Environment	An Environment is a collection of processes and programming tools that enables Data Providers to build, test, and debug an API and TPPs to view and use an API.	

4	Sandbox	A Sandbox is a test environment in which new or untested APIs can be viewed and used securely. They are typically created by Data Providers to allow TPPs to safely test the API with their own systems and services without impacting live operations	
5	Conformance System	A Conformance system is a collection of tools and services provided by a Data Provider that allows TPPs to safely test the integration of an API with their own systems and services.	
6	Digital Certificates	Digital Certificates are credentials that can be machine verified by a trusted source. They are the digital equivalent of physical credentials such as passports, and driving licenses	Credentials

3.2 Open Banking Stakeholder Map

This chapter contains the main stakeholders Involved in Open Banking. It Is not specific to Namibia but is a model that works across all ecosystems.

The Open Finance Ecosystem has four levels, Regulatory, Scheme, Industry and Market.



Konsentus commentary:

This model Includes a "Scheme" layer. It is clear within the Open Banking Position Paper that BoN will provide Open Banking regulation and will provide standards. It is to be confirmed where BoN's responsibilities stop and whether other responsibilities are managed by a "Scheme Administrator" (if any).

The standards contained in this document does not discuss the delivery model and who will take on the responsibilities of a scheme but generically covers "The Scheme" and "The Scheme Administrator", which may be BoN or which may be another party.

Comments should focus on the standards themselves and not who implements, maintains and polices them.

3.2.1 Regulatory Actors

3.2.1.1 Primary Regulator

In many jurisdictions Open Banking is mandated, but not always by the same type of regulator. In the UK it was brought in by the Competition and Markets Authority, in Europe by the European Commission, in Australia by the government as part of consumer data standards legislation.

The main activities of the Primary Regulator generally include:

- Defining Open Banking rules for behaviour between market participants
- Defining Admission Criteria for market participants
- Defining Open Banking Standards or delegating their provision
- Providing Open Banking Processes, or delegating their provision
- Providing Central Open Banking Infrastructure, or delegating their provision

3.2.1.2 Competent Authority

A Competent Authority is an organisation that has the capacity to authorise an entity as a TPP or a Data Provider, where such authorisation is required. There may be more than one Competent Authority within a country. A Competent Authority will provide a register or list of authorised entities.

Multiple competent authorities may exist, if the rules allow for participants in one list to be recognised in another list.

3.2.2 Scheme Level Actors

A Scheme is a set of rules, standards, processes and infrastructure that creates a formal legal and operational model between multiple participants. It may be created through national law, scheme rules, or private contracts.

3.2.2.1 Scheme Responsibilities

Where there is no regulation, or where regulation requires enhancement, it is often decided to organise the industry in a structured way to improve efficiency.

Global examples of such schemes in the payment's world are Visa, Mastercard, or SWIFT. A scheme manager compliments or plays the part of the Primary Regulator.

The main activities of the Scheme level actors generally include:

- Defining Open Banking rules for behaviour between market participants, that go further than the legislation.
- Defining Open Banking Standards, that go into more detail than the legislation.
- Providing Open Banking Processes.
- Providing Open Banking Infrastructure.

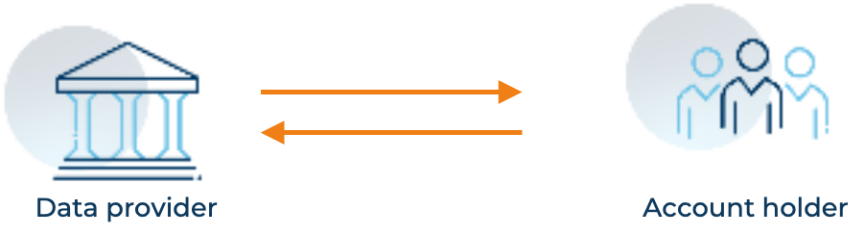
Multiple scheme level providers may exist, e.g. one body that manages standards, another body that runs onboarding processes, another body that provides central testing components.

3.2.3 Industry Level Actors

3.2.3.1 Data Providers and TPPs

The industry roles are that of Data Provider and TPP.

Historically, Account Holders banked with a financial or payment service provider, and there was a bilateral relationship between the two.



In recent years we have seen "third parties" come between the financial Institution and account holder and offer to provide services for the Account Holder based on their data, or ability to make payments.

A Data Provider is an organisation that holds an account for an Account Holder. It allows the customer to instruct a TPP to initiate payments or retrieve data.

A TPP is an organisation that provide services to Account Holders by accessing their data or services that are normally delivered by Data Providers. In the case where the TPP accesses the Data Provider through an outsourced service provider, the TPP is the party that legally captures the account holder consent and is legally responsible to the Account Holder for holding their data.

Depending on the jurisdiction Data Providers and TPPs must be regulated, and/or meeting minimum levels of security, reliability and trustworthiness.

These Standards do not define who can / must / be a Data Provider or TPP in the Namibian Open Banking ecosystem, although Chapter 8 (Participant Management Standards) contains placeholders for rules and standards relating to them.

There are other parties involved in Open Banking at operational level, e.g. service providers, certificate issues, Clearing Houses, as well as 4th Party Aggregators and Beneficiary Parties (see next chapters). In principle all these parties have responsibilities to protect Account Holder information and maintain security, but these obligations come from indirect outsourcing arrangements of the Data Providers and TPPs.

3.2.3.2 Data Providers: Key Responsibilities

These responsibilities are those typically found in Open Banking jurisdictions around the world. The responsibilities may be mandated by legislation, agreed by contract or implied. Konsentus presumes that the BoN legislation will include or capture responsibilities like those below.

Data Providers hold accounts for an Account Holder. In a "Banking" context they will be "Banking" accounts, although wallets and e-money accounts are included in this definition. Data Providers allow the customer to instruct a TPP to initiate payments or retrieve data.

Data providers MUST

- Allow Account Holders to request data or to initiate payments through Participating TPPs
- Provide a developer portal and APIs in compliance with the Standards.
- Grant access and connect to Participant TPPs in accordance with the Participant setup standards.
- Receive requests for data or payments to be initiated from Participating TPPs who communicate securely in compliance with the standards.
- Verify that such requests are made with the consent of the Account Holder.
- Respond by providing data or confirming the initiation of a payment to the Participating TPP in compliance with the standards.
- Follow the Ongoing Management Standards which cover, e.g. reporting, notification, support.

3.2.3.3 TPPs: Key Responsibilities

These responsibilities are those typically found in Open Banking jurisdictions around the world. The responsibilities may be mandated by legislation, agreed by contract or implied.

Konsentus presumes that the BoN legislation will include or capture responsibilities like those below.

TPPs sit between an Account Holder and their “Data Providers”. They perform two activities:

- Offering a service to an Account Holder based on the data or funds held in an account held at another institution.
- Securely obtaining the data, or to initiate payments with the consent of the Account Holder from the Data Provider.

TPPs MUST

- Provide an API Client that uses APIs in compliance with the Standards.
- Connect to Participating Data Users. in accordance with the Participant Management Standards.
- Contract with Account Holders to provide services
- Receive requests for data or payments from Account Holders, with whom they have a contract.
- Send requests for data or payments to Participating Data Providers, communicating securely and in compliance with the standards.
- Receive data or confirmations of payment initiation from the Data Provider in compliance with the standards.
- Forward or process the data or confirmation of payment initiation to or for the Account Holder in accordance with the Account Holder’s instructions.
- At all times Data Providers follow the Ongoing Management processes. See Chapters 6 and 10.

Furthermore

- TPPs MUST safeguard any data received.
- NOT use that data for any purpose other than that requested by the Account Holder.

3.2.3.4 Roles vs Institutions

Data Providers and TPPs are **Roles**. An Institution may be a Data Provider, an FI may also be a TPP. That institution will have one set of obligations as a Data Provider, and different set of obligations as a TPP.

3.2.3.5 4th Party Aggregators

Previously we noted that TPPs perform two actions:

- Offer a service to an Account Holder
- Securely obtain data from the Data Provider.

In many countries, particularly where there are many (hundreds) of FIs offering APIs these two actions are split, with one party offering the service to the Account Holder, and another party obtaining the data.

In this case the TPP is the party that legally captures the Account Holder consent and is legally responsible to the account holder for holding their data. It Is NOT the party that access the APIs.

The party accessing the APIs Is a "fourth party" that Is working on behalf of the TPP. They are generally considered an outsourcing partner and are captured within the regulatory framework, as any other supplier of the TPP, which is obliged to perform due diligence whether it builds or buys its data connections.

Fourth parties or Aggregators can make standard setting slightly more complicated when clear audit trails and credentials of different parties are required. Their existence should not be forgotten as they have an operational role, if not a legal role.

On the other hand, Aggregators can be great enablers, allowing TPPs to get access to bank APIs with working interfaces, and letting the TPPs focus on the proposition to Account Holders.

3.2.4 Market Level Actors (End Users / Account Holders / Beneficiaries)

3.2.4.1 Account Holders

The key Market Level Actor is the Account Holder. They own the account holding the data or money and they give consent for it to be shared.

There are four types of Account Holders that we should consider: Consumers, Small Businesses, Enterprises and Government bodies or other non-profit organizations.

The Account Holder types have different profiles both in terms of the (future) data protection regulation and the way authentication works.

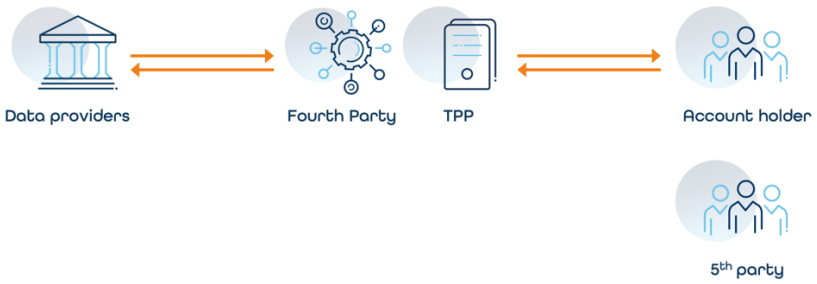
Account Holder Type	Draft Data Protection Regulation	Authentication
Consumers	In Scope	Synchronous / simple
Small Businesses	Not in scope	Synchronous / simple
Enterprises	Not in scope	Asynchronous / complex

Enterprise level Account Holders are excluded as their authentication flows are massively more complicated, already have direct interfaces and they have nothing to do with financial inclusion.

Government sector and charities have been mentioned as a type of Account Holder that are neither individuals nor businesses. From a banking perspective, we assume that their accounts are either similar to consumers, small businesses or enterprises, and so the same logic should be applied.

3.2.4.2 Beneficiary Party

For completeness we note that the Account Holder is not always the recipient of the data or money.



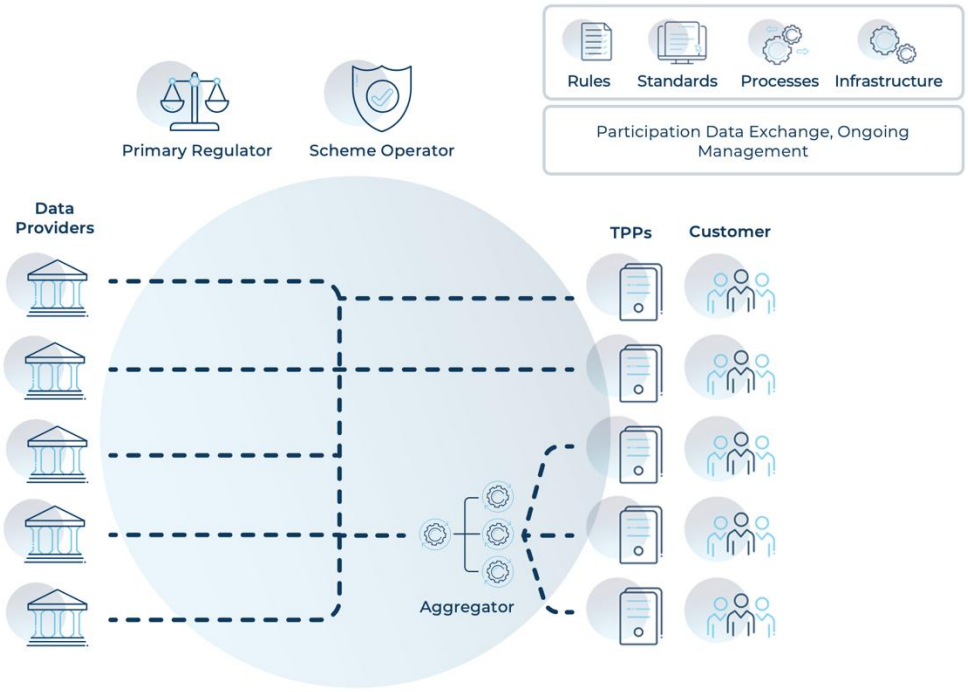
In a payments context, the Account Holder initiates a payment and gives consent, and we call them the debtor/payee/originator of that payment. In most cases, money goes to somebody else, the Payee/Creditor/Beneficiary.

The same may happen in a Data context. The Account Holder makes a request for data and asks that the data (or a summary of the data) is given to another party. This other "5th Party" may be a credit checking service, a tax authority, an on-line shop verifying age before the purchase of alcohol.

The Beneficiary Party is the recipient of the data or money that is/was held by the Account Holder.

3.2.5 An Operational View

A more operational view of Open Banking can be shown as follows. Industry participants (Data Providers and TPPs) exchange data with each other on the Instruction and with the consent of the Account Holders, but under the governance, rules and standards of the scheme and/or regulator.



3.2.6 Access Control: Participants Sectors and Services and Data Categories

3.2.6.1 Access Control Overview

Participants can be split into two roles: Data Provider and Data User.

Many/most jurisdictions also categorise data and functionality to a further level of granularity, so that Regulators, Data Providers and Account Holders can set access control rules and can grant access to data by Sector, by Service and by Type of Operation.

3.2.6.2 Sectors

Sectors define the industry to which the API Is allowed to exchange Data, e.g. transport, banking, insurance, healthcare.

- A TPP may be granted access to Banking and Insurance data, but not Healthcare, with the same Open Data scheme.
- A Data Provider may be required to offer Banking APIs but not Insurance APIs.

This document focuses on Banking Services and Consent Services.

Implementation note:

The API standards incorporate the concepts of Sectors, Services and Categories at least for future proofing reasons, but it should be remembered that if these ideas are to be used when onboarding participants (TPPs) and restricting what they can and cannot do, then this needs to be captured in the onboarding processes.

We are not convinced that there is practical value granting services separately for AIS and PIS at the beginning, but this is an implementation discussion.

3.2.6.3 Services

Services define groups of APIs that are similar. Within Banking five common services are often discussed.

Public data access (PUB). Public data access services refer to data such as branch opening times, cashpoint machine locations, pricing information which are available on websites. Mandating or standardising this is not useful, as the information is already publicly available, although it is often discussed as an easy first step.

Account Information Services (AIS) . Account Information Services return lists or details of bank accounts, account balances, account details, transactions and transaction details.

Payment Initiation Services (PIS). Payment Initiation Services allows the creation and the cancellation of payments, and the ability to get the statuses of payments that have been created.

Subscription Services (SUB). Subscription Services refer to APIs that allow the creation of an account or the application for a loan.

Common Services (Common). Common services are services that potentially span industries or are used administratively between the scheme participants. Consent services are the most important example.

Payment Initiation and Account Information are the common two that are seen in different countries, and in some jurisdictions a TPP could be granted access to AIS services but not to PIS services if the two risk profiles are felt to be different.

Account Opening / Loan application APIs are also less common as the business process between institutions are so different, it is hard to standardise an API.

3.2.6.4 Operation Types

Another way of dividing API functionality and access control is between “read” and “write” APIs.

This term is used a lot in the UK (for example) to mean Payment Initiation which is confusing for non-computer scientists.

As part of future-proofing, it is reasonable to think of access control for data in terms of read and write. The table below provides two examples showing the difference between read and write.

Read	**Write**
Get my address	Change my address
View my card limit	Change my card limit

3.2.6.5 Resource Types

Resources are the objects that can be reached, listed, queried, updated or created. Each resource type will have the same properties, e.g. data elements.

