

**Coventry Building Society**

**Payment Initiation API Specification V2.1**

## Contents

Coventry Building Society .....	1
Version control .....	5
Release Note .....	5
Overview .....	5
Document Overview .....	6
Design Principles .....	6
RESTful APIs .....	6
Idempotency .....	6
Payment API - Scheme Agnostic .....	6
Status Codes .....	7
Scope .....	7
Out of Scope .....	7
Basics .....	7
Steps .....	8
Sequence Diagram .....	10
Actors .....	11
Character Encoding .....	11
Date Formats .....	11
Resource URI Path Structure .....	12
Payment Limits .....	12
Headers .....	12
Request Headers .....	12
Response Headers .....	13
Return & Error Codes .....	13
400 (Bad Request) v/s 404 (Not Found) .....	14
403 (Forbidden) .....	14
422 (Invalid) .....	15
429 (Too Many Requests) .....	15
Pre-Conditions .....	15
Pre-conditions for TPPs .....	15
Idempotency .....	15
Filtering .....	16
Pagination .....	16
Regulatory Considerations .....	16
PSD2 - Article 48 .....	16
Endpoints .....	16
POST /token .....	17

POST /open-banking/v2.0/payments.....	17
Payment Status .....	18
Error Information.....	18
GET /open-banking/v2.0/payments/{PaymentId} .....	19
Payment Status .....	19
Error Information.....	20
GET /authorize.....	21
POST /open-banking/v2.0/payment-submissions .....	22
Payment Submission Status .....	22
Error Information.....	22
GET /open-banking/v2.0/payment-submissions/{PaymentSubmissionId} .....	23
Payment Submission Status .....	23
Error Information.....	23
GET /open-banking/v2.0/payments/{PaymentId}/funds-confirmation.....	24
Error Information.....	24
POST /open-banking-extensions/v2.0/mandates .....	25
Payment Status .....	25
Error Information.....	26
GET /open-banking-extensions/v2.0/mandates/{PaymentId}.....	28
Payment Status .....	28
Error Information.....	28
POST /open-banking-extensions/v2.0/mandate-submissions.....	29
Payment Submission Status .....	30
Error Information.....	30
GET /open-banking-extensions/v2.0/mandate-submissions/{PaymentSubmissionId}.....	30
Payment Submission Status .....	31
Error Information.....	31
Sandbox API Endpoints .....	32
Security & Access Control.....	32
API Scopes .....	32
Grants Types .....	32
Consent Authorisation .....	33
Payment Status .....	33
Error Condition .....	33
Consent Revocation .....	33
Changes to Selected Account.....	33
Handling Expired Access Tokens.....	35
Data Model.....	35
High Level Payload Structure .....	35

Request Structure.....	35
Response Structure.....	36
Data Payload .....	36
Payment Setup - Request .....	36
Payment Setup - Response .....	40
Payment Submission - Request.....	44
Payment Submission - Response .....	48
Payments/{PaymentId}/funds-confirmation - Response.....	48
Mandates setup - Request .....	49
Mandates setup – Response.....	54
Mandate Submission - Request .....	59
Mandate Submission - Response .....	64
Data Payload – Error Response Structure .....	64
Example.....	64
Data Payload - Enumerations.....	65
Identifier Fields.....	67
Merchant Flow .....	68
Person to Person Flow .....	68
Mapping to Schemes & Standards .....	68
Transaction Status.....	68
Usage Examples .....	69
Merchant.....	69
Sequence Diagram.....	70
Illustrative Interactions .....	71
POST /payments request.....	71
POST /payment-submissions Request .....	72
GET /payments Request .....	73
GET /payment-submissions Request.....	75
GET /payments/{PaymentId}/funds-confirmation .....	75
POST /mandates Request .....	76
POST /mandate-submissions Request .....	78
GET /mandates Request.....	81
GET /payment-submissions Request.....	84
Alternative and Error Flows .....	85
Idempotent Payment Setup .....	85
Idempotent Payment Submission .....	86
Missing or Expired Access Token.....	87
Incomplete or Malformed Request Payload .....	88

## Version control

Version	Date	Updated by	Changes made
1.0	07 Feb 2018	Coventry Building Society	Baseline version
1.1	13 Feb 2019	Coventry Building Society	Addition of token endpoints
2.0	28 Feb 2019	Coventry Building Society	New error contract for all APIs Additional confirm funds endpoint Additional data models for mandates and mandate-submissions
2.1	16 Mar 2020	Coventry Building Society	Addition of Authorize endpoint and complete endpoint for all api resources. Addition of mandatory X-Client-Id header for all api resources.

### Release Note

This release note explains what's new in The Payment API Specifications between versions.

#### Version 2.0 – Changes

Our APIs follow OpenBanking:

- Payment API Specification v1.1 for immediate payments.
- For version 1.1, CBS created endpoints for mandates (future dated and regular) under open-banking-extensions as these had not been defined in the OpenBanking Payment API Specification v1.1.
- error response structure as defined in [OpenBanking Payment API Specification v3.1](#). These comprise of:
  - The error contract for each API has been modified to comply with the OpenBanking Read/Write Data API Specification v3.1. defined contract.
  - A new funds-confirmation endpoint added as per [OpenBanking Payment Intitiation API Specification\Domestic Payments v3.1](#)

#### Version 2.1 – Changes

- Add details of authorize endpoint
- Add new mandatory field X-Client-Id which must be sent with every API resource request

### Overview

This Payment Initiation API Specification describes the flows and payloads for initiating a single immediate payment, regular payment and future dated payment.

The API endpoints described here allow a PISP to:

- Register an intent to **setup** a payment instruction
- Subsequently **submit** the payment instruction for processing
- Optionally retrieve the status of a payment setup or submission.

## Document Overview

This document consists of the following parts:

**Overview:** Provides an overview of the scope of the API and the key decisions and principles that contributed to the specification.

**Basics:** The section begins with an introduction to how the API is used to initiate an immediate, regular and future dated payment. It goes on to identify the resources and operations that are permitted on those resources and various special cases.

**Security & Access Control:** Specifies the means for PISPs and PSUs to authenticate themselves and provide consent.

**Swagger Specifications:** Provides links to the swagger specifications for the APIs.

**Data Model:** Describes the data model for the API payloads.

**Usage Examples:** Examples for normal flows, and alternate flows.

## Design Principles

### RESTful APIs

The API adheres to RESTful API concepts where possible and sensible to do so.

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

References:

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

### Idempotency

POST operations on all the /payments endpoint are designed to be *idempotent*.

### Payment API - Scheme Agnostic

The API will be designed so that it is agnostic to the underlying payment scheme that is responsible for carrying out the payment.

In doing so - this means we will not design field lengths and payloads to only match the Faster Payments message, and will instead rely on the field lengths and definitions in ISO 20022. Due diligence has been carried out to ensure that the API has the necessary fields to function with Bacs payments - as per agreed scope.

## Status Codes

The API uses two status codes that serve two different purposes:

- The HTTP Status Code reflects the outcome of the API call (the HTTP operation on the resource).
- The Status field in the Payment API payloads reflect the status of the payments and payment-submissions resources. This Status will be limited to the ISO 20022 PaymentStatusCode code-list enumeration.

## Scope

The APIs in this document allow a PISP to initiate an immediate payment or set-up a mandate (future dated or regular) made in GBP.

## Out of Scope

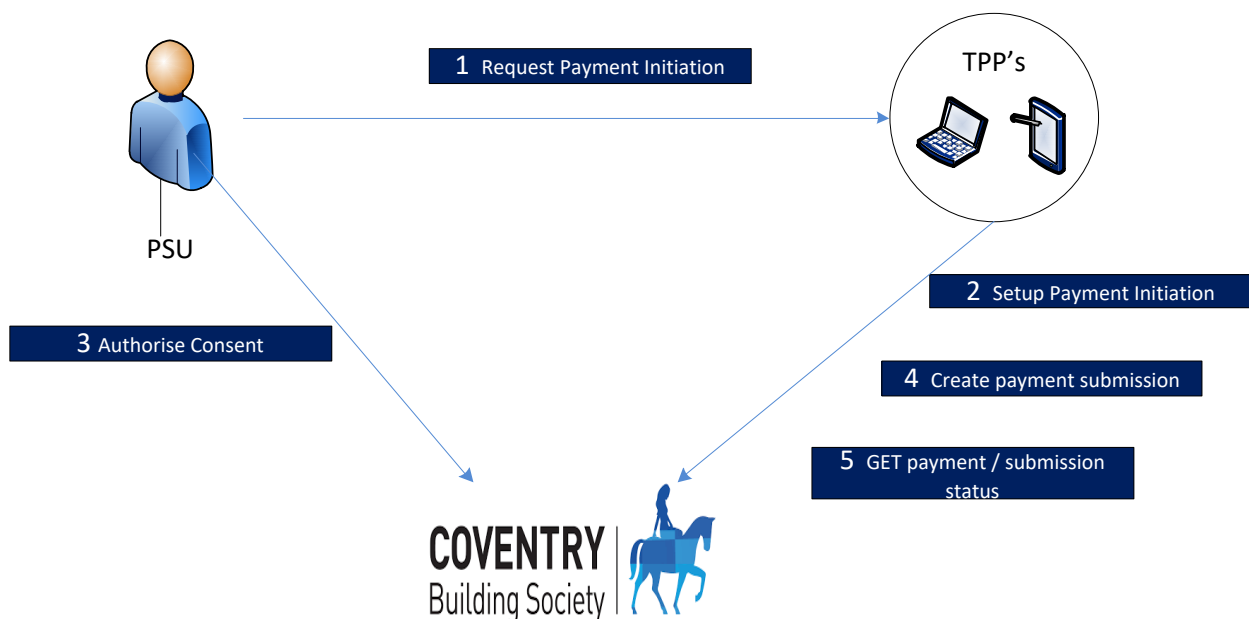
This v2.0 specification does not cater for:

- Payments that involve currency exchange.
- Payments that involve currencies other than GBP (no validation of EUR payment schemes has been completed for v2.0).