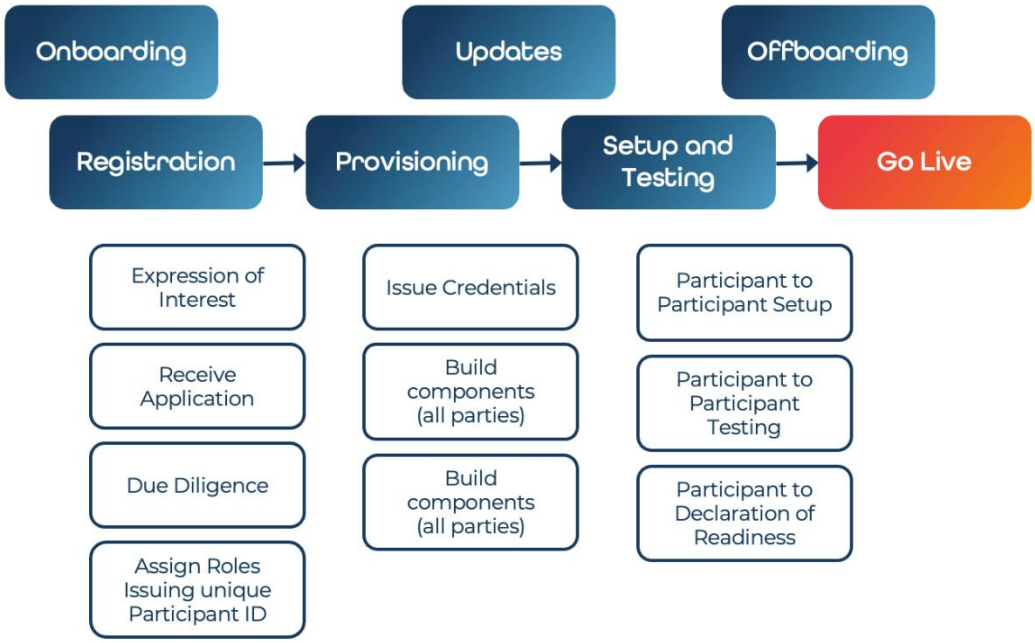
The resource types are: Accounts, Balances, Transactions, Payments, Payment Status.

4. Understanding Participant Management Standards

4.1 Participant Management Overview

"Participants" are organisations that are Data Providers or TPPs or both and so are allowed to exchange data within an Open Banking Scheme. The Participant Management Standards deal with the procedures relating to onboarding Participants, maintaining their access, and offboarding them, following revocation of access or termination of Participation.

Onboarding has three stages, Registration, Provisioning, Setup and Testing



A note on Environments

It is assumed based on common practice in other jurisdictions, that there will be at least two environments: a live environment and a testing environment.

Environments allow the segregation of activities, keeping live data from test data and live functionality from functionality that is under development.

4.2 The Registration Process

4.2.1 Registration Paths

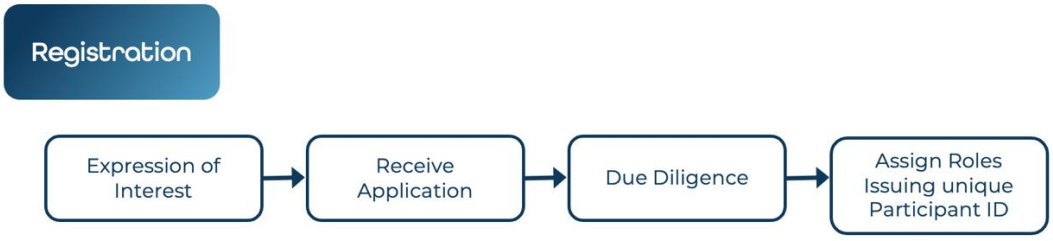
There are two paths to becoming a participant: Mandatory requirement or Voluntary Application. Both paths may apply In Namibia and may apply differently for different roles.

4.2.1.1 Mandatory (Compliance) Path

An organisation becomes a participant because they are compelled to do so through legislation, e.g. From June 30th, 2025, all Namibian Banks are considered Data Providers and must offer APIs in accordance with the standards.

4.2.1.2 Voluntary Application Path

An organisation applies to become a participant because they want to. In this case, there will be a series of registration steps. Based on experience in other jurisdictions, the process for voluntary application will be like the steps below.



Process Step	Process Description
Expression of interest	Organisations that wish to become Participants will express interest and receive information and discuss with the Scheme Administrator.
Receive Application	The Scheme Manager will receive applications to join and take applicants through the Participant onboarding process.
Due Diligence	The Scheme Administrator will assess the application and verify that the applicant meets the required standards, as defined in the rules
Issue a unique ID	The Scheme Administrator will register applicants into a central directory.

4.2.2 Registration Standards

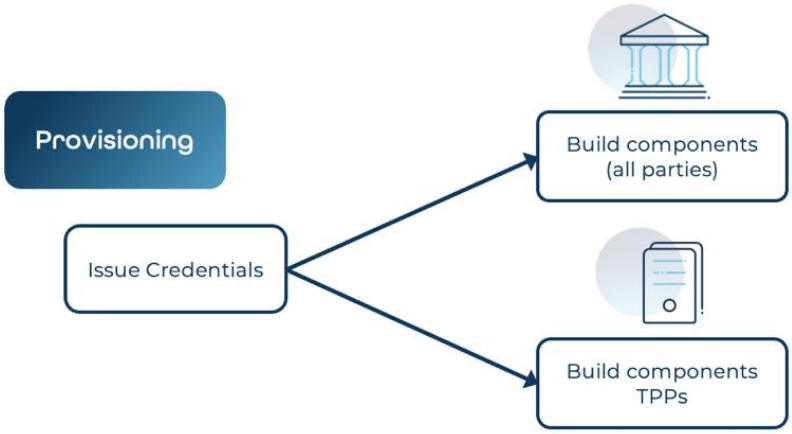
Chapter 8.1 contains registration standards, as follows.

Step	Description
Scheme Administrator Identification Standards	Scheme Administrator Identification Standards are the Name and ID of The Scheme Administrator published in an unambiguous way for use within data transfer flows.
Participant Role	The Participant Role Standards formalise the encoding that will be used for Data Provider and TPP.
Sector Identifiers	Sectors define the industry to which the API Is allowed to exchange Data, e.g. transport, banking, insurance, healthcare.
Service Identifiers	Services define groups of APIs that are similar. Within Banking common services are Payment Initiation, or Account Information.

Participant ID Standard	The Participant ID Standard is the standards for a unique "Participant ID" that will be used with the scheme. This ID Is known as the "Client ID" for technologists.
Participant Admissions Criteria	Participant admissions criteria are the Eligibility and Capability criteria for Providers.
Eligibility Criteria for Data Providers	Eligibility Criteria for Data Providers are the characteristics that a Data Provider must have or be to become a Participant. Eligibility Criteria are checked at the time of registration.
Eligibility Criteria for TPPs	Eligibility Criteria for TPPs are the characteristics that a TPP must have or be to become a Participant. Eligibility Criteria are checked at the time of registration.
Capability Criteria for Data Providers	Capability Criteria assess the ability of a Participant to carry out necessary functions. Capability Criteria may be proven after the initial application e.g. as part of a testing program.
Capability Criteria for TPPs	Capability Criteria assess the ability of a Participant to carry out necessary functions. Capability Criteria may be proven after the initial application e.g. as part of a testing program.

4.3 Provisioning Process

An organisation that has the legal status of participant must also have the technical and operational capability to be a participant and so must be given, build or buy the security credentials and software components that meet the standards.



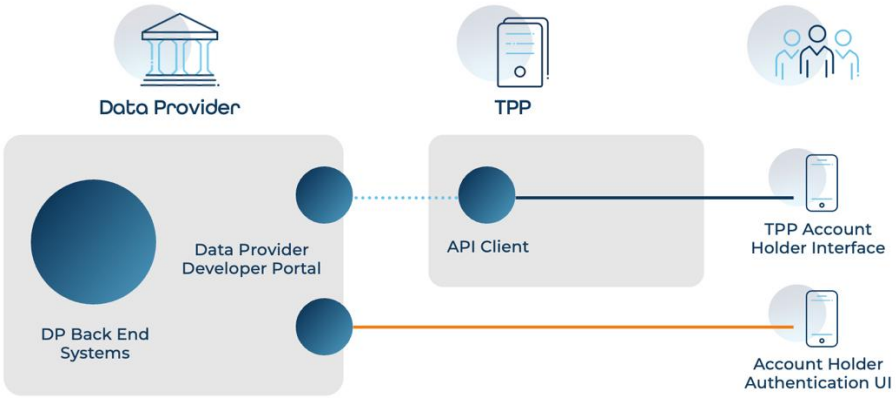
4.3.1 Provisioning Components

Data Provider Participants MUST Provide a Developer Portal and related technical, administrative and legal functions that meet the Developer Portal Standards.

Data Provider Participants MUST Provide an Account Holder Authentication Interface that meets the Standards.

TPPs MUST provide API Clients capable of securely interfacing with the Test and Live Developer portals of the Participant Data Providers.

TPPs MUST provide an Account Holder Interface that complies with the standards.



Provisioning Components	Description
Participant Credentials	The Participant Credentials are digital certificates used for the purpose of identification and securing the API. They may be issued centrally by the scheme owner or issued by Certificate authorities. Participant Credentials must meet trust and technical standards.
DP Customer Authentication UI	The device, app or interface provided by the Data Provider to the Account Holder to authenticate themselves, whether by entering an SMS One time password, a thumb print, a password, or any in other way.
DP Developer Portal	The specific modules that a TPP calls to get data, make a payment or request consent.
API Client	The gateway or interface located with the TPP, that sits between the DP Developer Portal and the TPP's back-end systems.
TPP Account Holder Interface	The TPP branded interface that the Account Holder sees and through which they request the TPP to get data or initiate a payment. For consumers, this is typically an App, but it could also be a website, accounting package, a point-of-sale machine.

4.3.2 Provisioning Standards

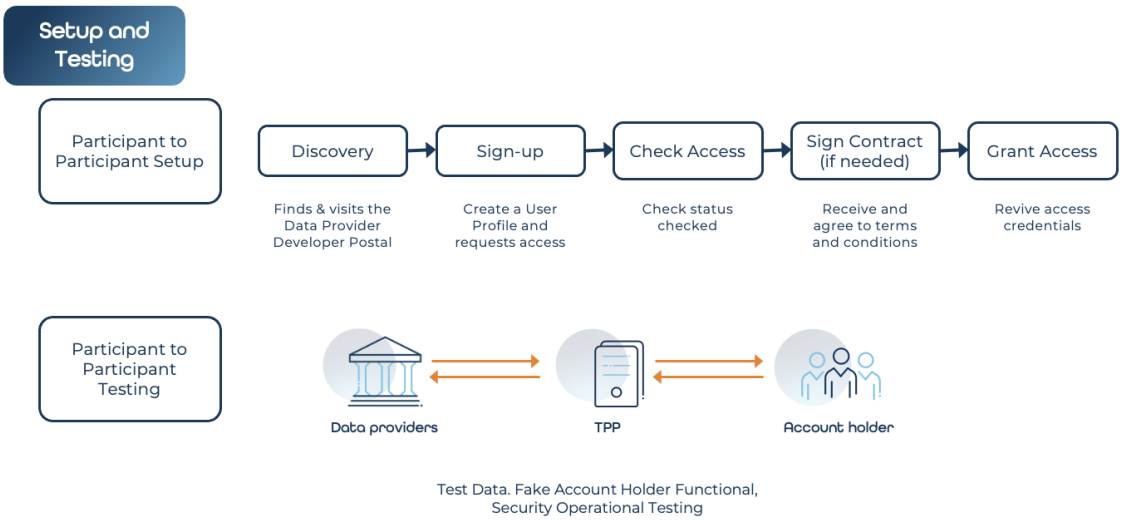
The provisioning standards define the minimum requirements for Participant Credentials and software components.

4.4 Set Up and Testing

4.4.1 Set Up and Testing Process

Set up describes the activity where a TPP connects to APIs of a Data Provider, through a developer portal. This activity includes discovery, registration, and getting ready to send through API calls, either for testing or in a live environment.

Testing describes the testing activity that takes place between the two participants.



4.4.2 Set Up and Testing Standards

The setup standards will depend on the legal environment, notably

i. Whether legal contracts are needed

ii. Whether pricing negotiations can take place.

This is not a topic for the OBF.

These standards will assume that legal contracts and pricing *may* be needed and so include references to them, to be removed later, if required.

The testing standards will depend on the testing approach from BoN, specifically, whether they or a scheme owner will provide

- Central testing tools against a reference site.
- Central testing scripts

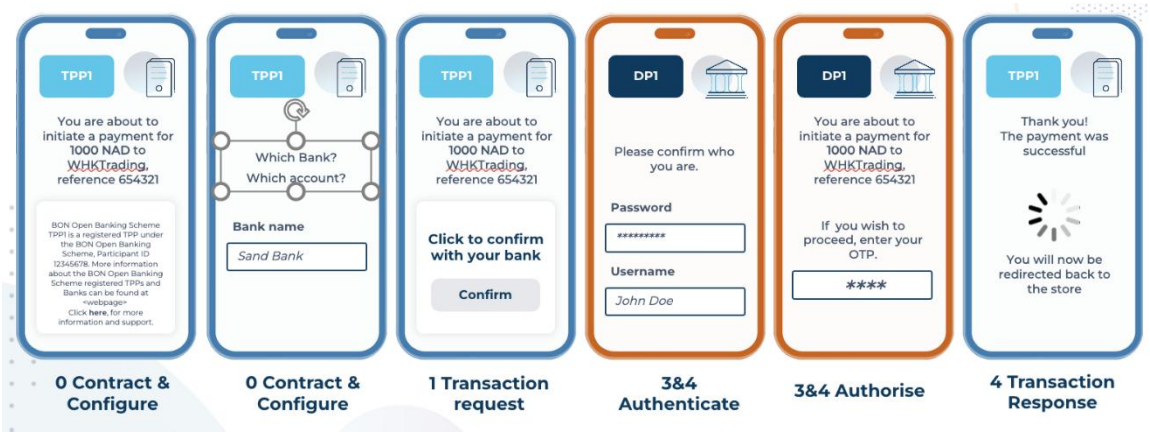
5. Understanding API Standards

5.1 The Customer Journey (Flow)

The customer journey is divided into three parts. Where these steps related to the Open Banking flow, the numbers are included.

- Market Awareness
- Contracting and configuration with the TPP
- Transaction flow
    - Account Holder request to TPP
    - Authentication and Authorisation with the Data Provider
    - Response to the Account Holder from the TPP.

The user experience for the contracting and transaction may look something like this:



5.1.1 Market Awareness

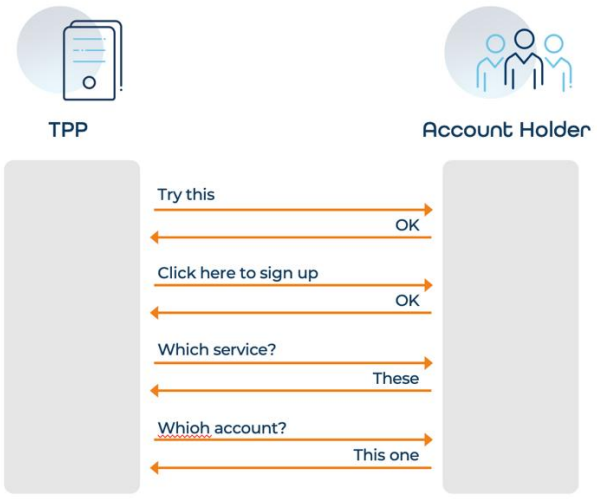
Even before an Account Holder sees a TPP, there should be positive, and consistent, information in the market that describe Open Banking.

See 9.6 UX Standards for mandatory information.

5.1.2 Contracting and Configuration with the TPP

In some cases, Account Holders will actively subscribe to a service provided by TPPs. In some cases, Account Holders may be less aware of the TPP, for example in making a payment for an e-commerce transaction, where they see the Merchant as the main party they are dealing with, and the TPP appears only during the checkout experience.

5.1.2.1 Case 1: Account Holder Subscribes to a Service



The user experience when subscribing to a TPP service will follow the steps below

1. The Account Holder is offered an App or service
2. The Account Holder creates an account and accepts the terms and conditions.
3. The Account Holder registers their Financial Institution and may register their primary Account and or other accounts.
   See UX standards for mandatory information to be displayed.

5.1.2.2 Case 2: Account Holder Subscribes as Part of Another Transaction

The user experience when subscribing to a TPP service will follow the steps below.

1. The Account Holder is in an environment when a payment is needed (e.g. a shopping basket) or bank data is needed (e.g. a confirmation of age).
2. The Account Holder says they wish to pay by bank transfer or provide data from their bank.
3. The Account Holder is taken to a page where they are informed that the payment service will be provided by <TPP Name> and that they must accept the terms and conditions.

See 9.6 UX standards for mandatory information to be displayed.

5.1.3 Transaction Flow (Account Holder perspective

The following flow is from the Account Holder perspective and this chapter provides the operational flow that is happening behind the scenes.

1. Account Holder request to the TPP. The Account Holder will make a request for a balance or a payment to the TPP.
2. Authentication and Authorisation with the Data Provider. The Account Holder receives a challenge from the Data Provider and responds to the challenge.
3. Response to the Account Holder from the TPP. Following a successful call, the Data Provider responds to the TPP who then takes the necessary action towards the Account Holder, whether this is confirming that a payment has been made, showing a balance, or providing a chart.

N.B. there are cases where the 2nd step may be invisible, either because recurring consent has already been granted, or because the Authentication and Authorisation is performed using the mobile handset (see chapter 5.3).

See UX standards for mandatory information to be displayed.

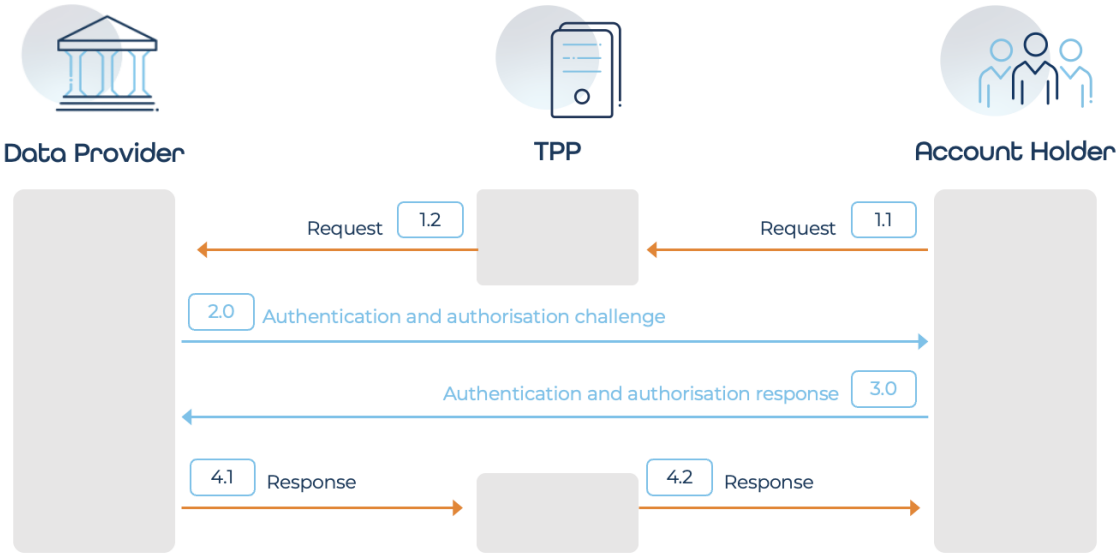
5.2 API Transaction flows

All flows are “logical” i.e. they do not include deeply technical representations of HTTPs handshakes, redirection flows, DNSlookups or other lower-level protocols, but focus on business data being transmitted between different parties.

5.2.1 Business Data Flows

5.2.1.1 Account Information Flow

The flow of information is represented simply by the diagram below. This diagram does not show additional consent flows (see chapter “Consent Flows”), nor the specific infrastructure components used by each party.



The Account Holder wants a balance request

1.1 The Account Holder sends a Balance Request to the TPP.

1.2 The TPP sends a Balance Request to the Data Provider.

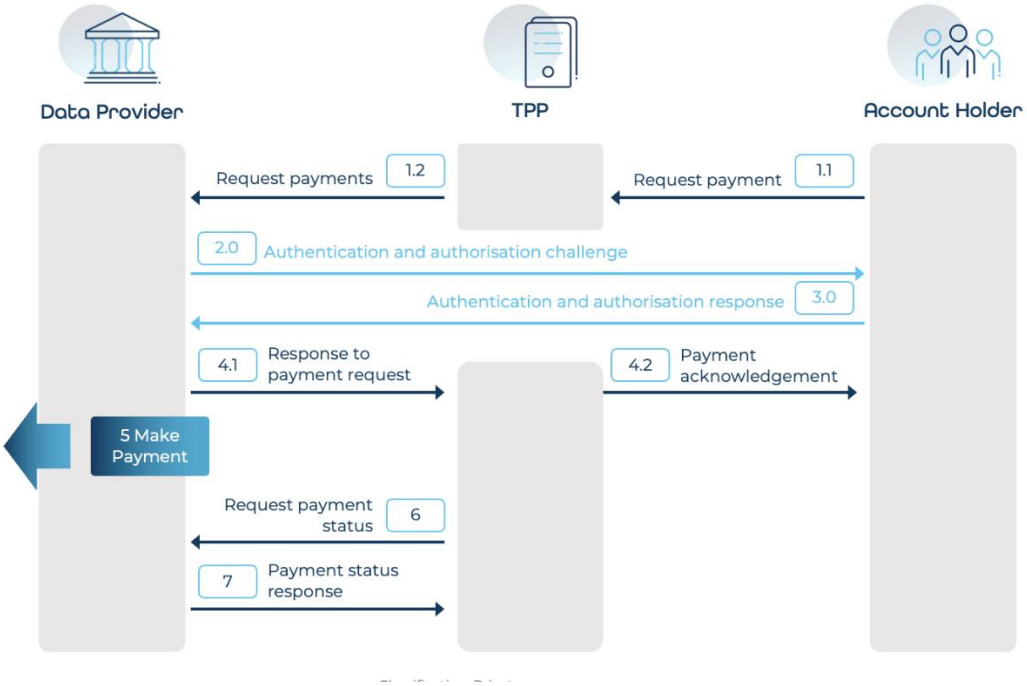
2.0 The Data Provider sends an Authentication and Authorisation Request to the Account Holder.

3.0 The Account Holder sends a confirmation of identity and consent response to the Data Provider. The Data Provider checks the consent.

4.1 The Data Provider sends a Balance Response to the TPP.

4.2 The TPP sends the Balance to the Account Holder.

5.2.1.2 Payment Initiation Flow



1.1 The Account Holders asks the TPP to initiate a payment

1.2 The TPP sends a Payment Initiation Request to the Data Provider.

2.0 The Data Provider sends an Authentication and Authorisation Request to the Account Holder.

3.0 The Account Holder sends a confirmation of identity and consent response to the Data Provider. The Data Provider checks the consent and tries to initiate the payment checking, for example, the account has funds, the Payer Bank exists, the cutoff times allow the payment to be made.

4.1 The Data Provider sends the result of the Payment Initiation request to the TPP in a Payment Initiation response.

4.2 The TPP may send an acknowledgement that the Payment has been initiated to the Account Holder.

5.0 The Data Participant makes the payment, i.e. sends the payment instruction to the Payment System (in a real time environment, this may happen as part of step 3. In a batch-based environment this may happen overnight.

6.0 (Later) The TPP sends a status request for the payment to the Data Provider.

7.0 The Data Provider sends the result of the Payment Status request to the TPP.

N.B. the status of the payment can be requested multiple times.

5.2.1.3 Negative Flows

The negative flows are identical to the positive flow

The TPP initiates an Account Information Request, on the request of the Account Holder, The Data Provider receives the request and after checking, the Data Provider sends a negative Response to the TPP.

The negative response will contain

i) A negative HTTP response code and

ii) An errors response object with appropriate application error codes.

The TPP receives the information and performs whatever action they choose based on the information received, e.g. asking the Account Holder for more information, or informing the Account Holder that the transaction cannot be completed.

5.3 Understanding Consent and Authentication

5.3.1 What is Consent

Before data or payments can be shared. The Account Holder must “consent”.

Consent involves all parties understanding

- “who” (who is asking who to provide data to who)
- “what” (what data is to be provided) and
- “for how long” (for how long this access should be granted).

The User Experience guidelines provide mandatory text to be shown to Account Holders.

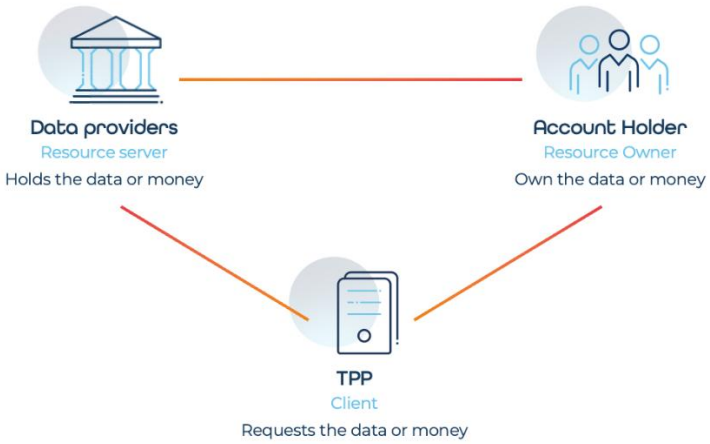
We use the term “consent token” to describe an object that contains six elements:

1. Duration
2. Scope
3. Identity of the Account Holder (Resource Owner)
4. Identity of the Data Provider (Resource Server)
5. Identity of the TPP (OAuth Client)
6. Status

N.B. Technicians will recognise Refresh Tokens and Access tokens, rather than “Consent Tokens”.

5.3.1.1 Technical Consent Flows

Granting consent is a legal concept but is heavily dependent on the technical arrangements. Technical consent mechanisms are well documented and the de facto universal standard in internet security is OAuth 2.0, although other variants exist such as OIDC and FAPI (see below).



Consent in Open Finance generally involves three parties who are not in the same room together, for example.

Term	Definition
Consent	Consent is the freely given agreement of the owner of data (Account Holder / Resource Owner), to the party that holds that data (Data Provider / Resource Server) to access or hold that data or to provide that data to another party (TPP / Client)
Resource Owner	A technical term for the Account Holder, that owns the data or money
Resource Server	A technical term for the Data Provider that holds the data or money for the Account Holder.
Client	The technical term for TPP that requests the data or money from the Resource Holder with the consent of the Account Holder.

Three technical consent standards that are relevant to Open Banking. See Annex 11 for references.

- OAuth 2.0 is a framework for sharing consent without needing all parties to be physically together. There are many ways of implementing OAuth 2.0.
- OIDC is a tighter definition of OAuth 2.0, with more restrictions and better security.
- FAPI (Financial API) is an even tighter definition of OAuth 2.0 and is used in Australia, Brazil and the UK. It is an international standard.

5.3.1.2 Consent Duration

We can consider two types of operation: those which are on- off or immediate, those which are longer term.

- Short-term consent: Consent is given right now, and for one transaction. “Pay this person, now”, “give me my balance, now”, “send a list of last weeks transactions, now”.

- Long-term consent: Consent is given so that a TPP to return again and again to ask for information, without the Account Holder having to go through a complicated authentication process each time. “Every morning message me my balance.” “Every month send my transactions to my accountant.”.

Duration: The duration is the time after which the consent expires, and the Account Holder MUST confirm a consent a second time.

Generally, rules are set making the duration finite with a “maximum consent duration”.

Maximum consent duration: The period after which the Account Holder MUST renew their consent again, with another authentication. This maximum duration is defined in the API Standards Consent Standards but is typically 90 to 180 days.

5.3.1.3 Consent Scope

Scopes define what rights, or permissions the Account Holder is consenting to share, or grant access to, for example:

- I authorise you to give my account numbers to the TPP.
- I authorise you to see the balance for one of my accounts to the TPP.
- I authorise you to see the balance for all my accounts to the TPP.
- I authorise you to give my last two months transactions to the TPP.

Scopes can be very general or very specific. The following list provides examples of scopes that going from general to the most specific.

1. See all information for an account.
2. See account lists and balances but not transactions for an account.
3. See transactions lists but not transactions details for an account.
4. See transaction details but not beneficiary information for an account.

Consent Scopes are defined in the API Standards 5.3.

5.3.1.4 Authentication and Authorisation

For consent to work, the Data Provider must be sure that the person claiming to be the Account Holder is really the Account Holder and not a fraudster. We call this “Authentication”. Authentication is the process of determining whether someone or something is who or what they say they are.

The data provider must also be certain that the Account Holder is giving permission for the Data Provider to give their data to the TPP. We call this “Consenting”, but we also call it “Authorising”.

The Account Holder *consents* for their data to be given to the TPP by the Data Provider.

The Account Holder *authorises* the Data Provider to give data to the TPP.

Sometimes Authentication and Authorisation happen in the same “Challenge”, for example I log on and I see my balances. Sometimes Authentication and Authorisation happen at separate moments, for example I log on and I see my balances, but then I must provide a second code to make a payment.

In most jurisdictions for financial data flows there are standards put in place by regulators about what minimum security levels for an Account Holder Authenticating themselves to a Bank, and Authorising transactions. It is generally not permitted to have a static username and password for “logging in” and then a simple “OK” to initiate a payment.

More secure methods use terms such as SCA, MFA, 2FA (see glossary) but all these terms mean responding to more than one challenge with more than one type (factor) of identification. The three types of factors are generally defined as

i) Knowledge. Something you know (e.g. a password)
ii) Possession. Something you have (e.g. a phone, key card)
iii) Inherence. Something you are (e.g. biometric fingerprint)

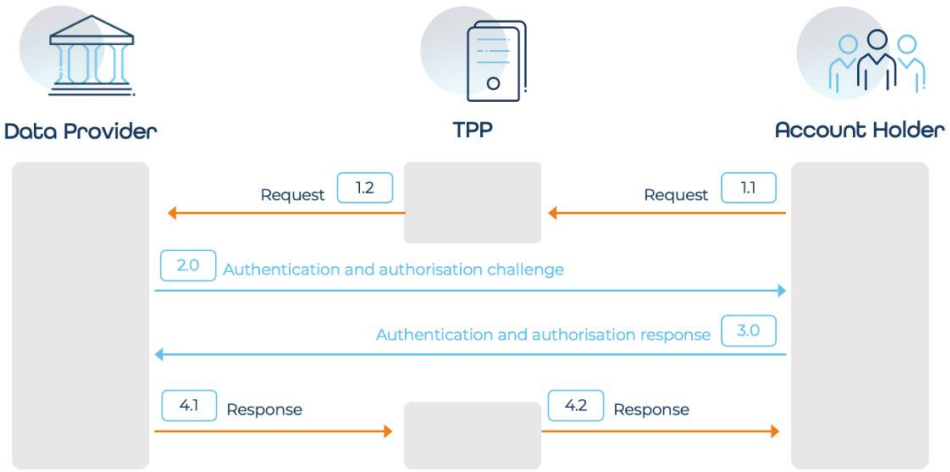
It is possible to prove two things at once, for example if my phone is tied to my account and my bank is app tied to my phone, then logging in to my bank app with a fingerprint proves who I am, who I am biometrically and through possession.

5.3.2 Consent Flows

Three representations of the same flow are shown below, one is “simplified” one is more complex”. They are both the same flow but are shown differently as different audiences want to understand the situation at different levels of detail.

5.3.2.1 The Basic Consent Flow (Simple)

This chapter describes the simple flow, that focuses on the User experience and flows between parties.



2.0 and 3.0 show a simple “Challenge” and response allowing the Data Provider to authenticate the Account Holder and the Account Holder to Authorise the transaction.

5.3.2.2 Granting Consent (More Complex)

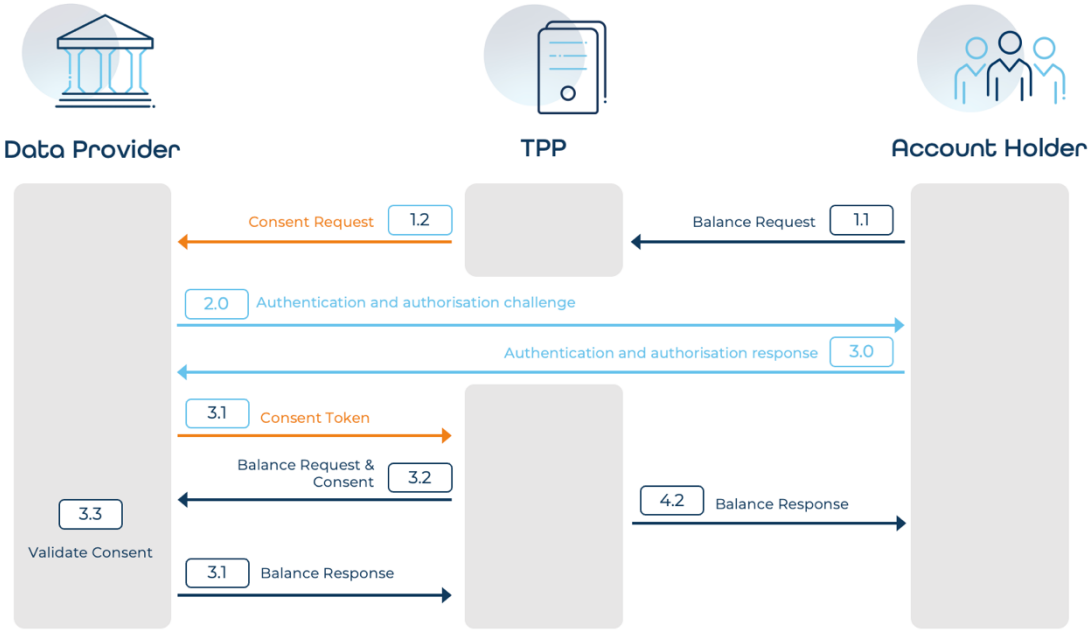
This description is a more accurate description what happens, although it still hides some of the complexity.

Behind the scenes, we see that the TPP is not directly sending the Account Holder request to the Data Provider but is sending a request for authorisation for a balance request.

The Authentication and Authorisation challenge and response happens successfully.