## Basics

### Overview

The figure below provides a general outline of a payment flow using the Payment APIs.



## Steps

### Step 1: Request Payment Initiation

- This flow begins with a PSU consenting to a payment being made. The request is sent through a PISP.
- The debtor account details can optionally be specified at this stage.

### Step 2: Setup Single Payment Initiation

- The PISP connects to CBS that services the PSU's payment account and creates a new **payments** resource. This informs CBS that one of its PSUs intends to make a payment. CBS responds with an identifier for the resource (the PaymentId - which is the intent identifier).
- This step is carried out by making a **POST** request to the **payments** resource.



### Step 3: Authorise Consent

- The PISP redirects the PSU to CBS. The redirect includes the PaymentId generated in the previous step. This allows CBS to correlate the payment that was setup. CBS authenticates the PSU. This could be an SCA if CBS determines that none of the SCA exemptions apply. CBS updates the state of the **payments** resource internally to indicate that the payment has been authorized.
- The PSU selects the debtor account at this stage.
- CBS compares the account submitted by PISP and the account selected by PSU and returns error if there is a mismatch.
- The PSU is redirected back to the PISP.

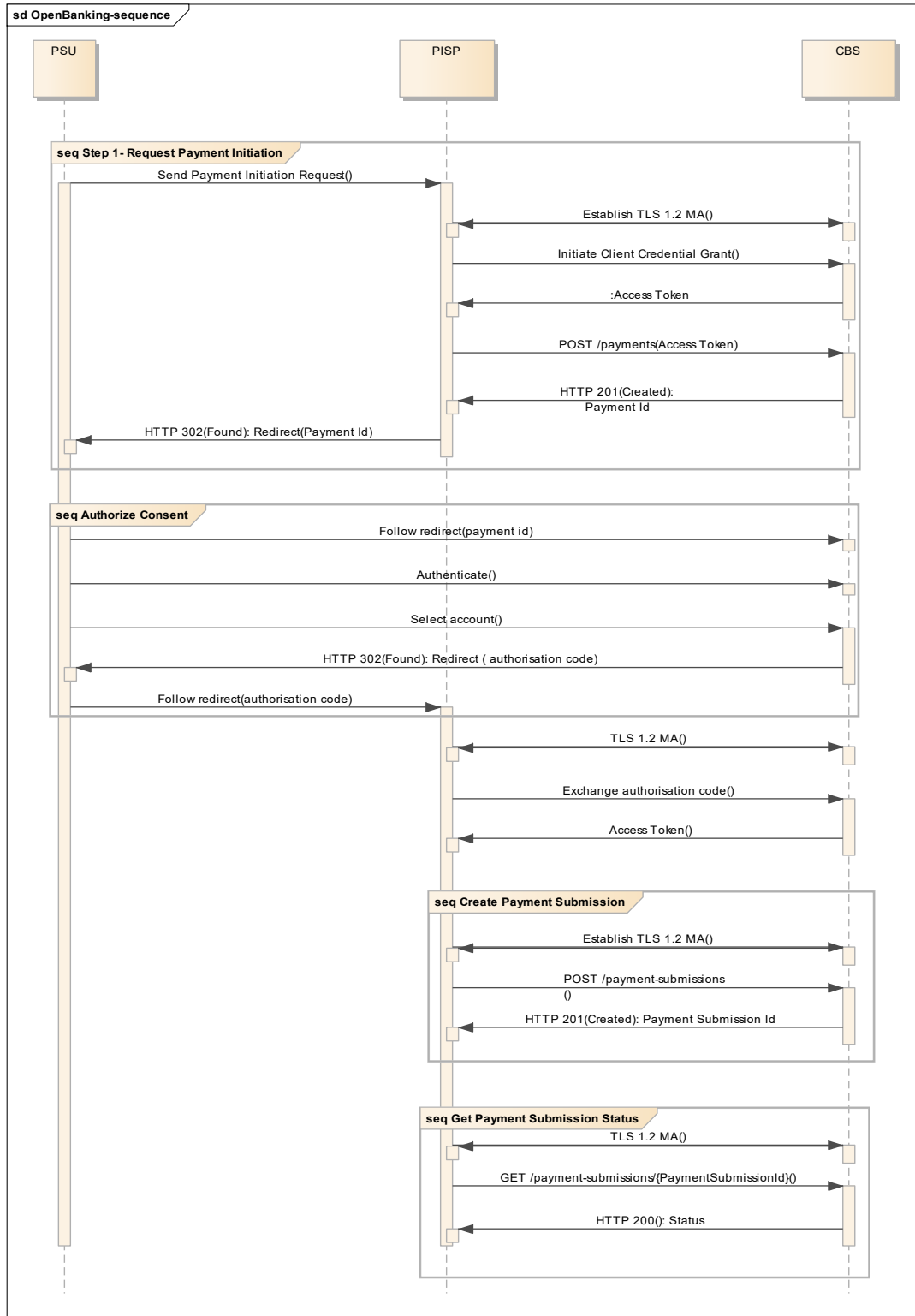
### Step 4: Create Payment Submission

- Once the PSU is redirected to the PISP, the PISP creates a **payment-submissions** resource to indicate that the payment created in the steps above should be submitted for processing.
- This is carried out by making a **POST** request to the **payment-submissions** resource.
- CBS returns the PaymentSubmissionId to the PISP.

### Step 5: Get Payment Submission Status

- This is carried out by making a **GET** request to the **payments** or **payment-submissions** resource.

# Sequence Diagram



## Actors

Actor	Abbreviation	Type	Specializes	Description
<b>Payment Service User</b>	PSU	Person	N/A	A natural or legal person making use of a payment service as a payee, payer or both (PSD2 Article 4(10))
<b>Payment Service Provider</b>	PSP	Legal Entity	N/A	A legal entity (and some natural persons) that provide payment services as defined by PSD2 Article 4(11)
<b>Third Party Providers / Trusted Third Parties</b>	TPP	Legal Entity	PSP	A party other than CBS that provides payment related services.  The term is not actually defined in PSD2, but is generally deemed to include all payment service providers that are 3rd parties (CBS and the PSU to whom the account belongs being the first two parties)
<b>Payment Initiation Service Provider</b>	PISP	Legal Entity	TPP	A TPP that provides Payment Initiation Services.  PSD2 does not offer a formal definition. Article 4(18) quite circularly defines a PISP as a PSP that provides Payment Initiation Services.
<b>Card Based Payment Instrument Issuer</b>	CBPII	Legal Entity	TPP	A TPP that provides Card Based Payment

## Character Encoding

The API requests and responses **must** use a UTF-8 character encoding. This is the default character encoding for JSON (RFC 7158 - [Section 8.1](#))

## Date Formats

All dates in the JSON payloads are represented in [ISO 8601](#) date-time format. All date-time fields in responses **must** include the timezone. An example is below:

2017-04-05T10:43:07+00:00

All dates in the HTTP headers are represented as [RFC 7231](#) Full Dates. An example is below:

Sun, 10 Sep 2017 19:43:31 UTC

JWT claims are expressed as a JSON number representing the number of seconds from 1970-01-01T0:0:0Z as measured in UTC until the date/time.

## Resource URI Path Structure

The resources defined by these APIs can be addressed through a path structure consisting of the following parts:

- The version of the APIs expressed as /v[major-version].[minor-version]/
- The resource name

Examples:

<https://resourcema.coventrybuildingsociety.co.uk/pd/open-banking/open-banking/v2.0/payments/{PaymentId}>

<https://resourcema.coventrybuildingsociety.co.uk/pd/open-banking/open-banking/v2.0/payment-submissions/{PaymentSubmissionId}>

## Payment Limits.

- The maximum amount allowed via this interface is £10000.
- The maximum allowed future date is 12 months.