The TPP is given a “Consent Token” that contains the identity of the Account Holder, and the “Scopes” of what that Account Holder can access and the duration of the token.

The TPP can then send the Balance Request, along with the Consent Token.



The Account Holder sends a Balance Request to the Data Provider, through the TPP.

1.1 The Account Holder sends a Balance Request to the TPP.

1.2 The TPP sends a Create Authorisation Request for a balance inquiry to the Data Provider.

2.0 The Data Provider sends an Authentication and Authorisation Request to the Account Holder.

3.0 The Account Holder sends a confirmation of identity and consent response to the Data Provider.

On getting a positive response

3.1 The Data Provider sends a Consent Token response to the TPP.

3.2 The TPP sends a balance Request with the Consent Token to the Data Provider.

3.3 The Data Provider Validates the Consent Token.

The Data Provider sends a Balance Response to the Account Holder.

4.1 The Data Provider sends a Balance Response to the TPP.

4.2 The TPP sends a Balance Response to the Account Holder.

5.3.2.3 Revoking Consent

There are three cases where consent can be revoked:

- Expiration of consent
- Account Holder revokes consent by instructing the TPP
- Account Holder revokes consent by instructing the Data Provider.

These three cases result in three different flows, which may have different data requirements for APIs or operational procedures.

Case 1: Expiration of consent

All consents will be timebound, i.e. Account Holders will grant consent for a certain duration of time. This could be very short (e.g. 20 seconds) for a one-off consent, or could be 4 days, or 4 months.

On the date/time of the expiration of consent, the consent tokens will no longer be valid. The Data Provider, the TPP and the Account Holder will all be aware of the expiration.

Case 2: Account Holder revokes consent by instructing the TPP

This chapter describes the revocation flow when the Account Holder informs the TPP that they no longer require the service and request that consent be revoked.

Start point. Consent has been granted as described in “Granting Consent”. All requests based on that consent token will succeed.

- The Account Holder tells TPP they no longer consent.
- The TPP destroys token (marks it as inactive) and sends a notification to the Data Provider with a Consent Revocation
- The Data Provider updates their consent data and acknowledges the request.

Case 3: Account Holder revokes consent by instructing the Data Provider

Start point. Consent has been granted as described in “Granting Consent”. All requests based on that consent token will succeed.

- The Account Holder tells the Data Provider they no longer consent to giving that TPP, that information.
- Account Holder marks the token as inactive.

At this point, the Account Holder and the Data Provider know that consent has been withdrawn, but the TPP is not aware and still has a consent token that they believe is valid.

- If the TPP requests a balance from the Data Provider for that Account Holder, the Data Provider will validate the token, realise that consent has been withdrawn and send back a failure code.

5.4 Types of API Standard

There are seven type of API standard described.

#	API Standard	Description
1	API Architecture Standards	API Architecture Standards describe the standards that apply to all APIs, not to specific endpoints, user experience guidelines, technical language standards, naming standards, publishing standards, consent standards.
2	API Use Case	The Use Cases and Endpoints standards define the functionality that will be contained in each API.
3	API Data Standards	API Data standards provide the data dictionary and definition for each endpoint and so include the mandatory and optional Response Headers, Parameters, Scopes, Request Datasets, Response Datasets, Meta, Links and Error objects, HTTP Response codes, Data Types and Complex Types
4	API Security Standards	Security Standards describe security between Participants.
5	Consent Standards	Consent standards describe how Account Holders give consent, revoke consent and how the consent is used between the parties.
6	API UX Standards	User Experience Standards includes rules on brands, signposting, text and flow that help Account Holders understand that they are in a working and trusted environment.
7	API Service Level Standards	Service Level Standards describe minimum requirements for processing, cut off time, availability and other non-functional requirements.

5.5 Understanding User Experience Standards

5.5.1 The Importance of User Experience: Trust

For Account Holders to use Open Finance services they must feel comfortable. Traditionally, banks have given the messages that you should not share personal credentials and should not “click” on external links. There are legitimate concerns about identity theft, fraud, hacking or loss of money.

An Account Holder does not know that a request to enter credentials from a TPP is part of a particular legal or national framework, unless they are told.

It is hard to be very prescriptive about User Experience as:

1. There are multiple market use cases.
2. There are multiple channels (web browser, mobile app, point of sale terminal, petrol station kiosk, etc.).
3. Each Data Provider has their own consent flows.

User Experience Guidelines provide minimum requirements and help the consumer receive a consistent experience and not abandon a transaction due to unexpected behaviour.

While the whole user journey may not be standardised, in some countries the consent flow is standardised, with regulatory standards in place and sometimes approved mechanisms.

5.5.2 UX: Branding

In the payments world, we are very used to seeing brands that are run by private schemes. National payments schemes and national open finance schemes do not generally have customer facing logos attached to them, e.g. Australia, Europe, UK although they may have mandatory text.

We do not propose that Namibian Open Banking provides a brand, but we do propose that consistent text and standards be used by all parties, to provide Account Holders with confidence in using relevant services.

See Standards 9.6.1.

5.5.3 UX: Public Information

Even before an Account Holder sees a TPP, there should be positive, and consistent, information in the market that describe Open Banking.

The scheme manager will provide a Webpage for Account Holders, so that all Account Holders can see a single centralised description of the Open Banking Services. This will include:

- A description of the mission and purpose of Namibian Open Banking
- A public list of participating TPPs
- Instructions for Account Holders about what they should do if they have problems.

TPPs and Data providers will provide a Webpage for Account Holders, so that all Account Holders can see a single centralised description of the Open Banking Services. This will include:

- A description of the mission and purpose of Namibian Open Banking
- A link to the Scheme Manager Open Banking Webpage.
- A statement that the TPP or Data Provider is a participating organisation.
- Instructions for Account Holders about what they should do if they have problems.

See Standards 9.6.2.

5.5.4 UX Mandatory Text

Mandatory Text describes what the Account Holder must see at different stages of their journey as displayed both by the TPP and the Data provider. See Standards 9.6.3.

6. Understanding Ongoing Management Standards

6.1 Ongoing Management Overview

After transactions and data have been exchanged there are a number of ongoing Management Processes that may/can/must periodically occur within the Scheme. These include notifications and reporting, support functions (helpdesk, dispute resolution, managing change. There may also be processes for the settlement of fees and charges.

These Ongoing Management Processes form an integral part of risk management and Trust and may be supported by standards, which are described in Chapter 10.

6.2 Ongoing Management Processes

Name	Description
Notifications and Reporting	Participants may be required to provide reports and notifications to the Scheme Administrator, whether around volumes of transactions, success and failure, incidents or usage. The Scheme Administrator may provide reports and notifications to Participants, internal management and, if needed, regulatory bodies. These reports or notifications may be operational (e.g. fees and charges, new Participants, incidents), or informational (e.g. transaction volumes).
Monitoring, Helpdesk and Support Processes	The Participants should monitor transactions, services levels, audit logs, Participant and market feedback to verify that the Scheme is performing well, and to act if problems arise. The Scheme administrator should (based on reports or other information) monitor transactions, complaints, incidents to verify that the Scheme is operating smoothly and to intervene when there appear to be systematic problems.
Dispute resolution	Participants must answer to their Account Holders and provide information about problems or complaints. As an Account Holder is a customer of both the Data Provider and the TPP, there may be a “triangular” flow of information, it may not always be clear where the error resides, and the customer may not always be clear who is responsible for what. There should be coordination

	to understand that Account Holders get answers in a timely manner. The Scheme Administrator may organise processes to help Participants resolve problems with each other, to ensure that Account Holders receive answers, and that repetitive problems can be tackled.
Managing Changes to Standards or other scheme components.	There will inevitably need to be changes applied to the Scheme and/or any of its components. The Scheme is a complicated ecosystem with Data Providers and TPPs who will not always agree on what should be changed, and what is the priority. Managing change involves identifying, socialising change requests and may require review and feedback. Time may be needed for Participants to adapt, and there are issues around versioning to be considered.

7. Definitions & Terminology Standards

7.1 General Open Data Definitions

#	Term	Definition
1	Open Banking	Open Banking is the act of allowing Account Holders to instruct their Payment Service providers (Banks) to securely provide their financial data to other FI third party providers, so that the data can be used to benefit the Account Holder who owns the data.
2	Open Finance	Open Finance is an extension of Open Banking allowing Account Holders to instruct their Financial Service providers (Banks, Insurance companies, Pension providers etc) to securely provide their data to trusted third party providers, so that the data can be used to benefit the Account Holder who owns the data.
3	Open Data	Open Data is the act of allowing Account Holders to instruct any institution that holds data on their behalf (telcos, health care providers, financial service providers, energy providers etc) to provide their data to third party providers, so that the data can be used to benefit the Account Holder who owns the data.
4	Use Case	A Use Case describes functionality provided within Open Banking or Open Finance or Open Data. The term “Use Cases” causes confusion as they can be seen from the regulatory technical, business or consent perspective. We split them into API Use Case and Market Use case.
5	API Use Case	API Use Cases are functionality provided within Open Banking, generally through APIs, e.g. Get a list of Accounts. Make a Payment.
6	Market Use Case	Market Use Cases are the services that TPPs offer consumers. They are built from data provided by API Use Cases, but may combine other services, e.g. credit checking, know your customer checking, accounting services, reconciliation, tax filing.
7	API	An API is an interface that provides access to data based on agreed upon standards. It may be open to the public and free, it may be open to the public and paid for, it may be restricted to a class of users, it may be agreed bilaterally between two parties.
8	Open API	An open API is a free, publicly accessible interface that allows access to data based on agreed upon standards.
9	Account (Open Data definition)	In Open Data, an account refers to a digital construct that allows an individual or organization to store, manage, and access information or money securely.

		Such an account is typically protected by authentication mechanisms, ensuring that only authorized users can access the data. Examples include email accounts, social media accounts, patient portal accounts in health care, and utility accounts in energy and utilities. Accounts are provided by Data Providers.
10	Account (Open Banking definition)	In Open Banking, an account refers to a store of value held by a regulated Payment Service Provider. In this document, the term "account" may include wallets or cards, even if this is not the legal definition. The scope of accounts is defined in the Standards.

7.2 Actors and Roles

7.2.1 Regulatory

#	Term	Definition	Also known as
1	Primary Regulator	The Primary Regulator is an entity that legislates or compels market participants to follow a scheme rules.	
2	Competent Authority	A Competent Authority is an organisation that has the capacity to authorise an entity as an TPP or a Data Provider, where such authorisation is required.	

7.2.2 Scheme

#	Term	Definition	Also known as
1	Open Banking Scheme Administrator	The Open Banking Administrator is the organisation that creates the Scheme rules, standards, processes and infrastructure and requirements that defines an Open Finance Ecosystem. This may be a national regulator, delegated by a national regulator, or market led.	

2	Open Banking Scheme Service Provider	The Open Banking Scheme Service Provider(s) runs processes and infrastructure that make up the Open Banking scheme. There may be multiple providers with different tasks or providing different components, e.g. certification, handling administration, providing infrastructure.	
3	Open Banking Specific Infrastructure Providers	An Open Banking Scheme specific infrastructure provider provides software that is specifically designed to support Open Banking, whether a full ecosystem solution or whether specific elements, such as dispute resolution software, authentication modules or testing components.	
4	Common Infrastructure Provider	These documents do not mention internet browser providers, internet service providers, telecom companies, cloud service providers, electricity providers, but all are essential to Open Banking. They are assumed to exist.	
5	Trust Service Provider	A Trust Service Provider is an entity that issues digital certificates.	Certificate Authority

7.2.3 Industry

#	Term	Definition	Also known as
1	Financial Institution	Financial Institution is a generic term applied to Banks, Credit Unions, Building Societies, E-money Institutions, and Payment Institutions, Insurance companies, Pension funds or any other regulated organisation that offers financial products to citizens	Bank, Credit Institution, Electronic Money Institution, PSP, NBFI.
2	Fintech	A Fintech is a Financial Technology company, generally assumed to be technically cutting edge, agile, reactive, and ready to offer services in their role as a TPP. They may or may not be regulated.	
3	Participants	Participants are organisations that are allowed to exchange data within an Open Banking Framework (i.e. a scheme, or environment).	

4	Non-Participants	Non-Participants are organisations that are not allowed to exchange data within an Open Banking Framework (i.e. a scheme, or environment), either because they do not meet access criteria, or they are in the process of joining.	
5	Data Provider	A Data Provider is an organisation that provides an Account for an Account Holder. It allows the Account Holder to instruct a TPP to initiate payments or retrieve data. In Namibian Open Banking, the Data Provider will be a Payment Service Provider licensed by BoN and providing accounts under that license.	Data Provider, Account Service Provider (ASP)
6	Third-Party Provider (TPP)	A TPP is an organisation that provide services to Account Holders allowing them to access their data or services that are normally delivered by Data Providers. In the case where the TPP accesses the Data Provider through an outsourced service provider, the TPP is the party that legally captures the account holder consent and is legally responsible to the account holder for holding their data.	TPP, Data User, Client.
6.1	Account Information Service Provider (AISP)	A TPP that provides Account Information Services.	
6.2	Payment Initiation Service Provider (PISP)	A TPP that provides Payment Initiation Services.	
7	Service providers	A Service Provider is a technology company that provides components of the Open Banking ecosystem. This is typically technology-based and may include API Hub or Aggregator services.	
7.1	Financial systems integrators	Data Providers will need and have accounting systems, payments systems, consent management systems, gateways to financial networks, firewalls, security systems. These systems are on the edge of the Open Finance journey and are often provided by a few known organisations.	

7.2	Aggregator	An Aggregator is an outsourcing partner or a service provider to a TPP. They connect to multiple Data Providers and offer a single API or connection to the TPP.	
7.3	Gateway Provider	Gateway providers supply gateways, developer portals, or other interfaces to Data Providers.	

7.2.4 Market

#	Term	Definition	Also known as
1	Account Holder	An Account Holder is a natural or legal person who is accessing Open Banking services through a TPP. They hold the account that is being exposed by the Data Provider and accessed by the TPP. In OAuth flows they are the "Resource Owner". There are four types of Account Holders that are considered, Consumers, Small Businesses, Enterprises Government bodies or other non-profit organizations.	Customer, Consumer, PSU, Resource Owner
2	Beneficiary Party	The Beneficiary Party is the recipient of the data or money that is/was held by the Account Holder.	5th Party, Payee, Creditor, Beneficiary.

7.3 Consent Definitions

Consent for data is not generally discussed except by regulators, lawyers or developers, and they have different sets of terminology.

#	Term	Definition	Also known as
1	Consent	Consent is the freely given indication of the owner of data (Resource Owner), to allow another party to access or hold that data or to provide that data to another party.	Authorisation
2	Authorisation Authorise	When talking about consent, the word authorise, is more natural way of saying "give my consent to you to ...", for example the following two sentences are the same:	

		I authorise you to give my data to a TPP. I give my consent to you to give my data to a TPP.	
3	Consent Token	The consent token is the technical embodiment of consent. The consent token contains a duration, a scope, the identity of the Account Holder (Resource Owner), the identity of the Data Provider (Resource Server), and the identity of the TPP.	
4	Authentication/ Authentication	When talking about consent Authentication means prove who you are and in doing so, that agree to something. Authentication involves receiving a challenge and responding to it correctly.	
5	Multi-factor authentication	Multi-factor authentication means responding to more than one challenge, e.g. not just a password, but a One Time Passcode (OTP). Often these challenges will mix different authentication methods, and require that one or more is "dynamic", i.e. it changes each time.	SCA, MFA, 2FA
6	Short-term consent	Consent is given right now, and for one transaction.	
7	Long-term consent	Consent is given so that a TPP to return again and again to ask for information, without the Account Holder having to go through a complicated authentication process each time.	
8	Maximum consent duration	The period after which the Account Holder MUST renew their consent again, with another authentication.	

7.4 Technical Components

7.4.1 Components Provided by the Scheme Manager

The following items *may* be provided by, or under the responsibility of, the Scheme Manager.

#	Item	Description
1	Central Directory	A central directory that holds a record of Participants that have joined Scheme along with the unique number that identifies them, the roles they hold, the APIs that they are authorised to use, and their status as Participants. The Directory also holds technical information about the Participant.

2	Certificate Authority	Provides credentials (certificates) to Participants to enable secure communications. The module will be able to Issue Trusted Certificates, Revoke Certificates, and provide revocation lists.
3	Administrative Tools	Administrative tools are technical components that are not normally thought of as infrastructure but still need to be selected, implemented and maintained. They are widely available commercially and are specific to Open Finance. They include a website, document portals, and helpdesk tools.
4	Reference Site	The reference site shows the API Standards as they would be displayed by Data Providers. Includes tools to allow "functional testing" of data elements to that TPPs can interact with it. It exists to help the TPPs understand what is being provided and to build and test software against a real implementation.
5	Conformance Platform	Allows the Scheme Administrator to verify that software provided by Participants conforms to standards, by providing access to simulated set of simulated APIs along relevant test data and test cases.