## Headers

### Request Headers

Header Value	Notes	POST Requests	GET Requests
X-Client-Id	Should be set to the client id provided to the TPP when they on-boarded with CBS	Mandatory	Mandatory
x-fapi-financial-id	Should be set to "CBSOpenBanking"	Mandatory	Mandatory
x-fapi-customer-last-logged-time	The time when the PSU last logged in with the TPP.	Optional	Optional
x-fapi-customer-ip-address	The PSU's IP address if the PSU is currently logged in with the TPP.	Optional	Optional
x-fapi-interaction-id	An RFC4122 UID used as a correlation id.  If provided, CBS <b>will</b> "play back" this value in the x-fapi-interaction-id response header.	Optional	Optional
Authorization	Standard HTTP Header; Allows Credentials to be provided to the Authorisation / Resource Server depending on the type of resource being requested. For OAuth 2.0 / OIDC, this comprises of either the Basic / Bearer Authentication Schemes.	Mandatory	Mandatory
Content-Type	Standard HTTP Header; Represents the format of the payload being provided in the request.  This <b>must</b> be set to application/json.	Mandatory	Do not use
Accept	Standard HTTP Header; Determine the Content-Type that is required from the Server.  If set, it <b>must</b> have the value: application/json.	Optional	Optional

	If set to any other value, CBS will respond with a 406 Not Acceptable.		
	Optional.		
x-idempotency-key	Custom HTTP Header; Unique request identifier to support idempotency.  Mandatory for POST requests.	Mandatory	Do not use

(Reference: Section 6.3 - [Financial API — Part 1: Read Only API Security Profile \(Implementer's Draft\)](#).)

## Response Headers

Header Value	Notes	Mandatory ?
Content-Type	Standard HTTP Header; Represents the format of the payload returned in the response.  CBS will return Content-type: application/json as a content header.	Mandatory
x-fapi-interaction-id	An RFC4122 UID used as a correlation id.  This <b>will</b> be the same value provided in the x-fapi-interaction-id request header.	Conditionally Mandatory

## Return & Error Codes

The following are the HTTP response codes for the different HTTP methods - across all Payment API endpoints.

Situation	HTTP Status	Notes	Returned by POST	Returned by GET
Query completed successfully	200 OK		No	Yes
Normal execution. The request has succeeded.	201 Created	The operation results in the creation of a new resource.	Yes	No
Delete operation completed successfully	204 No Content		No	No
Request has malformed, missing or non-compliant JSON body or URL parameters	400 Bad Request	The requested operation will not be carried out.	Yes	No
Authorization header missing or invalid token	401 Unauthorized	The operation was refused access.  Re-authenticating the PSU or sending correct client id may result in an appropriate token that can be used.	Yes	Yes
Token has incorrect scope or a security policy was violated.	403 Forbidden	The operation was refused access.  Re-authenticating the PSU is unlikely to remediate the situation.	Yes	Yes
The TPP tried to access the resource with a method that is not supported.	405 Method Not Allowed		Yes	Yes

The request contained an accept header that requested a content-type other than application/json and a character set other than UTF-8	406 Not Acceptable		Yes	Yes
Schema errors	422 Invalid	The requested operation will not be carried out due to an error with the schema  The moreInformation section provides additional information as to where the schema validation has failed.	Yes	No
The operation was refused as too many requests have been made within a certain timeframe.	429 Too Many Requests	Throttling is a NFR.  CBS <b>may</b> include a Retry-After header in the response indicating how long the TPP must wait before retrying the operation.	Yes	Yes
Something went wrong on the API gateway or micro-service	500 Internal Server Error	The operation failed.	Yes	Yes

#### 400 (Bad Request) v/s 404 (Not Found)

When a TPP tries to request a resource URL with n resource Id that does not exist, CBS will respond with a 400 (Bad Request) rather than a 404 (Not Found).

E.g., if a TPP tries to GET /payments/22289 where 22289 is not a valid PaymentId, CBS will respond with a 400.

If the TPP tries to access a URL for a resource that is not defined by these specifications (e.g. GET /card-accounts), CBS **will** respond with a 404 (Not Found).

The table below illustrates some examples of expected behaviour:

Situation	Request	Response
TPP attempts to retrieve an account with a PaymentId that does not exist	GET /payments/1001	400 (Bad Request)
TPP attempts to retrieve a resource that is not defined	GET /bulk	404 (Not Found)

#### 403 (Forbidden)

When a TPP tries to access a resource that it does not have permission to access, CBS **will** return a 403 (Forbidden).

The situation could arise when:

- The TPP uses an access token that does not have the appropriate scope to access the requested resource.
- The TPP attempted to access a resource with an Id that it does not have access to. E.g., an attempt to access GET /payments/1001 where a payment resource with id 1001 belongs to another TPP.

When the TPP uses an access token that is no longer valid, the situation could potentially be remedied by asking the PSU to re-authenticate. This should be indicated by a 401 (Unauthorized) status code.

## 422 (Invalid)

When there are schema errors CBS will return a 422 (invalid) response.

The situation could arise when:

- The TPP includes an additional field in the request.
- The TPP uses an invalid data type
- The TPP has not included a required field

Example:

```
{
  "httpCode": "422",
  "httpMessage": "Invalid",
  "moreInformation": "Validate REST: xa35://tmp/temp_962878:1: [JSV0002] Invalid object: the property 'Code' is missing."
}
```

## 429 (Too Many Requests)

When a TPP tries to access a resource too frequently CBS may return a 429 (Too Many Requests). This situation could arise when:

- A TPP decides to implement "Real Time Payment Status" functionality for its users and implements this badly by polling a GET endpoint or an Idempotent POST endpoint **once-per-second** constantly to provide pseudo "real-time" Status updates to the user.
- A TPP decides to use the Single Immediate Payment endpoint as if it were a BATCH payment facility and sends 1,000 payment requests in a very short space of time.

## Pre-Conditions

The following pre-conditions must be satisfied in order to use these APIs:

### Pre-conditions for TPPs

1. The TPP must have completed onboarding with CBS and have been issued with a Client Id and Secret.
2. The software application registered with CBS must have "payments" as one of the required scopes.
3. The TPP must have valid eIDAS QWAC or OB WAC network certificate.

## Idempotency

The APIs for creating payment and payment-submission resources are idempotent. The intent of this capability is to allow PISP to retry API requests that failed with a timeout or an unexpected error.

The Idempotency key provided in the header **must** be at most 40 characters in size. If a larger idempotency key length is provided, CBS **will reject** the request with a status code is 400 (Bad Request).

The PISP **must not** change the request body while using the same Idempotency Key. If the PISP changes the request body, CBS will **not** modify the end resource. CBS **may** treat this as a fraudulent action.

CBS will treat a request as idempotent if it had received the first request with the same Idempotency Key from the same PISP in the preceding 24 hours.

CBS will **not** create a new resource for a POST request if it is determined to be an idempotent request.

CBS will respond to the request with the current status of the resource (or a status which is at least as current as what's available on existing online channels) and a HTTP status code of 201 (Created).

The PISP **must not** use the idempotent behaviour to poll the status of the payment resource or payment-submission resource.

## **Filtering**

The Payment APIs do not support filtering.

## **Pagination**

The Payment APIs do not support pagination.

## **Regulatory Considerations**

Non-normative guidance

This section provides non-normative guidance on how the specifications can be used to comply with certain requirements of PSD2 and the RTS. This is not an exhaustive list. Detailed analysis will be provided separately - with full traceability matrix of requirements.

Although this specification refers to the use of SCA, the use of SCA is not mandated until the RTS comes into effect.

The RTS is not finalised at the point of publishing this version of the specification - this may lead to some changes as new drafts of the RTS are released.

## **PSD2 - Article 48**

Immediately after receipt of the payment order, the payer's payment service provider shall provide the payer with or make available to the payer, all of the following data with regard to its own services:

- (a) a reference enabling the payer to identify the payment transaction and, where appropriate, information relating to the payee;
- (b) the amount of the payment transaction in the currency used in the payment order;
- (c) the amount of any charges for the payment transaction payable by the payer and, where applicable, a breakdown of the amounts of such charges;
- (d) where applicable, the exchange rate used in the payment transaction by the payer's payment service provider or a reference thereto, when different from the rate provided in accordance with point (d) of Article 45(1), and the amount of the payment transaction after that currency conversion;

CBS will address this requirement by providing this information to PSUs just after they have completed "Step 3: Authorize payment instruction", but before they are redirected back to the PISP.

## **Endpoints**

This section looks at the list of available API endpoints to complete a Payment flow. For detail



Endpoint design considerations:

- Having a separate resource for the payment **setup** and payment **submission** means we can extend the flows in the future more easily for bulk and recurring payments.
- Separation in setup and submission also allows for cleaner separation in updating the status of resources - for CBSs that chose to implement the functionality

Resource	HTTP Operation	End-point	Scope	Idempotent
token	POST	<a href="https://resourcema.coventrybuildingsociety.co.uk/mga/sps/oauth/oauth20/token">https://resourcema.coventrybuildingsociety.co.uk/mga/sps/oauth/oauth20/token</a>	payments	NA
payments	POST	<a href="https://resourcema.coventrybuildingsociety.co.uk/pd/open-banking/open-banking/v2.0/payments">https://resourcema.coventrybuildingsociety.co.uk/pd/open-banking/open-banking/v2.0/payments</a>	payments	Yes
payments	GET	<a href="https://resourcema.coventrybuildingsociety.co.uk/pd/open-banking/open-banking/v2.0/payments/{PaymentId}">https://resourcema.coventrybuildingsociety.co.uk/pd/open-banking/open-banking/v2.0/payments/{PaymentId}</a>	payments	NA
authorize	GET	<a href="https://resource.coventrybuildingsociety.co.uk/pd/open-banking/cbs/authorize?">https://resource.coventrybuildingsociety.co.uk/pd/open-banking/cbs/authorize?</a>	payments	NA
payment-submissions	POST	<a href="https://resourcema.coventrybuildingsociety.co.uk/pd/open-banking/open-banking/v2.0/payment-submissions">https://resourcema.coventrybuildingsociety.co.uk/pd/open-banking/open-banking/v2.0/payment-submissions</a>	payments	Yes
funds-confirmation	GET	<a href="https://resourcema.coventrybuildingsociety.co.uk/pd/open-banking/open-banking/v2.0/payments/{payentId}/funds-confirmation">https://resourcema.coventrybuildingsociety.co.uk/pd/open-banking/open-banking/v2.0/payments/{payentId}/funds-confirmation</a>	payments	NA
payment-submissions	GET	<a href="https://resourcema.coventrybuildingsociety.co.uk/pd/open-banking/open-banking/v2.0/payment-submissions/{PaymentSubmissionId}">https://resourcema.coventrybuildingsociety.co.uk/pd/open-banking/open-banking/v2.0/payment-submissions/{PaymentSubmissionId}</a>	payments	NA
mandates	POST	<a href="https://resourcema.coventrybuildingsociety.co.uk/pd/open-banking/open-banking-extensions/v2.0/mandates">https://resourcema.coventrybuildingsociety.co.uk/pd/open-banking/open-banking-extensions/v2.0/mandates</a>	payments	Yes
mandates	GET	<a href="https://resourcema.coventrybuildingsociety.co.uk/pd/open-banking/open-banking-extensions/v2.0/mandates/{PaymentId}">https://resourcema.coventrybuildingsociety.co.uk/pd/open-banking/open-banking-extensions/v2.0/mandates/{PaymentId}</a>	payments	NA
mandate-submissions	POST	<a href="https://resourcema.coventrybuildingsociety.co.uk/open-banking/pd/open-banking/open-banking-extensions/v2.0/mandate-submissions">https://resourcema.coventrybuildingsociety.co.uk/open-banking/pd/open-banking/open-banking-extensions/v2.0/mandate-submissions</a>	payments	Yes
mandate - submissions	GET	<a href="https://resourcema.coventrybuildingsociety.co.uk/pd/open-banking/open-banking-extensions/v2.0/mandate-submissions/{PaymentSubmissionId}">https://resourcema.coventrybuildingsociety.co.uk/pd/open-banking/open-banking-extensions/v2.0/mandate-submissions/{PaymentSubmissionId}</a>	payments	NA

**POST /token**

- This flow begins with a TPP requesting a Client Credentials Access Token from our token endpoint:

<https://resourcema.coventrybuildingsociety.co.uk/mga/sps/oauth/oauth20/token>

Request must include:

```
grant_type="client_credentials"
scope="openId payments"
client_id={clientId provided by CBS when TPP on-boarded}
client_secret={client secret provided by CBS when TPP on-boarded}
```

**POST /open-banking/v2.0/payments**

## Single Payment Setup Endpoint

POST /payments

The API allows the PISP to ask CBS to create a new **payment** resource.

- This indicates to CBS that a payment should be initiated. At this stage, the PSU may not have been identified by CBS, and the request payload may not contain any information of the account that should be debited.
- This API effectively allows the PISP to send a copy of the consent to CBS to authorise for this payment.
- CBS creates the payments resource and responds with a unique PaymentId to refer to the resource.

## Payment Status

The state model for the Status field is in the Mapping to Schemes & Standards section. The Status field for the Payment API follows the behaviour and definitions for the ISO 20022 PaymentStatusCode code-set.

The payment resource that is created successfully will have one of the following PaymentStatusCode code-set:

Status	Payment Status Description
1 Pending	Payment initiation or individual transaction included in the payment initiation is pending. Further checks and status update will be performed.
2 Rejected	Payment initiation or individual transaction included in the payment initiation has been rejected.
3 AcceptedTechnicalValidation	Authentication and syntactical and semantic validation are successful.

## Error Information

ErrorCode	Message	Notes
1000	We're unable to complete this request due to an issue with the consent details received	The scope for the software application does not include payments which is required to access these APIs
1005	We're unable to complete this request due to an issue with the consent details received	The consent details are missing from the request
1006	We're unable to complete this request as it exceeds the maximum payment limit allowed	Payment amount exceeds OLS limit
1007	We're unable to complete this request due to an issue with the	There is a problem with the details we hold about the organisation.

	Organisation details provided	
1008	We're unable to complete this request due to an issue with the Software Application details provided	There is a problem with the software application.
1009	We're unable to complete this request as it is below the minimum payment limit allowed	Payment amount below minimum
1014	We're unable to complete this request due to a technical issue	There was a technical problem which occurred within CBS.
99997	We're unable to complete this request due to an Invalid Header Check	The headers in the request were incomplete
99998 99999	We're unable to complete this request due to a technical issue	There was a technical problem which occurred within CBS.

**GET /open-banking/v2.0/payments/{PaymentId}**

### Single Payment Status Endpoint

GET /payments/{PaymentId}

A PISP can optionally retrieve a **payment** resource that they have created to check its status.

### Payment Status

Once the PSU authorises the payment resource - the Status of the payment resource will be updated with AcceptedCustomerProfile.

The available PaymentStatusCode code-set enumerations for the payment resource are:

	Status	Payment Status Description
1	Pending	Payment initiation or individual transaction included in the payment initiation is pending. Further checks and status update will be performed.
2	Rejected	Payment initiation or individual transaction included in the payment initiation has been rejected.

<b>3</b>	AcceptedTechnicalValidation	Authentication and syntactical and semantic validation are successful.
<b>4</b>	AcceptedCustomerProfile	Preceding check of technical validation was successful. Customer profile check was also successful.

## Error Information

<b>ErrorCode</b>	<b>Message</b>	<b>Notes</b>
1001	We're unable to complete this request due to an issue with the consent details received	Problem retrieving consent id
1002	We're unable to complete this request due to an issue with the consent details received	
1003	We're unable to complete this request due to an issue with the consent details received	
1004	We're unable to complete this request due to an issue with the consent details received	
1005	We're unable to complete this request due to an issue with the payment information received	
1006	We're unable to complete this request due to an issue with the consent details received	
1007	We're unable to complete this request due to an issue with the payment information received	
1008	We're unable to complete this request due to an issue with the Organisation details provided	
99997	We're unable to complete this request due to an Invalid Header Check	The headers in the request were incomplete
99998 99999	We're unable to complete this request due to a technical issue	There was a technical problem which occurred within CBS.



## POST /open-banking/v2.0/payment-submissions

### Single Payment Submission Endpoint

POST /payment-submissions/

Once the payment has been authorised by the PSU, the PISP can proceed to submitting the payment for processing:

- This is done by making a POST request to the **payment-submissions** resource.
- This request is an instruction to CBS to begin the single immediate payment journey. The payment will be submitted immediately, however, there are some scenarios where the payment may not happen immediately (e.g. busy periods at CBS).
- The PISP **must** ensure that the Initiation and Risk sections of the payment submission match the corresponding Initiation and Risk sections of the original payment resource. If the two do not match, CBS **will not** process the request and **must** respond with a 400 (Bad Request).
- Any operations on the payment-submission resource will not result in a Status change for the payment resource.

### Payment Submission Status

A payment-submission can only be created if its corresponding payment resource has the status of 'AcceptedCustomerProfile'.

The payment-submission resource that is created successfully must have one of the following PaymentStatusCode code-set enumerations:

Status	Payment Status Description
1 Pending	Payment initiation or individual transaction included in the payment initiation is pending. Further checks and status update will be performed.
2 Rejected	Payment initiation or individual transaction included in the payment initiation has been rejected.
3 AcceptedSettlementInProgress	All preceding checks such as technical validation and customer profile were successful and therefore the payment initiation has been accepted for execution.

### Error Information

ErrorCode	Message	Notes
1001	We're unable to complete this request due to an issue with the consent details received	Problem retrieving consent id
99997	We're unable to complete this request due to an Invalid Header Check	The headers in the request were incomplete

99998	We're unable to complete this request	There was a technical problem which occurred within CBS.
99999	due to a technical issue	

## GET /open-banking/v2.0/payment-submissions/{PaymentSubmissionId}

### Single Payment Submission Status Endpoint

GET/payment-submissions/{PaymentSubmissionId}

A PISP can retrieve the payment-submission to check its status.

### Payment Submission Status

The payment-submission resource must have one of the following PaymentStatusCode code-set enumerations:

Status	Payment Status Description
1 Pending	Payment initiation or individual transaction included in the payment initiation is pending. Further checks and status update will be performed.
2 Rejected	Payment initiation or individual transaction included in the payment initiation has been rejected.
3 AcceptedSettlementInProgress	All preceding checks such as technical validation and customer profile were successful and therefore the payment initiation has been accepted for execution.
4 AcceptedSettlementCompleted	Settlement on the debtor's account has been completed.