7.4.2 Components Provided by the Data Provider

The following items are provided by, or under the responsibility of, the Data Provider.

#	Item	Description
1	Developer Portal	A Developer Portal is the interface between a set of APIs and/or other digital tools created by an API Provider and an API User.
2	API Catalogue	An API Catalogue is a library of APIs created by API Providers, organised by organisation, subject, purpose, and/or type. API Users can browse or search API Catalogues to find the APIs in which they are interested.
3	API Endpoints	The API endpoints are working endpoints that can receive requests and send responses.
4	API Server	An API Server is a software component that receives requests, handles authentication, validates requests, retrieves data from Back-End systems and sends responses.
5	Sandbox	A Sandbox is a test Environment in which new or untested APIs can be viewed and used securely. They are typically created by API Providers to allow API Users to safely test the API with their own systems and services without impacting live operations
6	Back-End Systems	The existing back-end system located at the Data Provider. These systems will include the accounting systems, validation servers, interfaces to payment systems.

7	Authentication Server	An authentication server is a special server responsible for verifying the identity of those to accessing the network.
8	Account Holder Authentication UI	The Account Holder Authentication UI is any mechanism that the Account Holder can use to Authenticate themselves to the Data Provider.

Components Provided by the TPP

The following items are provided by, or under the responsibility of, the Data Provider.

#	Item	Description
1	API Client	The API client is a software component that sends requests to an API server and receives the responses.
2	TPP Account Holder UI	The TPP Account Holder UI is the screen, app or interface that the Account Holder uses to communicate with the TPP.

8. Participant Management Standards

8.1 Registration Standards

8.1.1 The Scheme Administrator Identification Standards

Attribute	Description	Value
The Scheme Administrator Name	The Scheme Administrator will own the standards. The Name in plain text of The Scheme Administrator	Future
The Scheme Administrator Identifier	The Scheme Administrator will own the standards. The Identifier will be included in the API.	Future
The Competent Authority Name	The Competent Authority will approve Participants and issue Participant IDs. The Competent Authority MAY be the same as the Scheme Administrator, or it may be different. There may be multiple Competent Authorities.	Future
The Competent Authority Identifier	The Competent Authority ID and Participant IDs will be encoded into the digital certificates according to the TS 119.495 specification.	Future

8.1.2 Participant Roles

#	Name	Code
1	Data Provider	DP
2	Third Party Provider	TPP

8.1.3 Sector, Service and Operation Type Standards

These chapters are included in Chapter 9.2 to avoid duplication. Information here is for convenience, and the Data Dictionary should also be used as the master document for reference.

8.1.4 Participant ID

#	Item	Format	Notes
1	Participant ID	APInnnnnn	A unique identifier. Issued by the Competent Authority or Scheme Administrator “API” A standard code nnnnnn an 6-digit numeric code, randomly or sequentially chosen.

8.1.5 Participant Admissions Standards

8.1.5.1 Eligibility Criteria for Data Providers

Future. Out of scope of the standards.

8.1.5.2 Eligibility Criteria for TPPs

Future. Out of scope of the standards.

8.1.5.3 Capability Criteria for Data Providers

Future. Out of scope of the standards.

8.1.5.4 Capability Criteria for TPPs

Future. Out of scope of the standards.

8.2 Provisioning Standards

8.2.1 Participant Credential Standards

Credentials are the keys that allow access to APIs. In normal internet transactions, each API provided issues its own “access credential”, “client secret”, “API Key” or similar. In most / many Open Banking implementations each Participant has a single credential that is issued centrally, or by a number of trusted providers. This credential is standardised in terms of technical format for interoperability reasons and standardised in terms of trust.

8.2.1.1 Certificate Profile Standards

#	Attribute	Value	Notes
1	Certificate Profile	x.509 V3 / RFC5280	This standard profiles the X.509 v3 certificate revocation list (CRL) for use in the Internet.
2	Standard for encoding Scheme data in certificate	TS 119 495 v1.7.1	Electronic Signatures and Trust Infrastructures (ESI); Sector Specific Requirements; Certificate Profiles and TSP Policy Requirements for Open Banking
3	Revocation standards	TS 319 412-2	

4	Certificate type	QWACs	Qualified Website Authentication Certificate (QWAC) is a type of public key certificate defined by TS 119 495. QWACs will be issued to both TPPs and Data Providers and are used to establish mutually authenticated TLS (mTLS) session, to authenticate Participants and encrypt data passing between them.

8.2.1.2 Certificate Attributes

#	Attribute	Value	Notes	Example
1	Subject Distinguished Name OrganizationIdentifier	Participant ID	The Participant ID (defined in the Participation Standards: Participant ID standard) is placed in the organizationIdentifier attribute of the Subject Distinguished Name field in the public key certificate.	API123456
2	Authorisation Number	Participant ID	TS119 495 requirement: The Participant ID is encoded using the Authorisation Number table encoding below.	API123456
3	Roles	Participant Roles	TS119 495 requirement: The Participant Roles (defined in the Participation Standards) belong to the certificate subject are entered into the Roles fields.	DP TPP
4	National Competent Authority Name	Competent Authority Name	TS119 495 requirement: The Competent Authority ID will be entered into the NCA Name attribute.	Bank of Namibia
5	National Competent Authority ID	Competent Authority ID	TS119 495 requirement: The Competent Authority ID will be entered into the NCAId attribute structure in the presented order: 2 character ISO 31661-1 country code; hyphen-minus "-"; NCA ID (A-Z uppercase only, no separator).	NA-BON

8.2.1.3 Authorisation Number Table Encoding

#	Value	Notes	Example
1	PSD	3 character legal person identity type reference. PSD is the standard used within TS 119 495	PSD
2	Country Code	2 character ISO 3166-1 [8] country code representing the Competent Authority country	NA
3	-	hyphen-minus "-" (0x2D (ASCII), U+002D (UTF-8))	-
4	Competent Authority ID	2-8 character Competent Authority identifier without country code (A-Z uppercase only, no separator);	BON
5	-	hyphen-minus "-" (0x2D (ASCII), U+002D (UTF-8))	-
6	Participant ID	The Participant ID as specified by the Competent Authority. There are no restrictions on the characters used.	APInnnnnn

8.2.2 Data Provider and TPP Software Standards

8.2.2.1 Data Provider Developer Portal Standards

Data Provider Participants MUST Provide a Developer Portal and related technical, administrative and legal functions that meet the Developer Portal Standards, below.

Criteria	Standard for Test	Standard for Production
Access to the Developer Portal	The Developer Portal MUST be available to all active Participating TPPs. The Developer Portal MAY be available to other institutions or individuals.	The Developer Portal MUST be available to all active Participating TPPs. The Developer Portal MAY be available to other institutions or individuals.
Access to Data and funds.	The Developer Portal MUST NOT provide access to production Account Holder data or funds during testing.	The Developer Portal MUST provide access to production Account Holder data and funds. Account Holder data MUST only be exposed with the consent of the Account Holder and following the rules of the Scheme.
Contractual arrangements	The Developer Portal MUST allow Participating Data Providers to access the APIs and therefore their services and functionality with no additional contractual arrangements.	TBD
Pricing	Data Providers MUST allow Participating Data Providers to access the sandbox at no cost.	TBD

Criteria	Standard for Test	Standard for Production
Registration function for TPPs	The Developer Portal MUST allow Participating TPPs to register via a manual registration process. The Developer Portal MAY support an automated registration function, for TPPs. This process MUST mirror the registration to the Live environment, unless otherwise specified.	The Developer Portal MUST allow Participating TPPs to register via a manual registration process. The Developer Portal MAY support an automated registration function, for TPPs.
API documentation and publication standards	The Developer portal MUST provide documentation for all relevant APIs available as described in the API standards. The Developer Portal MAY provide reference, postman collection, and customer journey guidance for all APIs available.	
API Functionality	The Developer Portal MUST contain the relevant APIs (endpoints). The Developer Portal and the APIs it contains must follow the standards described in the API Standards.	The Developer Portal MUST contain the relevant APIs (endpoints). The Developer Portal and the APIs it contains must follow the standards described in the API Standards.
Reporting Functionality	The Developer Portal MUST be able to generate the reports required.	The Developer Portal MUST be able to generate the reports required.
Service Levels	The Developer Portal MUST be available 24/7 and support must be available during normal working hours.	The Developer Portal MUST meet the service level requirements defined in the Service Level Standards.
Test specific considerations	The Developer Portal MUST allow realistic testing of APIs.	

8.2.2.2 Data Provider Account Holder Authentication Software Standards

Data Provider Participants MUST Provide an Account Holder Authentication Interface that meets the Account Holder Authentication standards, the User Experience standards and the Service Level Standards.

8.2.2.3 TPP API Client Standards

TPPs MUST provide an API Client (Test) capable of securely interfacing with the Test and Live Developer portals of the Participant Data Providers, In compliance with the standards below.

Criteria	Standard for Test	Standard for Production
Functionality	The API Client MUST be able to communicate with a Data Provider's APIs, in compliance with the documentation.	The API Client MUST be able to communicate with a Data Provider's APIs, in compliance with the documentation.
Security	The API Client MUST be able to communicate with a Data Provider's APIs, securely in compliance with the documentation.	The API Client MUST be able to communicate with a Data Provider's APIs, securely in compliance with the documentation.
Use of Data and funds.	The API Client MUST use agreed Test Data during testing.	The API Client MUST be able to securely manage Account Holder Data.
Account Holder Authentication	The API Client MUST be able redirect to or use simulated Account Holder Authentication mechanisms as agreed during testing.	The API Client MUST be able redirect to agreed Account Holder Authentication mechanisms.

8.2.2.4 TPP Account Holder Interface Standards

TPPs MUST provide an Account Holder Interface that complies with the standards.

Criteria	Standard for Test	Standard for Production
Functionality	The TPP Account Holder Interface MUST be able to capture information entered from a simulated Account Holder.	The TPP Account Holder Interface MUST be able to capture information entered by an Account Holder.
Functionality	The TPP Account Holder Interface MUST be able to display relevant information received from the API Client.	The TPP Account Holder Interface MUST be able to display relevant information received from the API Client.
Security	The TPP Account Holder Interface MUST be able to securely exchange information with the TPP API Client.	The TPP Account Holder Interface MUST be able to securely exchange information with the TPP API Client.
User Experience	The TPP Account Holder Interface MUST comply with the User Experience standards.	The TPP Account Holder Interface MUST comply with the User Experience standards.

8.3 Setup and Testing Standards

TPPs and Data Providers MUST go through a set up and testing process in the sandbox (test) environment before going into the live environment.

TPPs and Data Providers MUST go through a set up process in the live environment.

8.3.1 Discovery Standards

A developer portal must be made available in Test and Live environments, as defined in Chapter 8.2

8.3.2 Sign-Up Standards

Sign up standards define the data that MUST be provided and whether sign up by an API is needed (known as DCR – Dynamic Client Registration).

Data Providers are free to implement as they choose.

8.3.3 Access Check Standards

In the live environment, Data Providers MUST check that the TPP trying to register

1) Is a valid TPP
2) Has access to the services to the services they wish to connect to.

In the Test environment, Data Providers may choose their own policy.

8.3.4 Contract Standards

Contract Standards define whether a contract is permitted; what elements a contract must contain; what form a contract must take.

There are not contract Standards defined.

8.3.5 Testing Standards

The standards that are required, effectively take the form of test cases and criteria that must be passed to move to prove that software is working properly and that the participant is ready to move to a live environment.

The testing program is out of scope and would generally form a separate document or annex. This chapter is kept in case specific requirements should be listed.

9. API Standards

9.1 API Architecture Standards

This chapter should be read in conjunction with the Data Dictionary, which also forms part of these standards.

9.1.1 Publication Standards

#	Requirement	Standard
1	API Architecture Standard	REST
2	API Documentation Standard	OpenAPI 3.1
3	Data Serialisation standard	YAML 1.2
4	Data Encoding standard	JSON
5	Data Exchange Standard	HTTP

9.1.1.1 API Versioning

#	Item	Standard
1	Major Version	The API standards will be versioned, with one major version across all endpoints. The major version number will be embedded in the URL Structure for the APIs. This allows the Scheme to support multiple major versions of the standards in production even if significant breaking changes occur between major versions.
2	Minor Versions	Each Endpoint may have multiple minor versions within a major version. Each endpoint version is independent of other endpoints. This allows Data Providers to upgrade functionality without waiting for the whole community to move

9.1.1.2 Open API Information Block

API standards will contain the following information, encoded in the info block of the API document.

Attribute	Value
Openapi	3.1.0
info/title	Namibian Open Banking APIs (or similar to be decided)
info/version	1.0

info/description	<Agreed text such as This document contains the technical definition of the Namibian Open Banking business and consent APIs which are provided as part of the ongoing Open Banking program supervised by the Bank of Namibia and run by the Namibian Open Banking Authority. More details can be found at www.namibianopenbanking.com.na. >
info/contact/name	< Specific to each Data Provider >
info/contact/URL	< Specific to each Data Provider >
info/contact/email	< Specific to each Data Provider >
info/license/name	<Licensing arrangement to be decided.>
info/license/URL	<Licensing arrangement to be decided.>

9.1.2 Resource Naming Standards (URI Structure)

URI Structure
Every API will have at least one endpoint, and every endpoint will have a “resource URI” code. The URI structure for API endpoints in the standards will be implemented as follows:
"https://"{provider}"/bon/"{version}"/" (<industry>) "/"<Resource URI>{?query-parameters}

Endpoint component	Description	Example
Provider	The location of the endpoint as provided by the Data Provider. The holder path is a path set by the Data Provider. It can be any URI desired by the Data Provider. While all authenticated endpoints must be accessible under the same holder path the Data Provider may stipulate a different holder path for unauthenticated endpoints.	api.anybank.na
Administrator	This is a static string representing the scheme endpoints. This static string allows for separation from other APIs available at the same base holder path and also allows for extension if the standards are adopted by another jurisdiction in whole or in part.	bon

Version	The major version of the API standards. This is not the version of the endpoint or the payload being requested but the version of the overall standards being applied. This version number will be "v" followed by the major version of the standards as a positive integer (e.g. v1, v12 or v76).	vX v1 will be used for phase 1 production
Industry	A static string used to separate APIs for a specific industry. As standards for new industries are defined the list of industry strings will be extended. Note that the currently accepted values for the industry component of the Base Path are: banking = "banking" string. For APIs related to banking and potentially wider financial services data common = "common" string. For APIs that potentially span industries	banking will be used initially.
Resource URI	The URI for the specific resource requested. This endpoint URI will be defined as part of the endpoint definitions for each API group. This service MUST follow the naming described in the endpoint dictionary.	accountbalance
query-parameters	These query-parameters MUST follow the parameters described in the endpoint dictionary.	123456789

Example
The endpoint to get a balance could be:
GET https://www.api.anybank.na/bon /v1/banking/accountbalance?123456789

Resource Path
The Resource Path is the portion of the URL including a Base Path and a resource URI location. The Resource Path string is defined as: <base-path> / <resource>

The Base Path
Base Path is the portion of the URL up to but not including the endpoint resource, i.e. the base path is the portion of the URL up to and including the <industry> component. The Base Path string is defined as follows:
https:// <provider> / bon / <version> / (<industry>)

The Resource URI

The resource URI follows the JSONAPI.org recommendation and for simplicity is quoted as the Method followed by the resource and any parameters.
GET .../accounts
GET .../accounts/{id}
GET .../accounts/transactions

9.1.3 Field Formatting Standards

9.1.3.1 Field Names and Valid Characters

All field names defined in either a request or response payload MUST be treated as case sensitive by clients and servers, and they MUST meet all the following conditions:

- Member names MUST contain at least one character.
- Member names MUST contain only the allowed characters listed below:
- U+0061 to U+007A, a-z
- U+0041 to U+005A, A-Z
- U+0030 to U+0039, 0-9

Additionally, the following characters are allowed in field names, except as the first or last character:

- U+002D HYPHEN-MINUS, '-'
- U+005F LOW LINE, '_'
- U+0024 DOLLAR SIGN, '$'

Other characters MUST NOT be used in field names.

9.1.3.2 Field Naming Style

Field names will be meaningful names with defined semantics.

Fields representing the same data in different payloads or different parts of a payload will have the same name.

Fields MUST NOT be named using reserved javascript tokens.

If a field name is a single acronym, it SHOULD be lowercase.

If a field name contains an acronym along with other words, it MAY be uppercase.

The first character in a field name SHOULD be lower case unless it is part of an acronym.

Array types SHOULD have plural field names. All other field names SHOULD be singular.

9.1.3.3 Field Data Types

Each field defined for the payloads of an endpoint will have an assigned data type.

The list of valid data types will be set out in the API data standards.

If a custom data type is required for a field, then the field SHOULD be classified as a string with a clear description of how the property value is to be interpreted or defined.

9.1.3.4 Empty/Null Fields

An empty field (i.e., a field that is not present in a payload) will be considered equivalent with a field that is present with a null value.