### Error Information

ErrorCode	Message	Notes
1001	We're unable to complete this request due to an issue with the consent details received	Problem retrieving consent id
1002	We're unable to complete this request due to an issue with the consent details received	
1003	We're unable to complete this request due to an issue with the consent details received	
1004	We're unable to complete this request	

	due to an issue with the consent details received	
1005	We're unable to complete this request due to an issue with the payment information received	
1006	We're unable to complete this request due to an issue with the consent details received	
1007	We're unable to complete this request due to an issue with the payment information received	
1008	We're unable to complete this request due to an issue with the Organisation details provided	
99997	We're unable to complete this request due to an Invalid Header Check	The headers in the request were incomplete
99998 99999	We're unable to complete this request due to a technical issue	There was a technical problem which occurred within CBS.

### GET /open-banking/v2.0/payments/{PaymentId}/funds-confirmation

A PISP can confirm funds on a payment consent.

#### Error Information

ErrorCode	Message	Notes
1000	We're unable to complete this request due to an issue with the consent details received	



1001	We're unable to complete this request due to an issue with the consent details received	
1002	We're unable to complete this request due to an issue with the consent details received	The consent id has expired
1003	We're unable to complete this request due to an issue with the consent details received	The consent id is invalid
99997	We're unable to complete this request due to an Invalid Header Check	The headers in the request were incomplete
99998 99999	We're unable to complete this request due to a technical issue	There was a technical problem which occurred within CBS.

## POST /open-banking-extensions/v2.0/mandates

### Regular, Future Dated Payment Setup Endpoint

POST /mandates

The API allows the PISP to ask CBS to create a new regular or future dated **payment** resource.

- This indicates to CBS that a payment should be initiated. At this stage, the PSU may not have been identified by CBS, and the request payload may not contain any information of the account that should be debited.
- This API effectively allows the PISP to send a copy of the consent to CBS to authorise for this payment.
- CBS creates the payments resource and responds with a unique PaymentId to refer to the resource.

### Payment Status

The state model for the Status field is in the Mapping to Schemes & Standards section. The Status field for the Payment API follows the behaviour and definitions for the ISO 20022 PaymentStatusCode code-set.

The payment resource that is created successfully will have one of the following PaymentStatusCode code-set:

Status	Payment Status Description
1 Pending	Payment initiation or individual transaction included in the payment initiation is pending. Further checks and status update will be performed.
2 Rejected	Payment initiation or individual transaction included in the payment initiation has been rejected.
3 AcceptedTechnicalValidation	Authentication and syntactical and semantic validation are successful.

## Error Information

ErrorCode	Message	Notes
1000	We're unable to complete this request due to an issue with the consent details received	The scope for the software application does not include payments which is required to access these APIs
1005	We're unable to complete this request due to an issue with the consent details received	The consent details are missing from the request
1006	We're unable to complete this request as it exceeds the maximum payment limit allowed	Payment amount exceeds OLS limit
1007	We're unable to complete this request due to an issue with the Organisation details provided	There is a problem with the details we hold about the organisation.
1008	We're unable to complete this request due to an issue with the Software Application details provided	There is a problem with the software application.
1009	We're unable to complete this request as it is below the minimum payment limit allowed	Payment amount below minimum
1010	We're unable to complete this request as the payment date must be within the next 12 months	
1011	We're unable to complete this request as the first payment date	

	must be within the next 12 month	
1012	We're unable to complete this request as the regular payment date must be between the first and last payment dates	
1013	We're unable to complete this request as the final payment date must be after the first and regular payment dates	
1014 1015 1016 1017	We're unable to complete this request due to a technical issue	There was a technical problem which occurred within CBS.
99997	We're unable to complete this request due to an Invalid Header Check	The headers in the request were incomplete
99998 99999	We're unable to complete this request due to a technical issue	There was a technical problem which occurred within CBS.

**GET /open-banking-extensions/v2.0/mandates/{PaymentId}**

**Future Date / Regular Payment Status Endpoint**

GET /mandates/{PaymentId}

A PISP can optionally retrieve a **payment** resource that they have created to check its status.

**Payment Status**

Once the PSU authorises the payment resource - the Status of the payment resource will be updated with AcceptedCustomerProfile.

The available PaymentStatusCode code-set enumerations for the payment resource are:

Status		Payment Status Description
1	Pending	Payment initiation or individual transaction included in the payment initiation is pending. Further checks and status update will be performed.
2	Rejected	Payment initiation or individual transaction included in the payment initiation has been rejected.
3	AcceptedTechnicalValidation	Authentication and syntactical and semantic validation are successful.
4	AcceptedCustomerProfile	Preceding check of technical validation was successful. Customer profile check was also successful.

**Error Information**

ErrorCode	Message	Notes
1001	We're unable to complete this request due to an issue with the consent details received	Problem retrieving consent id
1002	We're unable to complete this request due to an issue with the consent details received	
1003	We're unable to complete this request due to an issue with the consent details received	
1004	We're unable to complete this request due to an issue with the consent details received	
1005	We're unable to complete this request due to an issue with the payment information received	
1006	We're unable to complete this request	

	due to an issue with the consent details received	
1007	We're unable to complete this request due to an issue with the payment information received	
1008	We're unable to complete this request due to an issue with the Organisation details provided	
99997	We're unable to complete this request due to an Invalid Header Check	The headers in the request were incomplete
99998 99999	We're unable to complete this request due to a technical issue	There was a technical problem which occurred within CBS.

## POST /open-banking-extensions/v2.0/mandate-submissions

### Future Dated/ Regular Payment Submission Endpoint

POST /mandate-submissions/

Once the payment has been authorised by the PSU, the PISP can proceed to submitting the payment for processing:

- This is done by making a POST request to the **mandate-submissions** resource.
- This request is an instruction to CBS to begin the Future date or regular payment journey. The payment will be submitted immediately and executed on the date specified in the payment consent.
- The PISP **must** ensure that the Initiation and Risk sections of the payment submission match the corresponding Initiation and Risk sections of the original payment resource. If the two do not match, CBS **will not** process the request and **will** respond with a 400 (Bad Request).
- Any operations on the cbs-payment-submission resource will not result in a Status change for the payment resource.

## Payment Submission Status

A payment-submission can only be created if its corresponding payment resource has the status of 'AcceptedCustomerProfile'.

The payment-submission resource that is created successfully must have one of the following PaymentStatusCode code-set enumerations:

Status	Payment Status Description
1 Pending	Payment initiation or individual transaction included in the payment initiation is pending. Further checks and status update will be performed.
2 Rejected	Payment initiation or individual transaction included in the payment initiation has been rejected.
3 AcceptedSettlementInProgress	All preceding checks such as technical validation and customer profile were successful and therefore the payment initiation has been accepted for execution.

## Error Information

ErrorCode	Message	Notes
1001	We're unable to complete this request due to an issue with the consent details received	Problem retrieving consent id
99997	We're unable to complete this request due to an Invalid Header Check	The headers in the request were incomplete
99998 99999	We're unable to complete this request due to a technical issue	There was a technical problem which occurred within CBS.

**GET /open-banking-extensions/v2.0/mandate-submissions/{PaymentSubmissionId}**

### Future Dated/ Regular Payment Submission Status Endpoint

GET/mandate-submissions/{PaymentSubmissionId}

A PISP can retrieve the payment-submission to check its status.

## Payment Submission Status

The payment-submission resource must have one of the following PaymentStatusCode code-set enumerations:

Status		Payment Status Description
1	Pending	Payment initiation or individual transaction included in the payment initiation is pending. Further checks and status update will be performed.
2	Rejected	Payment initiation or individual transaction included in the payment initiation has been rejected.
3	AcceptedSettlementInProgress	All preceding checks such as technical validation and customer profile were successful and therefore the payment initiation has been accepted for execution.
4	AcceptedSettlementCompleted	Settlement on the debtor's account has been completed.

## Error Information

ErrorCode	Message	Notes
1001	We're unable to complete this request due to an issue with the consent details received	Problem retrieving consent id
1002	We're unable to complete this request due to an issue with the consent details received	
1003	We're unable to complete this request due to an issue with the consent details received	
1004	We're unable to complete this request due to an issue with the consent details received	
1005	We're unable to complete this request due to an issue with the payment information received	
1006	We're unable to complete this request	

	due to an issue with the consent details received	
1007	We're unable to complete this request due to an issue with the payment information received	
1008	We're unable to complete this request due to an issue with the Organisation details provided	
99997	We're unable to complete this request due to an Invalid Header Check	The headers in the request were incomplete
99998 99999	We're unable to complete this request due to a technical issue	There was a technical problem which occurred within CBS.

## Sandbox API Endpoints

Sandbox APIs have been provided to assist with testing connectivity. The APIs all return mocked data; they are only accessible using TLS 1.2 MA with a valid eIDAS QWAC or OB WAC network (uses Client Credentials Grant only)

Please refer to the separate Coventry SandBox API Specification for further details.

## Security & Access Control

### API Scopes

The access tokens required for accessing the Payment APIs must have at least the following scope:

**Scopes**  
payments

### Grants Types

PISPs **must** use a client credentials grant to obtain a token to make POST requests to the payments resource.

PISPs **must** use an authorization code grant to obtain a token to make POST requests to the payment-submissions resource.



PISPs **must** use a client credentials grant to make GET requests.

## Consent Authorisation

OAuth 2.0 scopes are coarse grained and the set of available scopes are defined at the point of client registration. There is no standard method for specifying and enforcing fine grained scopes (e.g. a scope to enforce payments of a specified amount on a specified date).

A *consent authorisation* is used to define the fine-grained scope that is granted by the PSU to the PISP.

The PISP **must** begin a single immediate payment request by creating a **payments** resource through a **POST** operation. This resource indicates the *consent* that the PISP claims it has been given by the PSU. At this stage, the consent is not yet authorised as CBS has not yet verified this claim with the PSU.

CBS responds with a PaymentId. This is the intent-id that is used when initiating the authorization code grant.

As part of the authorization code grant:

- CBS authenticates the PSU.
- CBS plays back the consent (registered by the PISP) back to the PSU - to get consent authorisation. The PSU may accept or reject the consent in its entirety (but not selectively).
- CBS presents the PSU a list of accounts from which the PSU may select one. This should match the debtor account presented by PISP while registering the consent with CBS else CBS will raise an exception.

Once these steps are complete, the consent is considered to have been authorised by the PSU.

## Payment Status

The Payment resource can have one of the following ISO status codes after authorisation has taken place:

	Payment Status Code	Payment Status Description
1	AcceptedCustomerProfile	Preceding check of technical validation was successful. Customer profile check was also successful.
2	Rejected	Payment initiation or individual transaction included in the payment initiation has been rejected.

## Error Condition

If the PSU does not complete a successful consent authorisation (e.g. if the PSU is not authenticated successfully), the authorization code grant ends with a redirection to the TPP with an error response as described in [RFC 6749 Section 4.1.2.1](#). The PSU is redirected to the TPP with a error parameter indicating the error that occurred.

## Consent Revocation

A PSU cannot revoke consent for a single immediate payment - once it has been authorized.

This is required to comply with Article 80 of PSD2.

## Changes to Selected Account

For a single immediate payment, the selected debtor account cannot be changed once the consent has been authorized.

## Handling Expired Access Tokens

### *Access Token issued through Client Credentials Grant*

When an access token issued through a Client Credentials Grant expires, the TPP must get a new access token by executing a client credential grant again.

### *Access Token issued through Authorization Code Grant*

CBS will issue an access token at the end of an authorisation code grant. This token will expire after 3600 seconds (1 hour), after which a new access token will be required using a new paymentId.

## Data Model

### High Level Payload Structure

This section gives an overview of the top level structure for the API payloads for the Payment APIs.

The Data and Risk sections of the payload structure are documented in Data Dictionary section; while the Links and Meta are standardised - which are explained in the Response Structure.

### Request Structure

The top level request structure for Payment APIs:

#### Payment API Request

```
{
  "Data": {
    "Initiation": {
      ...
    }
  },
  "Risk": {
    ...
  }
}
```

The top level structure for the Payment API POST requests will be:

- Data
  - Initiation
- Risk

The Data section contains the payment initiation object.

A separate Initiation section within the Data section gives us the flexibility to extend and modify the Initiation section in isolation.

A Risk section for the request structure has been separated out - so that this can evolve in isolation from the Initiate section of the payload.

## Response Structure

The top level response structure for Payment APIs:

### Payment API Response

```
{
  "Data": {
    ...
    "Initiation": {
      ...
    }
  },
  "Risk": {
    ...
  },
  "Links": {
    ...
  },
  "Meta": {
    ...
  }
}
```

In line with the principle on RESTful API practices - we are replaying the full resource as part of the response.

Two additional top level sections are included in the response for:

- Links
- Meta

## Data Payload

The data dictionary section gives the detail on the payload content for the Payment API flows.

### Payment Setup - Request

The CBSPaymentSetup1 object will be used for the call to:

- POST /payments

Name	Card-inality	XPath	EnhancedDefinition
CBSPaymentSetup1		CBSPaymentSetup1	
Data	1..1	CBSPaymentSetup1/Data	
Initiation	1..1	CBSPaymentSetup1/Data/Initiation	The Initiation payload is sent by the initiating party to the CBS. It is used to request movement of funds from the debtor account to a creditor.

InstructionIdentification	1..1	CBSPaymentSetup1/Data/Initiation/InstructionIdentification	<p>Unique identification as assigned by an instructing party for an instructed party to unambiguously identify the instruction.</p> <p>Usage: the instruction identification is a point to point reference that can be used between the instructing party and the instructed party to refer to the individual instruction. It can be included in several messages related to the instruction.</p>
EndToEndIdentification	1..1	CBSPaymentSetup1/Data/Initiation/EndToEndIdentification	<p>Unique identification assigned by the initiating party to unambiguously identify the transaction. This identification is passed on, unchanged, throughout the entire end-to-end chain.</p> <p>Usage: The end-to-end identification can be used for reconciliation or to link tasks relating to the transaction. It can be included in several messages related to the transaction. The Faster Payments Scheme can only access 31 characters for the EndToEndIdentification field.</p>
InstructedAmount	1..1	CBSPaymentSetup1/Data/Initiation/InstructedAmount	<p>Amount of money to be moved between the debtor and creditor, before deduction of charges, expressed in the currency as ordered by the initiating party.</p> <p>Usage: This amount has to be transported unchanged through the transaction chain.</p>
Currency	1..1	CBSPaymentSetup1/Data/Initiation/InstructedAmount/Currency	<p>A code allocated to a currency by a Maintenance Agency under an international identification scheme, as described in the latest edition of the international standard ISO 4217 "Codes for the representation of currencies and funds".</p>
DebtorAgent	0..1	CBSPaymentSetup1/Data/Initiation/DebtorAgent	<p>Financial institution servicing an account for the debtor.</p>
SchemeName	1..1	CBSPaymentSetup1/Data/Initiation/DebtorAgent/SchemeName	<p>Name of the identification scheme, in a coded form as published in an external list.</p>
Identification	1..1	CBSPaymentSetup1/Data/Initiation/DebtorAgent/Identification	<p>Unique and unambiguous identification of a person.</p>
DebtorAccount	0..1	CBSPaymentSetup1/Data/Initiation/DebtorAccount	<p>Unambiguous identification of the account of the debtor to which a debit entry will be made as a result of the transaction.</p>
SchemeName	1..1	CBSPaymentSetup1/Data/Initiation/DebtorAccount/SchemeName	<p>Name of the identification scheme, in a coded form as published in an external list.</p>
Identification	1..1	CBSPaymentSetup1/Data/Initiation/DebtorAccount/Identification	<p>Identification assigned by an institution to identify an account. This identification is known by the account owner.</p>
Name	0..1	CBSPaymentSetup1/Data/Initiation/DebtorAccount/Name	<p>Name of the account, as assigned by the account servicing institution, in agreement with the</p>

			<p>account owner in order to provide an additional means of identification of the account.</p> <p>Usage: The account name is different from the account owner name. The account name is used in certain user communities to provide a means of identifying the account, in addition to the account owner's identity and the account number.</p>
SecondaryIdentification	0..1	CBSPaymentSetup1/Data/Initiation/DebtorAccount/SecondaryIdentification	This is secondary identification of the account, as assigned by the account servicing institution. This can be used by building societies to additionally identify accounts with a roll number (in addition to a sort code and account number combination).
CreditorAgent	0..1	CBSPaymentSetup1/Data/Initiation/CreditorAgent	Financial institution servicing an account for the creditor.
SchemeName	1..1	CBSPaymentSetup1/Data/Initiation/CreditorAgent/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSPaymentSetup1/Data/Initiation/CreditorAgent/Identification	Unique and unambiguous identification of a person.
CreditorAccount	1..1	CBSPaymentSetup1/Data/Initiation/CreditorAccount	Unambiguous identification of the account of the creditor to which a credit entry will be posted as a result of the payment transaction.
SchemeName	1..1	CBSPaymentSetup1/Data/Initiation/CreditorAccount/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSPaymentSetup1/Data/Initiation/CreditorAccount/Identification	Identification assigned by an institution to identify an account. This identification is known by the account owner.
Name	1..1	CBSPaymentSetup1/Data/Initiation/CreditorAccount/Name	<p>Name of the account, as assigned by the account servicing institution, in agreement with the account owner in order to provide an additional means of identification of the account.</p> <p>Usage: The account name is different from the account owner name. The account name is used in certain user communities to provide a means of identifying the account, in addition to the account owner's identity and the account number.</p> <p>ASPSPs <b>may</b> carry out name validation for Confirmation of Payee, but it is not mandatory.</p>
SecondaryIdentification	0..1	CBSPaymentSetup1/Data/Initiation/CreditorAccount/SecondaryIdentification	This is secondary identification of the account, as assigned by the account servicing institution. This can be used by building societies to additionally identify accounts with a roll number (in addition to a sort code and account number combination).
RemittanceInformation	0..1	CBSPaymentSetup1/Data/Initiation/RemittanceInformation	Information supplied to enable the matching of an entry with the items that the transfer is intended to settle, such as commercial invoices in an accounts' receivable system.
Unstructured	0..1	CBSPaymentSetup1/Data/Initiation/RemittanceInformation/Unstructured	Information supplied to enable the matching/reconciliation of an entry with the items that the payment is intended to settle, such as