An empty string (“”) is not considered to be equivalent to null.

A Boolean value of false is not considered to be equivalent to null. Optional Boolean fields, by implication, have three possible values: true, false and indeterminate (i.e., null).

9.1.3.5 Mandatory/Optional Fields

Each field defined for the payloads of an endpoint MUST have an assigned status of mandatory, optional, or conditional.

- Mandatory fields MUST be present and have a non-null value in a request or response payload for the payload to be considered valid.
- Optional fields MAY be present, but this is not guaranteed. It is also valid for these fields to be present but have a null value. Note that optional fields indicate that data may sometimes not be held by a Data Provider, and this is an expected scenario.
- Conditional fields MUST have an associated conditional statement. If the conditional statement is true in a specific request or response, the field is considered mandatory. If the conditional statement is false, then the field is considered optional.

Optional fields are not considered optionally implementable by a Data Provider if they hold the data. If a Data Provider holds data in digital form for an Account Holder that is represented in a payload, then it is expected that this data will be shared when authorised by the Account Holder.

9.1.3.6 Currency formats

Currency value format MUST have two decimal places. The last two digits of the number shall be the decimal digits. Example shown below for Namibian Dollar (NAD) 10000:

Amount: 100.00

9.1.4 Pagination

#	Attribute	Value
1	Maximum Page size	A maximum page size of 1000 records is assumed for all endpoints (unless otherwise stipulated in the endpoint definition). If a page size greater than this maximum is requested, then an Invalid Page Size error SHOULD be returned.

Each API endpoint that can return multiple records will stipulate whether pagination is supported for the endpoint or not, in the API Data Standard. For endpoints that will return less than a reasonably sized page of results in most circumstances support for paging may not be included.

Note that the use of paging for an endpoint does not require or preclude the use of filtering query parameters. It is expected that filtering and paging will be applied independently of each other.

Data Providers are not expected to implement pagination with transaction isolation. The underlying dataset may change between two subsequent requests. This may result in situations where the same transaction is returned on more than one page.

If a page size greater than this maximum is requested, then an Invalid Page Size error SHOULD be returned.

9.1.4.1 Pagination and Query Parameters

The TPP will stipulate pagination requirements on the request using query parameters. When paging is supported, the consumer MAY provide the following query parameters:

#	Attribute	Value	Default
1	Page	The page number being requested (with the first page being 1)	1
2	Page-size	The number of records to return in each page	25 records

9.1.4.2 Pagination and Links Response Object

In addition to the data requested a holder MUST provide the following Links object in the response payload. In the links object the following fields are to be provided.

#	Attribute	Value
1	first	first - A URI to request the first page. Mandatory if this response is not the first page.
2	last	last - A URI to request the last page. Mandatory if this response is not the last page.

3	prev	prev - A URI to the previous page. Mandatory if this response is not the first page.
4	next	next - A URI to the next page. Mandatory if this response is not the last page.

9.1.4.3 Pagination and Meta Response Object

In addition to the data requested a holder MUST provide the following meta object in the response payload. In the meta object the following fields are to be provided:

#	Attribute	Value
1	totalRecords	totalRecords - The total number of records in the set. This field MUST be present.
2	totalPages	totalPages - The total number of pages in the set. This field MUST be present. If totalRecords is 0 totalPages MUST be 0.

For each of these fields, the page size specified in the request can be assumed when calculating values.

For performance reasons, Data Providers may wish to support other pagination patterns such as cursors or continuation tokens. While the standard does not explicitly support these additional mechanisms, it is considered allowable to implement these patterns and expose them via the pagination links.

In this scenario, the URIs included in the links for other pages may not be compliant with the standard and may, instead, include other query parameters that support another pagination pattern. It is expected that all other pagination requirements such as link fields and meta fields will still be supported if other patterns are implemented.

To allow for a more performant implementation, data consumers are encouraged to utilise pagination links wherever possible and only use constructed URIs for the first page or if random access to a specific set of records is required.

9.1.5 HTTP Request Headers

HTTP Request Header Standards define the supported HTTP Request Headers. HTTP request headers MAY/MUST contain the following elements.

Header Field	Description	Mandatory?
ParticipantId	Field referencing the Participant ID of the TPP Participant. This value must match the ID found within their Participant Certificate.	Mandatory

ContentType	Standard HTTP Header. Represents the format of the payload provided in the request. The media type must be set to application/json. Mandatory for PUT and POST calls.	Conditional
x-v	Version of the API endpoint requested by the client. Must be set to a positive integer. If the version requested is not supported, then the holder must respond with a 406 Not Acceptable.	Mandatory
Accept	If specified, the media type must be set to application/json, unless otherwise specified in the resource endpoint standard. If set to an unacceptable value, the holder must respond with a 406 Not Acceptable. If not specified, or a wildcard (*/*) is provided, the default media type is application/json.	Optional

9.1.6 HTTP Response Headers

HTTP Request Header Standards define the supported HTTP Request Headers. HTTP Response headers contain the following elements.

Header Field	Description	Mandatory?
ParticipantId	Field referencing the Participant ID of the Data Provider Participant. This value must match the ID found within their Participant Certificate.	Mandatory
RetryAfter	Field indicating the time (in seconds) that the client should wait before retrying an operation. The holder should include this header along with responses with the HTTP status code of 429 Too many requests.	Optional
x-v	The version of the API endpoint that the holder has responded with.	Mandatory

9.1.7 Request Payloads

Each API request with a payload MUST have a JSON object at the root level known as the root object.

This root object MUST contain a data object to hold the primary data for the request. The definition of the contents for the data object will be defined separately for each endpoint.

The root object MAY contain a meta object, which is used to provide additional information such as second factor authorisation data, traffic management, pagination counts, or other purposes that are complementary to the workings of the API. The root object will contain a meta object if, and only if, it is specifically REQUIRED by the endpoint as defined in the API Data Standard.

#	Root Object	Needed in Request Object
1	Data Object	Mandatory
2	Links Object	N/A
3	Meta Object	Conditional (if specified)
4	Errors Object	N/A

9.1.8 Response Payloads

Each API response payload MUST have a JSON object at the root level known as the root object.

9.1.8.1 Successful Responses (200 code)

All endpoints MUST have HTTP Status codes 200 as defined in the data dictionary

If the response is successful (200 OK), the root object:

#	Root Objects	Needed in 200 Response
1	Data Object	Mandatory
2	Links Object	Conditional
3	Meta Object	Conditional
4	Errors Object	N/A

The definition of the contents for the data object and meta object is defined separately for each endpoint in the API Data standards.

9.1.8.2 Unsuccessful Responses (not 200 code)

All endpoints MUST have responses defined for HTTP Status codes 400, 401, 403, 404, and 500 as defined in the data dictionary and MAY have HTTP status codes, 201, 409 or other codes, as defined in the data dictionary.

#	Root Objects	Needed in unsuccessful Response
1	Data Object	N/A
2	Links Object	N/A
3	Meta Object	N/A
4	Errors Object	Conditional

Unsuccessful responses MAY contain an errors object (as per the specific endpoint definition)

9.2 API Use Cases

Functional endpoints MUST be tagged with one or more of the following sector and services identifiers.

Data Providers MUST NOT let TPPs without these permissions, access these endpoints.

9.2.1 Supported Sectors

#	Sector Code	Sector Name	Description
1	All	Common	Common Services. For APIs that potentially span sectors or services, e.g. Consent endpoints, reporting endpoints.
2	Banking	Banking	For APIs related to banking and potentially wider financial services data

Other industries can be added in later phases.

9.2.2 Supported Services

The following service types will be supported.

#	Sector Code	Service Code	Service Name	Description
1	All	Common	Consent Endpoints	The supported endpoints and their versions that are used for consent.
2	Banking	PIS	Payment Initiation Endpoints	The supported endpoints and their versions that are used for Payment Initiation Services
3	Banking	AIS	Account Information Endpoints	The supported endpoints and their versions that are used for Account Information Services.

NB. Subscription services will not be supported in this phase.

9.2.3 Supported Operation Types

The following operation types will be supported for each service.

#	Service Code	Operation Code	Name	Description
1	AIS	Read	AIS.Read	Account information will be limited to reading (not updating) information in the first phase.

2	PIS	Write	PIS.Write	Payments can be created by Account Holders through TPPs.
3	PIS	Read	PIS.Read	Payment statuses information can be read by TPPs for payments they have made, support account holders.

N.B. The ability to change account information (AIS.Write), will not be supported in this phase.

9.2.4 Supported Resource Objects (Banking)

9.2.4.1 List of Resource Objects

The following resource objects are supported.

#	Name	Description
1	Accounts	Information about an account, such as identifier, account holder name, type.
2	Balances	Balances of an account, typically a type, an amount and a currency. There may be multiple balance types.
3	Transactions	Key information relating to a specific account. Transactions information includes identifiers, dates, amounts, currency, status.
1	Payments	Payments or movements on the account made from a payer to a payee, including interest payments, fees and internal transfers.
2	Payment Status	The status of a payment that has been made.
3	Beneficiaries	Beneficiaries (Payees) that have been added by Account Holders and organisations or individuals they pay frequently.

9.2.4.2 Accounts

Information about an account includes information such as an identifier, account holder name, type. See data dictionary.

Supported Account Types.

Whether an account type is supported or not, is shown in the table below.

Data provider	Account type	Consumer	Small Business	Enterprise
PSPs regulated by BON	e-Wallet	Yes	Yes	No
PSPs regulated by BON	Current Account	Yes	Yes	No
PSPs regulated by BON	Savings Account	Yes	Yes	No

PSPs regulated by BON	Credit Card	Optional	Optional	No
PSPs regulated by BON	Loan Accounts	Optional	Optional	No
Financial Institutions regulated by NAMFISA or others	Insurance Accounts	No	No	No
Any	Email Accounts, Credit Bureau Accounts, Prepaid cards, Unit trust accounts Investment accounts, Vouchers	No	No	No

There will be one Account Data object, that will hold data about accounts. Any account that can support this data set may be supported. Additional data for specific account types will require additional and specific API Standards that are outside the scope of this phase.

9.2.4.3 Balances

Balances of an account, typically include a type, an amount and a currency.

See Namibian Open Banking Standards Data Dictionary.

Supported Balance types

There are multiple types of balance possible within the scope of supported accounts, e.g. actual balance, available balance.

Balance types are not defined, based on the assumption that each Data Provider may have its own way displaying balance to Account Holders.

Each Data Provider must define, encode and document Account Types on their developer portal.

9.2.4.4 Transactions

Key information relating to a specific account. Transactions information includes identifiers, dates, amounts, currency, status. See data dictionary.

Supported Transaction types

There are multiple types of transaction possible within the scope of supported accounts, e.g. card payment, incoming EFT payment, outgoing EFT payment, bank charge.

Transaction types are not defined, based on the assumption that each Data Provider may have its own way displaying Transactions to Account Holders.

Each Data Provider must define, encode and document Transaction Types on their developer portal.

9.2.4.5 Payments

Payments or movements on the account made from a payer to a payee, including interest payments, fees and internal transfers.

Supported Payment Types.

Which payment types are supported is shown in the table below.

National	Payment Type	Consumer initiated*	Small Business initiated*	Enterprise initiated*
Domestic	On-us transactions whether between accounts with different account holders, the same account holders or wallets.	Yes	Yes	No
Domestic	EFT: Enhanced credit transfers. EnCR	Yes	Yes	No
Domestic	EFT: Near real time credit transfers. NRTC	Yes	Yes	No
Domestic	RTGS	No	No	No
Domestic	Instant Payment (IPP)	Future	Future	No
Domestic	Card Payments (i.e. transactions that goes across the card network)	No	No	No
Domestic	Debit	No	No	No
Cross border	CMA	No	No	No
Cross border	SADC RTGS	No	No	No
Cross border	FOREX	No	No	No

*Initiated refers to the case where the Party is the Payer / Originator of the payment for a credit transfer

9.2.4.6 Payment Status

The status of a payment that has been made with visibility of its progress.

9.2.5 Supported API Use Cases

9.2.5.1 Supported AIS API Use Cases

The following API Use Cases will be supported for the AIS service

#	API Use Case	Summary	API Use Case description
1	List Accounts	Obtain a list of accounts	The List Accounts API allows a third party to obtain a list of accounts at the request of an Account Holder. A list of accounts will be returned and should reflect the list of accounts that the same Account Holder would see if they accessed their account through a mobile app or web browser. A query parameter allows the list of accounts returned to be filtered by "open" and "closed" status.
2	Get Account Balance	Obtain the balance for a single specified account	The Get Account Balance API allows a third party to obtain the balance for a specific account on the request of an Account Holder. The request returns the current balance, but also returns other balances, such as the available balance and credit limits, if appropriate.
3	List Transactions	Obtain transactions for a specific account.	The List Transactions API allows a third party to obtain a list of transactions for a specific account on the request of an Account Holder. Each transaction can contain a type, status, description, posting date, amount, currency reference.

9.2.5.2 Supported API Use Cases: PIS

The following API Use Cases will be supported for the PIS service.

#	API Use Case	Summary	API Use case description
1	Make Payment	Make a bank payment	The Make Payment API allows a third party to initiate a payment on the request of a customer. A positive API response code indicates whether the payment initiation has been successful or not, i.e., whether the financial institution has successfully accepted the payment instruction, not whether the payment itself is credited correctly.

			The API returns a PaymentId, which is then used to track the progress of the payment itself. Multiple “Make Payment” endpoints may be needed if there are multiple Payment types.
2	List Beneficiaries	Obtain the list of Beneficiaries	The Get Beneficiaries API allows a third party to get the list of Beneficiaries (Payees) that have been added by Account Holders and organisations or individuals they pay frequently. This allows the TPP to present the list of Beneficiaries to the Account Holder when the Account Holder is making a payment
3	Get Payment Status	Get the status of a payment	The Get Payment Status API allows a third party to track a payment that they previously initiated, on the request of a customer. The PaymentId is used as the reference for the payment, as well as to track the progress of the payment itself.

Example Market use cases based on Account Information.	N.B. These **API use cases** can be defined, and will support **market use cases** listed, although it will be the TPPs that decide what is needed.
Account aggregation	**Example Market use cases based on Payment Initiation.**
Confirmation of funds	Bill payments
Personal finance management	Ecommerce payments
Wealth management	Wallet top ups
Credit scoring	Loan repayments
Know Your Customer (KYC)	Domestic remittances
Account ownership verification	Payments to friends, family, colleagues
Transaction monitoring	Invoice payments
Consumer spending profile	Purchase on account
Loyalty programs	Payment method integration
Accounting Services	
Verify Income (for house rental)	
Verify age (for gambling authorisation)	
Car Financing	
Automated budgeting and alerts, with e.g. transaction monitoring	

9.3 API Data Sets and Data Dictionary (Data Standards)

The Data sets are included in Namibian Open Banking Standards Data Dictionary.

9.4 Security Standards

Security standards include security between Data providers and TPPs. Security that is linked to consent is handled in the following chapters.

The following standards are put in place to technically secure communications between participants.

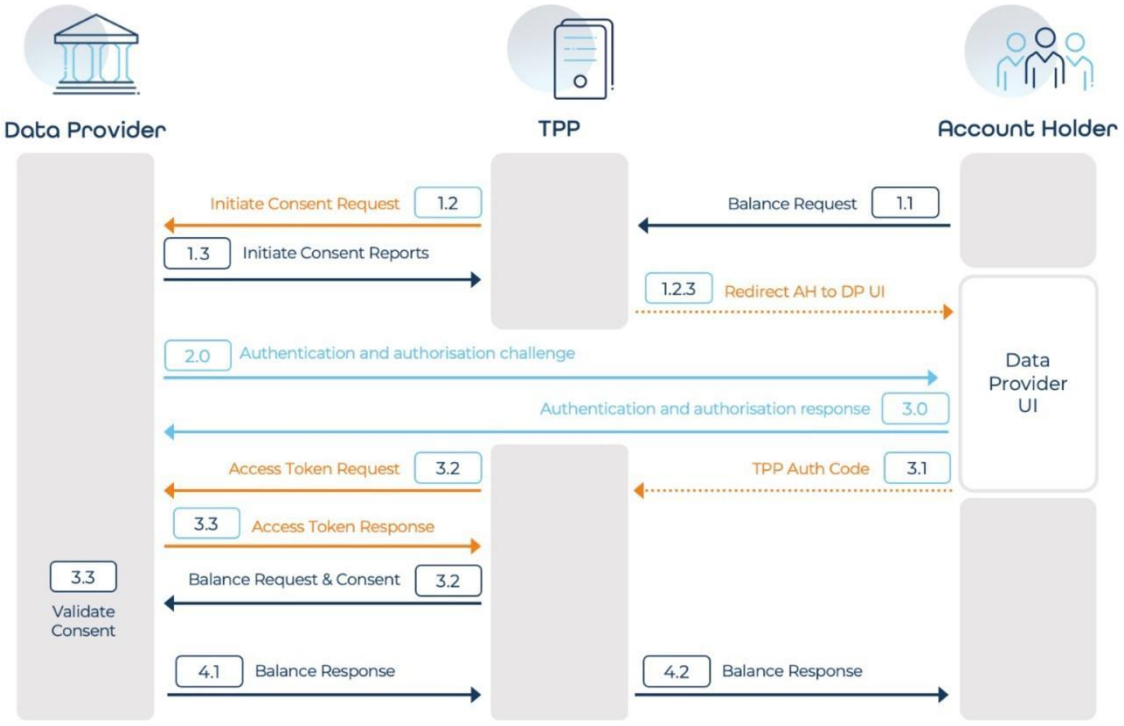
Area	Standard
Transport layer security	All API requests and all API responses MUST be secured by Mutual TLS (MTLS) as described in RFC 8210. *mTLS helps ensure that traffic is secure and trusted in both directions between a client and server. This provides an additional layer of security for users who log in to an organisation's network or applications. It also verifies connections with client devices that do not follow a login process. mTLS prevents various kinds of attacks and is used in zero trust security frameworks.*
Authentication of TPPs by Data providers	Data Providers MUST validate TPPs in real time, i.e. • Authenticate the identity of the TPP using the allocated Participant Credential. • Verify that they are Participants in the Scheme. • Verify that they have the correct status. • Verify that they have permission to carry out the action as determined by the Access control rules.
Authentication of Data Providers by TPPs	TPPs MUST validate that they are exchanging data with the correct Data Provider and authenticate the identity of the Data Provider using their Participant Credential.
TLS certificates	All Participants MUST use a production grade Participant Credential that meets the standards described in the chapter “Participant Credential Standards” for securing TLS communications.
Signing certificates	All Participants MUST use a production grade Participant Credential that meets the standards described in the chapter “Participant Credential Standards”.

9.5 Consent and Customer Authentication Standards

Consent has so far been described in terms of a “Consent Token”. At the technical level this is a pair of tokens, and Access Token and a Refresh Token.

The standards described here are based on the Authorization Code with PKCE standard as outlined in RFC7636 sections 4.1 and 4.2.

9.5.1 Consent Steps



9.5.1.1 Initiate Consent request: Pushed Authorisation Request API

The TPP will initiate a consent flow with the Data Provider.

1. To do this, the TPP will generates a **code challenge** and **code verifier** in accordance with “RFC 7636 Proof Key for Code Exchange by OAuth Public Clients,” sections 4.1 and 4.2.
2. The TPP will submit the code challenge to the Data Provider with details of the consent request as a 'Pushed Authorization Request'.
3. This is submitted to the Data Provider which will validate the request and return a **request_uri**.

Initiating consent is done through a Common API “par” (pushed Authorisation request) based on RFC 9126 “OAuth 2.0 Pushed Authorization Requests”

9.5.1.2 Account Holder grants Consent

Account Holder Consent has three parts

1) After obtaining a **request_uri**, the TPP will redirect the Account Holder to the Data Provider to authenticate their identity and confirm the details of the consent request.

2) The Data provider confirms consent request with the Account Holder. This includes
    a. The Data Provider sending a challenge to the Account Holder
    b. The Account Holder sending a successful response to the Data Provider.
3) If the Account Holder grants the consent request, the Data Provider issues an **authorization code.** This authorization code is delivered it to the TPP.

The authorization code MUST expire shortly after it is issued to mitigate the risk of leaks. A maximum authorization code lifetime of 10 minutes is recommended.

API definitions are not provided as the data sent is not between the TPP and the Data Provider, nevertheless we suggest a recommended format is provided for

i) A mechanism to redirect the Account Holder app to the Data Provider App.
ii) A call back webhook definition to tell the TPP how to get the Authorization code.

9.5.1.3 Retrieve Access Token(s): POST Token API

After obtaining the **authorization code**, from the Account Holder, the TPP will request a Token from the Data Provider using the Token API.

1) The TPP sends the Token request and provides both the **Authorization code** and the **code challenge** created during the Pushed Authorization request..
2) The Data Provider validates the authorization code against the code verifier and code challenge supplied previously before.
3) If valid, the Data Provider responds with an **access token**. Depending on the duration and type of consent granted, the Data Provider may also issue a **refresh token.** This refresh token may be used to obtain a new access token after a previous token expires in future transactions.

For API definitions see the data dictionary.

9.5.1.4 Consent revocation: POST Revoke API

Should the Account Holder withdraw their consent via the TPP, the TPP must instruct the Data Provider to revoke their refresh token. Alternatively, if the Account Holder instructs the Data Provider to withdraw consent, then the Data Provider will revoke the refresh token themselves. Once a refresh token has been revoked it may no longer be used to obtain new access tokens.

For API definitions see the data dictionary.

9.5.2 Consent Scopes

Scopes Name	Description
banking:accounts.basic.read	Ability to read account information
banking:payments.write	Ability to initiate a payment
banking:payments.read	Ability to read the status of a payment request
consent:authorisationcode.write	Ability to send an authorisation code
consent:authorisationtoken.write	Ability to send an access token

9.5.3 Maximum Consent Duration

Value	Description
180 days	The Maximum Duration of Consent

9.5.4 Strong Customer Authentication Standards

No minimum requirements are provided as it is assumed that

- All Data providers are regulated by BoN.
- All Data providers therefore have compliance obligations and minimum standards for providing access to bank accounts and wallets.

9.6 API UX Standards

9.6.1 Branding Standards

Item	Description
Scheme Name	The Name of the Scheme as known to the public
Scheme Brand	An image that is consistently shown when the Scheme is used.
Scheme Glossary	A set of terms that will be used in public phasing communication by all participants. These terms may or may not be the same as the terms used in the Scheme standards documentation.

9.6.2 Public Information Standards

Public information standards represent a minimum of information that MUST be presented. Other information can also be provided.

9.6.2.1 Scheme Webpage

The Scheme Administrator will provide a Webpage for Account Holders, so that all Account Holders can see a single centralised description of the Open Banking Services.

Item	Description
Scheme Description	A description of the mission and purpose of the Scheme
List of Data Providers	A public list of Participating Data Providers
List of TPPs	A public list of Participating TPPs
Account Holder Instructions	Instructions for Account Holders about what they should do if they have problems

9.6.2.2 TPP Webpage

TPPs will provide a Webpage for Account Holders, so that all Account Holders can see a single centralised description of the Open Banking Services.

Item	Description
Scheme Description	A description of the mission and purpose of the Scheme.
Scheme webpage link	A link to the Scheme Webpage.
TPP Participation identifier	A statement that the TPP is a participating organisation, including Identification numbers and details.
Account Holder Instructions	Instructions for Account Holders about what they should do if they have problems

9.6.2.3 Data Provider Webpage

Data Providers will provide a webpage for Account Holders, so that all Account Holders can see a single centralised description of the Open Banking Services.

Item	Description
Scheme Description	A description of the mission and purpose of the Scheme.
Scheme webpage link	A link to the Scheme webpage.
Data Provider Participation identifier	A statement that the Data Provider is a participating organisation, including Identification numbers and details.
Account Holder Instructions	Instructions for Account Holders about what they should do if they have problems

9.6.3 Flow and Mandatory Text

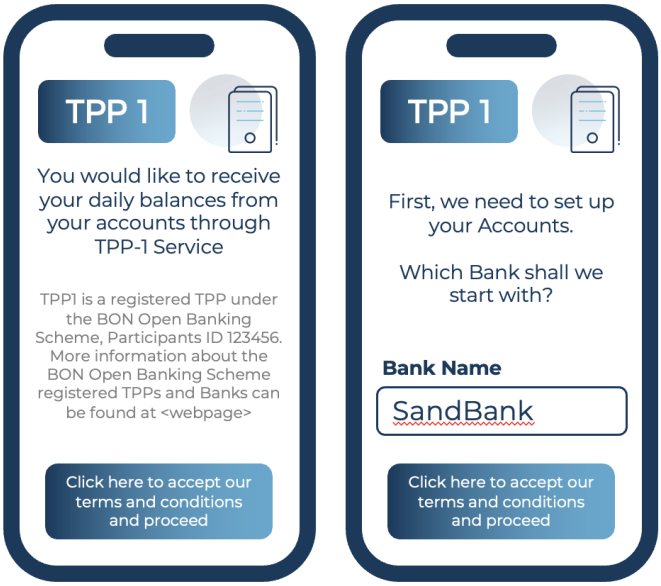
The following mandatory text is to be shown during the user journey. The text itself may be developed, but the following elements must be in place.

9.6.3.1 Contracting and Configuration with the TPP

The Account Holder subscribes to a TPP service The Account Holder is offered an App or service. During this process the following text must be shown.

#	Mandatory Text Element	Example
1	Scheme Statement	This service is made within the rules of the <Scheme Name>.
2	TPP Participation Statement	<TPP Name> is a registered TPP under the <Scheme Name>, Participant ID <12345678>
3	Scheme webpage link	More information about the <Scheme Name>, registered TPPs and Banks can be found at <Scheme Webpage>
4	TPP webpage link	Click here <webpage> for more information and support.

Mockup



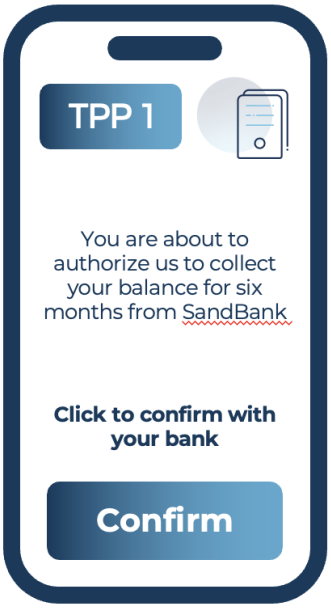
Contract & Configure **Contract & Configure**

9.6.3.2 Transaction: Account Holder Request to TPP

The Account Holder will make a request for a balance, a payment to the TPP.

#	Mandatory Text Element	Example
1	Confirmation that the transaction is taking place within the rules of Scheme.	This transaction is made by <TPP Name>
2	Confirmation of the what the Account Holder is about to consent to.	Allow <who> to <do what> for <how long>

Mockup



9.6.3.3 Transaction: Authentication and Authorisation

The Account Holder receives a challenge from the Data Provider and responds to the challenge.

#	Mandatory Text Element	Example
1	DP Participation Statement	<DP Name> is a registered participant of the Scheme
2	DP webpage link	Click here for more information and support.
3	Mandatory text on Consent rights	You are about to share <what>, <with who>, <for how long>.
		If you wish to proceed, please <consent action>

Mockup

Data Providers should not unnecessarily require multiple consent flows in the transaction authentication and authorisation process for multiple AIS actions with the same Account Holder and Bank, i.e. it cannot be that an Account Holder

look at accounts (consent requested)
look at my transactions (consent requested)
look at transaction detail (consent requested).

There may be cases where more than one consent and authorization may be needed on one customer journey.

Account Holders logs on to get into the online banking or app and views accounts and transactions (consent requested)
the Account Holder makes a payment and is requested to provide a second strong customer authentication to authorise the payment.

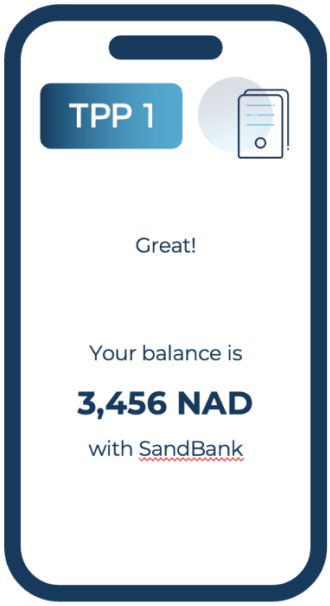
Account Holders are used to providing two logons today for good risk-based reasons. There is no issue with the same happening within the API flow. The golden rule should be that if it is acceptable in the bank App or browser, it should be OK on the API.

9.6.3.4 Transaction: Response to the Account Holder from the TPP

Following a successful call, the Data Provider responds to the TPP who then takes the necessary action towards the Account Holder, whether this is confirming that a payment has been made, showing a balance or providing a pie-chart.

Response	No specific text	

Mockup



Response

9.7 API Service Level Standards

9.7.1 TPP Service Levels Towards Data Providers

TPPs have an obligation not to send too many requests to data providers, and that the requests that are sent are well formed.

All requests that are sent by TPPs are effectively requested by Account Holders, but a TPP could (theoretically) perform a balance check every second as part of a monitoring function. This would be an unreasonable load on Data Providers, and so a limit is imposed.

#	Name	Description	Level
1	Maximum requests	Maximum automated requests per day from the same Account Holder	4

2	Error rate.	The Error Rates is the percentage of API requests that result in errors.	

Errors are defined as follows:

1. HTTP Status Codes: Responses with status codes in the 4xx (client errors) and 5xx (server errors) ranges
    a. 4xx Errors: 400 Bad Request, 401 Unauthorized, 403 Forbidden, and 404 Not Found.
    b. 5xx Errors: 500 Internal Server Error, 502 Bad Gateway, 503 Service Unavailable, and 504 Gateway Timeout.
2. Timeouts: If the API does not respond within a specified time frame, it is considered an error. This can be due to network issues, server overload, or other factors causing delays.
3. Invalid Responses: Even if the API returns a 2xx status code, the response might still be considered an error if the content is invalid or does not meet the expected format or schema.
4. Application-Specific Errors: These are custom error codes or messages defined by the API provider to indicate specific issues related to the application's business logic.

9.7.2 Data Provider Service Levels Towards TPPs

#	Name	Item	Service Level
1	Availability.	Availability measures the percentage of time the API is operational and accessible. An Allowable Downtime Window (ADW) is permitted each month for maintenance or updates.	99.9% excluding ADW
2	Median Response time	Response time measures how quickly the API responds to requests.	300 milliseconds
3	Error rate.	The Error Rates is the percentage of API requests that result in errors.	

10. Ongoing Management Standards

10.1 Notifications and Reporting

The Scheme manager will provide reports and notifications to Participants and if needed, regulators. These reports or notifications may be operational (e.g. notice of new Participants, or informational (e.g. transaction volumes).

Participants will provide reports to the Scheme Manager of transaction volumes, service level performance or other information as required.

Reporting standards define the content, periodicity and audience of reports that will be created.

10.1.1 Transaction Reporting

A monthly report to be sent by Participants to the Scheme Manager.

#	Report	Description
1	Report Name	Transaction Report
2	Report Purpose	To provide summary data on the volume of transactions, their type and their success rate.
3	Report Content Type	API Calls
4	Report Sender	Data Providers and TPPs
5	Report Audience	Scheme Manager
6	Report Periodicity	Monthly
7	Report Distribution Channel	Email.
8	Report Contents	For each endpoint # Calls received # Calls received successfully # Calls received failed For each endpoint & TPP # Calls received # Calls received successfully # Calls received failed For each endpoint & TPP & Call received (failed) & error code

10.1.2 Service Level Reporting

#	Report	Description
1	Report Name	Service Level Report
2	Report Purpose	To provide summary data performance against service levels.
3	Report Content Type	Service Levels
4	Report Sender	Data Providers and TPPs
5	Report Audience	Scheme Manager
6	Report Periodicity	Monthly
7	Report Distribution Channel	Email
8	Report Contents	For each service level • A statement of target • Compliance metric against that target • Service level met (Y/N) • Other information (free text if required)

10.1.3 Dispute Reporting

#	Report	Description
1	Report Name	Dispute Report
2	Report Purpose	To provide summary of Account Holder disputes.
3	Report Content Type	Account Holder Disputes
4	Report Sender	Data Providers and TPPs
5	Report Audience	Scheme Manager
6	Report Periodicity	Monthly
7	Report Distribution Channel	Email
8	Report Contents	Statistics on each type of Account Holder Dispute (in scope) at the priority defined to be in scope.

		Total number of disputes in scope. Total number of disputes in scope by type Total number of disputes in scope by priority Total number of disputes in scope by counterpart Participant.

10.2 Monitoring, Helpdesk and Support Processes

10.2.1 Monitoring Standards

Participants monitor transactions, services levels, audit logs to verify that APIs are meeting service level standards to take actions if problems arise.

The Scheme Manager will monitor reports and market activity.

10.2.2 Helpdesk Standards

Helpdesk covers topics between participants. Problems related to Account Holders are covered under "Dispute Resolution". Helpdesk Standards should acknowledge that there is a difference between calls and problems from developers on technical matters vs calls on payment reconciliation or business matters, providing different channels as needed.

Participants will provide support to other Participants should questions or problems arise. The Scheme Manager will provide support to Participants if needed.

Helpdesk standards describe how participants provide support to each other, including communication channels and response times.

10.2.3 Incident Management Standards

The Scheme Manager and scheme Participants will have incident management processes designed to resolve problems, minimise impacts and alert other Participants, Account Holders (if necessary).

Incident Management standards describe what types of incidents are defined, how different incident types are prioritised and what information must be shared with which parties in the case of different types of incidents.