			commercial invoices in an accounts' receivable system, in an unstructured form.
Reference	0..1	CBSPaymentSetup1/Data/Initiation/RemittanceInformation/Reference	<p>Unique reference, as assigned by the creditor, to unambiguously refer to the payment transaction.</p> <p>Usage: If available, the initiating party should provide this reference in the structured remittance information, to enable reconciliation by the creditor upon receipt of the amount of money.</p> <p>If the business context requires the use of a creditor reference or a payment remit identification, and only one identifier can be passed through the end-to-end chain, the creditor's reference or payment remittance identification should be quoted in the end-to-end transaction identification.</p> <p>The Faster Payments Scheme can only accept 18 characters for the ReferenceInformation field - which is where this ISO field will be mapped.</p>
Risk	1..1	CBSPaymentSetup1/Risk	The Risk section is sent by the initiating party to the CBS. It is used to specify additional details for risk scoring for Payments.
PaymentContextCode	0..1	CBSPaymentSetup1/Risk/PaymentContextCode	Specifies the payment context
MerchantCategoryCode	0..1	CBSPaymentSetup1/Risk/MerchantCategoryCode	Category code conform to ISO 18245, related to the type of services or goods the merchant provides for the transaction.
MerchantCustomerIdentification	0..1	CBSPaymentSetup1/Risk/MerchantCustomerIdentification	The unique customer identifier of the PSU with the merchant.
DeliveryAddress	0..1	CBSPaymentSetup1/Risk/DeliveryAddress	Information that locates and identifies a specific address, as defined by postal services or in free format text.
AddressLine	0..2	CBSPaymentSetup1/Risk/DeliveryAddress/AddressLine	Information that locates and identifies a specific address, as defined by postal services that is presented in free format text.
StreetName	0..1	CBSPaymentSetup1/Risk/DeliveryAddress/StreetName	Name of a street or thoroughfare.
BuildingNumber	0..1	CBSPaymentSetup1/Risk/DeliveryAddress/BuildingNumber	Number that identifies the position of a building on a street.
PostCode	0..1	CBSPaymentSetup1/Risk/DeliveryAddress/PostCode	Identifier consisting of a group of letters and/or numbers that is added to a postal address to assist the sorting of mail.
TownName	1..1	CBSPaymentSetup1/Risk/DeliveryAddress/TownName	Name of a built-up area, with defined boundaries, and a local government.
CountrySubDivision	0..2	CBSPaymentSetup1/Risk/DeliveryAddress/CountrySubDivision	Identifies a subdivision of a country, for instance state, region, county.
Country	1..1	CBSPaymentSetup1/Risk/DeliveryAddress/Country	Nation with its own government, occupying a particular territory.

## Payment Setup - Response

The CBSPaymentSetupResponse1 object will be used for a response to a call to:

- POST /payments
- GET /payments/{PaymentId}

### Notes

The Payment Setup **response** contains the full **original** payload from the Payment Setup POST **request** - with these additional elements:

- PaymentId.
- Status of the Payment resource.
- Date time the Payment resource was created.

### Data Dictionary

Name	Card-inality	XPath	EnhancedDefinition
CBSPaymentSetupResponse1		CBSPaymentSetupResponse1	
Data	1..1	CBSPaymentSetupResponse1/Data	
PaymentId	1..1	CBSPaymentSetupResponse1/Data/PaymentId	Unique identification as assigned by CBS to uniquely identify the payment setup resource.
Status	0..1	CBSPaymentSetupResponse1/Data/Status	Specifies the status of the payment resource.
CreationDateTime	1..1	CBSPaymentSetupResponse1/Data/CreationDateTime	Date and time at which the resource was created.
Initiation	1..1	CBSPaymentSetupResponse1/Data/Initiation	The Initiation payload is sent by the initiating party to CBS. It is used to request movement of funds from the debtor account to a creditor.
InstructionIdentification	1..1	CBSPaymentSetupResponse1/Data/Initiation/InstructionIdentification	Unique identification as assigned by an instructing party for an instructed party to unambiguously identify the instruction.  Usage: the instruction identification is a point to point reference that can be used between the instructing party and the instructed party to refer to the individual instruction. It can be included in several messages related to the instruction.
EndToEndIdentification	1..1	CBSPaymentSetupResponse1/Data/Initiation/EndToEndIdentification	Unique identification assigned by the initiating party to unambiguously identify the transaction. This identification is passed on, unchanged, throughout the entire end-to-end chain.



			<p>Usage: The end-to-end identification can be used for reconciliation or to link tasks relating to the transaction. It can be included in several messages related to the transaction.</p> <p>The Faster Payments Scheme can only access 31 characters for the EndToEndIdentification field.</p>
InstructedAmount	1..1	CBSPaymentSetupResponse1/Data/Initiation/InstructedAmount	<p>Amount of money to be moved between the debtor and creditor, before deduction of charges, expressed in the currency as ordered by the initiating party.</p> <p>Usage: This amount has to be transported unchanged through the transaction chain.</p>
Currency	1..1	CBSPaymentSetupResponse1/Data/Initiation/InstructedAmount/Currency	A code allocated to a currency by a Maintenance Agency under an international identification scheme, as described in the latest edition of the international standard ISO 4217 "Codes for the representation of currencies and funds".
DebtorAgent	0..1	CBSPaymentSetupResponse1/Data/Initiation/DebtorAgent	Financial institution servicing an account for the debtor.
SchemeName	1..1	CBSPaymentSetupResponse1/Data/Initiation/DebtorAgent/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSPaymentSetupResponse1/Data/Initiation/DebtorAgent/Identification	Unique and unambiguous identification of a person.
DebtorAccount	0..1	CBSPaymentSetupResponse1/Data/Initiation/DebtorAccount	Unambiguous identification of the account of the debtor to which a debit entry will be made as a result of the transaction.
SchemeName	1..1	CBSPaymentSetupResponse1/Data/Initiation/DebtorAccount/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSPaymentSetupResponse1/Data/Initiation/DebtorAccount/Identification	Identification assigned by an institution to identify an account. This identification is known by the account owner.
Name	0..1	CBSPaymentSetupResponse1/Data/Initiation/DebtorAccount/Name	<p>Name of the account, as assigned by the account servicing institution, in agreement with the account owner in order to provide an additional means of identification of the account.</p> <p>Usage: The account name is different from the account owner name. The account name is used in certain user communities to provide a means of identifying the account, in addition to</p>

			the account owner's identity and the account number.
SecondaryIdentification	0..1	CBSPaymentSetupResponse1/Data/Initiation/DebtorAccount/  SecondaryIdentification	This is secondary identification of the account, as assigned by the account servicing institution.  This can be used by building societies to additionally identify accounts with a roll number (in addition to a sort code and account number combination).
CreditorAgent	0..1	CBSPaymentSetupResponse1/Data/Initiation/CreditorAgent	Financial institution servicing an account for the creditor.
SchemeName	1..1	CBSPaymentSetupResponse1/Data/Initiation/CreditorAgent/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSPaymentSetupResponse1/Data/Initiation/CreditorAgent/Identification	Unique and unambiguous identification of a person.
CreditorAccount	1..1	CBSPaymentSetupResponse1/Data/Initiation/CreditorAccount	Unambiguous identification of the account of the creditor to which a credit entry will be posted as a result of the payment transaction.
SchemeName	1..1	CBSPaymentSetupResponse1/Data/Initiation/CreditorAccount/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSPaymentSetupResponse1/Data/Initiation/CreditorAccount/Identification	Identification assigned by an institution to identify an account. This identification is known by the account owner.
Name	1..1	CBSPaymentSetupResponse1/Data/Initiation/CreditorAccount/Name	Name of the account, as assigned by the account servicing institution, in agreement with the account owner in order to provide an additional means of identification of the account.  Usage: The account name is different from the account owner name. The account name is used in certain user communities to provide a means of identifying the account, in addition to the account owner's identity and the account number.
SecondaryIdentification	0..1	CBSPaymentSetupResponse1/Data/Initiation/CreditorAccount/  SecondaryIdentification	This is secondary identification of the account, as assigned by the account servicing institution.  This can be used by building societies to additionally identify accounts with a roll number (in addition to a sort code and account number combination).
RemittanceInformation	0..1	CBSPaymentSetupResponse1/Data/Initiation/RemittanceInformation	Information supplied to enable the matching of an entry with the items that the transfer is intended to settle, such as commercial invoices in an accounts' receivable system.