We do not intend to define these standards in further detail as we believe this is party of future ongoing operationalisation, and that channels, priorities etc are already defined within the Namibian payments ecosystem.

10.3 Dispute Resolution Standards

The Scheme Manager will organise processes to help Participants resolve problems with each other, to ensure that Account Holders receive answers, and that repetitive problems can be tackled. Dispute resolution will not include direct communication to Account Holders.

Dispute Standards include dispute types, response times, error codes and channels that will be followed during a Dispute.

10.3.1 Dispute Type Standards

Dispute Type Standards define the type of dispute, allowing all Participant to share the same interpretation of an issue across the market and so to create uniformity around the terms and issues.

10.3.2 Dispute Channel Standards

Dispute Data Exchange standards provide consistent and standardized requirements about what data is exchanged between Participants.

10.3.3 Dispute Service Level Standards

Dispute Service Level Standards describe maximum response times to ensure that critical issues are treated quickly, and that Account Holders get answers in a reasonable time frame.

10.3.4 Dispute Priority Standards

Dispute Priority Standards provide a way of signalling the urgency and the importance of a dispute.

We do not intend to define these standards in further detail as we believe this is party of future ongoing operationalisation, and that channels, priorities etc are already defined within the Namibian payments ecosystem.

10.4 Standards for Change Management

Changes to or within the Scheme should be managed with care. Managing change involves identifying, socializing change requests and may require review and feedback. Time may be needed for Participants to adapt, and there are issues around versioning to be considered.

10.4.1 Change Management Standards for Data Providers

Changes made by Data Providers to API functionality can have a significant impact on TPPs, even if the changes are communicated and bring additional benefits.

Data Providers should respect the timings below when making changes to published APIs.

#	Name	Description	Value
1	Minimum change notification window	If changes are to be made to APIs, they must be notified in advance. The Minimum change notification window provides the minimum time period in days.	180
2	Minimum Testing availability period.	If changes are to be made to APIs, they must be made available in the developer environment. The Minimum Testing availability period provides the minimum time period in days.	90

10.4.2 Change Management Standards for the Scheme Manager

We do not intend to define this chapter in further detail as we believe this is party of future ongoing operationalisation, and that channels, priorities etc are already defined within the Namibian payments ecosystem. Nevertheless, we would expect to see rules around

- Governance: who submits requests for changes, who prioritises requests for changes, who is consulted on changes.
- Timelines for major releases (e.g. an annual cycle)
- Standards around emergency changes, what constitutes an emergency change, and who decides whether an emergency change is needed.

11. ANNEX: External Standards

11.1 Standards References

The following table provides a cross reference of external technical standards used, mapping them to a requirement or purpose.

Requirement	Purpose of standard	Proposed Standard	Proposed Standard Name
General	Provides a standard architecture.	REST	REST
General	Provides a standard way of exchanging data.	RFC 7231	HTTP
Language	Provides the standard for encoding data.	RFC 7159	JSON
Date and Time	Provides the standard for designating date and time.	RFC 3339	ISO Date & Time
Currency	Provides the standard for designating currencies.	ISO 4217	ISO Currency codes
Payload(s) standards	Provides a data dictionary where all parties understand the meaning of each data element.	ISO 20022	ISO 20022
Naming and addressing	Provides the standard for uniquely locating the resource.	RFC 3986	URI Syntax
Security: Transport layer	Secure (transport) communication - with one way authentication.	RFC 5246	TLS Protocol MA-TLS
Security: Transport layer	Transport with encoding	RFC 2818	HTTP over TLS
Security: Application layer	Signing standards	RFC 7797	JWS
OAUTH 2.0	Proof Key for Code Exchange by OAuth Public Clients	RFC 7636	OAUTH 2.0
OAUTH 2.0	The OAuth 2.0 Authorization Framework	RFC 6796	
OAUTH 2.0	OAuth 2.0 Pushed Authorization Requests	RFC 9126	
OAUTH 2.0	Best Current Practice for OAuth 2.0 Security	RFC 9700	

11.2 Normative References

Reference	Description	Version
[RFC2119]	Key words for use in RFCs to Indicate Requirement Levels https://tools.ietf.org/html/rfc2119	Mar 1997
[JSON]	The JavaScript Object Notation (JSON) Data Interchange Format: https://tools.ietf.org/html/rfc8259	Dec 2017
[RFC2397]	The "data" URL scheme: https://tools.ietf.org/html/rfc2397	Aug 1998
[RFC3339]	Date and Time on the Internet: Timestamps: https://tools.ietf.org/html/rfc3339	Jul 2002
[RFC4122]	A Universally Unique IDentifier (UUID) URN Namespace: https://tools.ietf.org/html/rfc4122	Jul 2005
[JWA]	JSON Web Algorithms (JWA): https://tools.ietf.org/html/rfc7518	May 2015
[JWE]	JSON Web Encryption (JWE): https://tools.ietf.org/html/rfc7516	May 2015
[JWK] / [JWKS]	JSON Web Key (JWK): https://tools.ietf.org/html/rfc7517	May 2015
[JWS]	JSON Web Signature (JWS): https://tools.ietf.org/html/rfc7797	Feb 2016
[JWT]	JSON Web Token (JWT): https://tools.ietf.org/html/rfc7519	May 2015
[MTLS]	OAuth 2.0 Mutual TLS Client Authentication and Certificate Bound Access Tokens: https://tools.ietf.org/html/rfc8705	Feb 2020
[TDIF]	Digital Transformation Agency - Trusted Digital Identity Framework https://www.dta.gov.au/our-projects/digital-identity/trusted-digital-identity-framework	Apr 2019
[RFC5322]	Internet Message Format: https://tools.ietf.org/html/rfc5322	Oct 2008
[RFC4627]	The application/json Media Type for JavaScript Object Notation (JSON): https://tools.ietf.org/html/rfc4627	Oct 2006
[RFC4648]	The Base16, Base32, and Base64 Data Encodings: https://tools.ietf.org/html/rfc4648	Oct 2006
[OAUTH2]	The OAuth 2.0 Authorization Framework: https://tools.ietf.org/html/rfc6749	Oct 2012
[OIDC]	OpenID Connect Core 1.0 incorporating errata set 1: http://openid.net/specs/openid-connect-core-1_0.html	Nov 2014
[OIDD]	OpenID Connect Discovery 1.0 incorporating errata d.net/specs/openid-connect-discovery-1_0.html	Nov 2014

Reference	Description	Version
[PAR]	OAuth 2.0 Pushed Authorization Requests: https://tools.ietf.org/html/draft-ietf-oauth-par-01	Feb 2020
[PKCE]	Proof Key for Code Exchange by OAuth Public Clients: https://datatracker.ietf.org/doc/html/rfc7636	Sep 2015
[RFC6750]	The OAuth 2.0 Authorization Framework: Bearer Token Usage: https://tools.ietf.org/html/rfc6750	Oct 2012
[RFC7009]	OAuth 2.0 Token Revocation: https://tools.ietf.org/html/rfc7009	Aug 2013
[RFC7523]	JSON Web Token (JWT) Profile for OAuth 2.0 Client Authentication and Authorization Grants: https://tools.ietf.org/html/rfc7523	May 2015
[RFC7662]	OAuth 2.0 Token Introspection: https://tools.ietf.org/html/rfc7662	Oct 2015
[RFC8414]	OAuth 2.0 Authorization Server Metadata: https://tools.ietf.org/html/rfc8414	Jun 2018
[RFC9126]	OAuth 2.0 Pushed Authorization Requests: https://tools.ietf.org/html/rfc9126	
[DCR]	OAuth 2.0 Dynamic Client Registration Protocol: https://datatracker.ietf.org/doc/html/rfc7591	Jul 2015
[FAPI-R-Draft]	Financial-grade API - Part 1: Read Only API Security Profile: https://openid.net/specs/openid-financial-api-part-1-ID2.html	Draft-06
[FAPI-RW-Draft]	Financial-grade API - Part 2: Read and Write API Security Profile: https://openid.net/specs/openid-financial-api-part-2-ID2.html	Draft-06
[FAPI-1.0-Baseline]	Financial-grade API Security Profile 1.0 - Part 1: Baseline: https://openid.net/specs/openid-financial-api-part-1-1_0.html	Mar 2021
[FAPI-1.0-Advanced]	Financial-grade API Security Profile 1.0 - Part 2: Advanced: https://openid.net/specs/openid-financial-api-part-2-1_0.html	Mar 2021
[JARM]	Financial-grade API: JWT Secured Authorization Response Mode for OAuth 2.0 (JARM): https://bitbucket.org/openid/fapi/src/master/Financial_API_JWT_Secured_Authorization_Response_Mode.md	Oct 2020
YAML 1.2	https://yaml.org/	Oct 2022
OpenAPI 3.1	OpenAPI Specification - Version 3.1.0 | Swagger	

12. ANNEX: Industry Abbreviations

Abbreviation / Acronym	Description
2-FA	Two-factor authentication
AIS	Account Information Services
AISP	Account Information Service Provider
ADW	Allowable Downtime Window
AML	Anti-Money Laundering
APP	Application
API	Application Programming Interface
ASP	Account Servicing Payment
ATM	Automated Teller Machine
BoN	Bank of Namibia
DP	Data Provider
ESI	Electronic Signature and Infrastructures
EFT	Electronic Fund Transfer
FAPI	Financial API
FI	Financial Institution
IPP	Instant Payment Programme
ISO	International Standards Organisation
KYC	Know Your Customer (Customer Due Diligence)
MFA	Multi Factor Authentication
MTLS	Mutual Transport Layer Security
MVP	Minimum Viable Product
NIST CRSC	National Institute of Standards and Technology Computer Security Resource Centre
NBFI	Non-Banking Financial Institution
OBF	Open Banking Forum
OIDC	Open IDConnect
OTP	One Time PIN
OWASP	Open Worldwide Application Security Project
P2P	Person-to-Person
PCI DSS	Payment Card Industry Data Security Standard
PIN	Personal Identification Number

PIS	Payment Information Services
PISP	Payment Initiation Service Provider
PKCE	Proof Key for Code Exchange
POS	Point of Sale
PSP	Payment Service Provider
PSU	Payment Services User
PUB	Public Data Services
QWAC	Qualified Website Authentication Certificate
SCA	Secure Customer Authentication
SLA	Service Level Agreement
SoV	Store of Value
SUB	Subscription Service
TLS	Transport Layer Security
TPP	Third Party Provider
TSP	Technical Service Provider
USSD	Unstructured Supplementary Service Data

13. ANNEX: Open Banking Glossary of Terms

The following glossary provides a description of key roles, terms, actors used throughout this document which are not already defined. These are standard across Open Banking environments.

Term	Definition	Also Known As
Scheme Level Actors	A Scheme is set of rules, standards, processes and infrastructure that creates a formal legal and operational model between multiple participants. It may be created through national law, scheme rules, or private contracts.	Trust Framework, ecosystem
Account Information Services (AIS)	Returns lists or details of bank accounts, account balances, account details, transactions and transaction details	
Payment Initiation Services (PIS)	Allow the creation and the cancellation of payments, and the ability to get the statuses of payments that have been created.	
Fourth party	Works on behalf of the TPP and is considered an outsourcing partner.	
Financial systems integrators	Data Providers will need and have accounting systems, payments systems, consent management systems, gateways to financial networks, firewalls, security systems. These systems are on the edge of the Open Finance journey and are often provided by a few known organisations.	
Technical terms		
Environment	An **Environment** is a collection of processes and programming tools that enables API Providers to build, test, and debug an API and API Users to view and use an API.	
Conformance System	A **Conformance system** is a collection of tools and services provided by an API Provider that allows API Users to safely test the	

	integration of an API with their own systems and services.	
Digital Certificates	Digital Certificates are credentials that can be machine verified by a trusted source. They are the digital equivalent of physical credentials such as passports, and driving licenses	Credentials
Participant Credentials	Digital certificates used for the purpose of identification and securing the API . Can be issued centrally by the scheme or by certificate authorities.	
Customer Support Terms		
Dispute	A situation where a payment event is challenged by a customer on their account. This could be potential fraud or unauthorised transactions. The dispute is typically raised by the transaction originator	
Chargeback	A charge returns after a customer successfully disputes an item on an already completed transaction.	

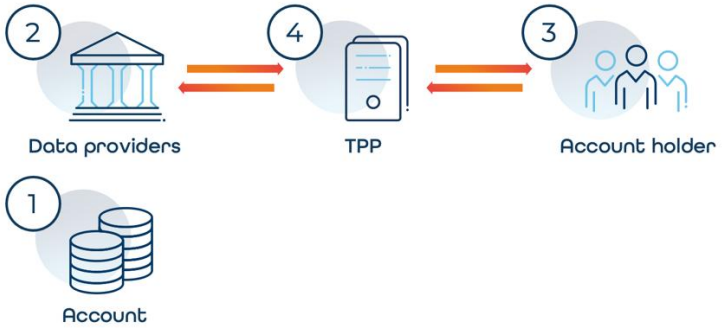
14. ANNEX: Key Topic: Account Definition

14.1 Defining Accounts, Account Providers and other parties

The word “account” causes problems, account means many things, and there are things which are not called accounts, which are!

The diagram below shows the basic elements. The Table underneath defines each element in terms of

- Open Data
- Open Finance
- Open Banking
- Payments
- OAuth protocol



#	Open Data	Open Finance	Open Banking	Payments	OAUTH
1	Account May or may not contain money	Account May or may not contain money	Account Bank Account e-Wallet Store of Value Always contains money.	Store of Value Always contains money.	Resource
2	Data Provider	Financial Institution	Bank Non-Bank Financial Institution e-money	Payment Service Provider Store of Value Provider	Resource Server
3	Account Holder	Account Holder	Account Holder Customer	Account Holder Customer Payer	Resource Owner

4	Third Party Provider TPP	Third Party Provider TPP	Third Party Provider TPP	Third Party Provider TPP	Client

14.2 Bank Accounts vs e-Wallets

14.2.1 Definitions

"Electronic money wallet" or "e-money wallet or e-wallet" means an application software or device on which monetary value is stored, and which allows the holder of the wallet to make electronic transactions. The wallet may be stored on the internet or devices such as a mobile phone or computer, or products such as prepaid cards. (PSD3).

N.B. The term "wallet" is also used in many different countries for stores of value such as a virtual store of virtual cards, that may include credit cards, store cards, or airline tickets (e.g. google wallet). These are out of scope.

14.2.2 Consumer Perspective

From a consumer perspective, anywhere they can put their money is an account. The public do not see a clear difference between a bank account and a wallet, especially as many banks provide accounts. Both are a store of value, both can receive payments, both can make payments, both provide balances on request.

14.2.3 Compliance & Operational Perspective

From a compliance and operational perspective, there are differences between wallets and accounts.

1. The Account Holder does not go through a KYC process.
2. The Data Provider does not (necessarily) know the name of the Account Holder.
3. The authentication mechanism for the wallet is tied to the phone.
4. The method for reading balances is through an Unstructured Supplementary Service Data (USSD) code on a feature phone.
5. There is no payment interoperability (today) between wallet providers, although this is changing.

15. ANNEX: Suggestions for future changes

This annex contains suggestions that were raised during the standards building process but have not been considered in version 1.0 due to them being a minority request. They may be considered in future versions.

15.1 New Sectors

These standards explicitly support banking but they have been written to be extended to other sectors. Insurance would be an obvious next choice, but healthcare, energy or telecommunications would also be relevant.

15.2 New Services

These standards support account information, payment initiation and consent flows.

Subscriptions. The standards do not support subscriptions, i.e. there is no API that allows a person to apply for a load or open an account.

Investment Contributions. We received feedback that an API to buy investments could also be welcomed within this service category.

Product information. We received feedback that an API to list bank products could be useful. As this is generally public information, there may be other ways of obtaining it, but a standardised API could be beneficial.

15.3 New Resource Objects (Banking)

15.3.1 Scheduled Payments

These standards allow Account Holders to create recurring payments through an API, i.e. Account holders can request that a payment be made every month to the same beneficiary for the same amount.

It is NOT possible for an Account Holder to view a list of future scheduled payments or to modify or delete future scheduled payments through the API, although the functionality will exist through the bank browser.

Arguably they should be able to perform this action, and the standards could be modified to extend this. On the other hand, this functionality would come at a cost and would have minority usage, so it may not be cost-effective. We note that in the UK and France APIs do not always support this functionality.

15.4 Extended Resource Objects

15.4.1 New Account Types

Support for Enterprise accounts with multiple authentications.

More complex information for specific account types, e.g. terms or interest rates for savings accounts.

15.4.2 New Payment Types

The supported payment types are shown in the table below.

The next obvious step would be to allow CMA or other cross border payments to be created.

15.4.3 New Fields in Transaction Information

Transaction Detail: More detailed information relating to a transaction, such as beneficiary address.

15.4.4 New Fields in Account Information

Limits: Limits include maximum volumes or values of transactions that can be made in a day, week or month. It also includes overdraft limits.