Unstructured	0..1	CBSPaymentSetupResponse1/Data/Initiation/ RemittanceInformation/Unstructured	Information supplied to enable the matching/reconciliation of an entry with the items that the payment is intended to settle, such as commercial invoices in an accounts' receivable system, in an unstructured form.
Reference	0..1	CBSPaymentSetupResponse1/Data/Initiation/ RemittanceInformation/Reference	<p>Unique reference, as assigned by the creditor, to unambiguously refer to the payment transaction.</p> <p>Usage: If available, the initiating party should provide this reference in the structured remittance information, to enable reconciliation by the creditor upon receipt of the amount of money.</p> <p>If the business context requires the use of a creditor reference or a payment remit identification, and only one identifier can be passed through the end-to-end chain, the creditor's reference or payment remittance identification should be quoted in the end-to-end transaction identification. The Faster Payments Scheme can only accept 18 characters for the ReferenceInformation field - which is where this ISO field will be mapped.</p>
Risk	1..1	CBSPaymentSetupResponse1/Risk	The Risk section is sent by the initiating party to CBS. It is used to specify additional details for risk scoring for Payments.
PaymentContextCode	0..1	CBSPaymentSetupResponse1/Risk/PaymentContextCode	Specifies the payment context
MerchantCategoryCode	0..1	CBSPaymentSetupResponse1/Risk/MerchantCategoryCode	Category code conform to ISO 18245, related to the type of services or goods the merchant provides for the transaction.
MerchantCustomerIdentification	0..1	CBSPaymentSetupResponse1/Risk/MerchantCustomerIdentification	The unique customer identifier of the PSU with the merchant.
DeliveryAddress	0..1	CBSPaymentSetupResponse1/Risk/DeliveryAddress	Information that locates and identifies a specific address, as defined by postal services or in free format text.
AddressLine	0..2	CBSPaymentSetupResponse1/Risk/DeliveryAddress/AddressLine	Information that locates and identifies a specific address, as defined by postal services, that is presented in free format text.
StreetName	0..1	CBSPaymentSetupResponse1/Risk/DeliveryAddress/StreetName	Name of a street or thoroughfare.
BuildingNumber	0..1	CBSPaymentSetupResponse1/Risk/DeliveryAddress/BuildingNumber	Number that identifies the position of a building on a street.
PostCode	0..1	CBSPaymentSetupResponse1/Risk/DeliveryAddress/PostCode	Identifier consisting of a group of letters and/or numbers that is added to a postal address to assist the sorting of mail.

TownName	1..1	CBSPaymentSetupResponse1/Risk/DeliveryAddress/TownName	Name of a built-up area, with defined boundaries, and a local government.
CountrySubDivision	0..2	CBSPaymentSetupResponse1/Risk/DeliveryAddress/CountrySubDivision	Identifies a subdivision of a country, for instance state, region, county.
Country	1..1	CBSPaymentSetupResponse1/Risk/DeliveryAddress/Country	Nation with its own government, occupying a particular territory.

### Payment Submission - Request

The CBSPaymentSubmission1 object will be used for a call to:

- POST /payment-submissions

### Notes

The payment-submission request object contains the:

- PaymentId
- The full payload from the payment setup request (including the Initiation and Risk sections)

The **Initiation** and **Risk** sections of the payment-submission request **must** match the **Initiation** and **Risk** sections of the corresponding payment setup request.

### Data Dictionary

Name	Cardinality	XPath	EnhancedDefinition
CBSPaymentSubmission1		CBSPaymentSubmission1	
Data	1..1	CBSPaymentSubmission1/Data	
PaymentId	1..1	CBSPaymentSubmission1/Data/PaymentId	Unique identification as assigned by CBS to uniquely identify the payment setup resource.
Initiation	1..1	CBSPaymentSubmission1/Data/Initiation	The Initiation payload is sent by the initiating party to CBS. It is used to request movement of funds from the debtor account to a creditor.
InstructionIdentification	1..1	CBSPaymentSubmission1/Data/Initiation/InstructionIdentification	Unique identification as assigned by an instructing party for an instructed party to unambiguously identify the instruction.  Usage: the instruction identification is a point to point reference that can be used between the instructing party and the instructed party to refer to the individual instruction. It can be included in several messages related to the instruction.
EndToEndIdentification	1..1	CBSPaymentSubmission1/Data/Initiation/EndToEndIdentification	Unique identification assigned by the initiating party to unambiguously identify the transaction. This identification is passed on,

			<p>unchanged, throughout the entire end-to-end chain.</p> <p>Usage: The end-to-end identification can be used for reconciliation or to link tasks relating to the transaction. It can be included in several messages related to the transaction.</p>
InstructedAmount	1..1	CBSPaymentSubmission1/Data/Initiation/InstructedAmount	<p>Amount of money to be moved between the debtor and creditor, before deduction of charges, expressed in the currency as ordered by the initiating party.</p> <p>Usage: This amount has to be transported unchanged through the transaction chain.</p>
Currency	1..1	CBSPaymentSubmission1/Data/Initiation/InstructedAmount/Currency	A code allocated to a currency by a Maintenance Agency under an international identification scheme, as described in the latest edition of the international standard ISO 4217 "Codes for the representation of currencies and funds".
DebtorAgent	0..1	CBSPaymentSubmission1/Data/Initiation/DebtorAgent	Financial institution servicing an account for the debtor.
SchemeName	1..1	CBSPaymentSubmission1/Data/Initiation/DebtorAgent/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSPaymentSubmission1/Data/Initiation/DebtorAgent/Identification	Unique and unambiguous identification of a person.
DebtorAccount	0..1	CBSPaymentSubmission1/Data/Initiation/DebtorAccount	Unambiguous identification of the account of the debtor to which a debit entry will be made as a result of the transaction.
SchemeName	1..1	CBSPaymentSubmission1/Data/Initiation/DebtorAccount/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSPaymentSubmission1/Data/Initiation/DebtorAccount/Identification	Identification assigned by an institution to identify an account. This identification is known by the account owner.
Name	0..1	CBSPaymentSubmission1/Data/Initiation/DebtorAccount/Name	<p>Name of the account, as assigned by the account servicing institution, in agreement with the account owner in order to provide an additional means of identification of the account.</p> <p>Usage: The account name is different from the account owner name. The account name is used in certain user communities to provide a means of identifying the account, in addition to</p>

			the account owner's identity and the account number.
SecondaryIdentification	0..1	CBSPaymentSubmission1/Data/Initiation/DebtorAccount/  SecondaryIdentification	This is secondary identification of the account, as assigned by the account servicing institution. This can be used by building societies to additionally identify accounts with a roll number (in addition to a sort code and account number combination).
CreditorAgent	0..1	CBSPaymentSubmission1/Data/Initiation/CreditorAgent	Financial institution servicing an account for the creditor.
SchemeName	1..1	CBSPaymentSubmission1/Data/Initiation/CreditorAgent/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSPaymentSubmission1/Data/Initiation/CreditorAgent/Identification	Unique and unambiguous identification of a person.
CreditorAccount	1..1	CBSPaymentSubmission1/Data/Initiation/CreditorAccount	Unambiguous identification of the account of the creditor to which a credit entry will be posted as a result of the payment transaction.
SchemeName	1..1	CBSPaymentSubmission1/Data/Initiation/CreditorAccount/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSPaymentSubmission1/Data/Initiation/CreditorAccount/Identification	Identification assigned by an institution to identify an account. This identification is known by the account owner.
Name	1..1	CBSPaymentSubmission1/Data/Initiation/CreditorAccount/Name	Name of the account, as assigned by the account servicing institution, in agreement with the account owner in order to provide an additional means of identification of the account.  Usage: The account name is different from the account owner name. The account name is used in certain user communities to provide a means of identifying the account, in addition to the account owner's identity and the account number..
SecondaryIdentification	0..1	CBSPaymentSubmission1/Data/Initiation/CreditorAccount/  SecondaryIdentification	This is secondary identification of the account, as assigned by the account servicing institution. This can be used by building societies to additionally identify accounts with a roll number (in addition to a sort code and account number combination).
RemittanceInformation	0..1	CBSPaymentSubmission1/Data/Initiation/RemittanceInformation	Information supplied to enable the matching of an entry with the items that the transfer is intended to settle, such as commercial invoices in an accounts' receivable system.
Unstructured	0..1	CBSPaymentSubmission1/Data/Initiation/	Information supplied to enable the matching/reconciliation of an entry with the items that the payment is

		RemittanceInformation/Unstructured	intended to settle, such as commercial invoices in an accounts' receivable system, in an unstructured form.
Reference	0..1	CBSPaymentSubmission1/Data/Initiation/  RemittanceInformation/Reference	<p>Unique reference, as assigned by the creditor, to unambiguously refer to the payment transaction.</p> <p>Usage: If available, the initiating party should provide this reference in the structured remittance information, to enable reconciliation by the creditor upon receipt of the amount of money.</p> <p>If the business context requires the use of a creditor reference or a payment remit identification, and only one identifier can be passed through the end-to-end chain, the creditor's reference or payment remittance identification should be quoted in the end-to-end transaction identification. The Faster Payments Scheme can only accept 18 characters for the ReferenceInformation field - which is where this ISO field will be mapped.</p>
Risk	1..1	CBSPaymentSubmission1/Risk	The Risk section is sent by the initiating party to CBS. It is used to specify additional details for risk scoring for Payments.
PaymentContextCode	0..1	CBSPaymentSubmission1/Risk/PaymentContextCode	Specifies the payment context
MerchantCategoryCode	0..1	CBSPaymentSubmission1/Risk/MerchantCategoryCode	Category code conform to ISO 18245, related to the type of services or goods the merchant provides for the transaction.
MerchantCustomerIdentification	0..1	CBSPaymentSubmission1/Risk/MerchantCustomerIdentification	The unique customer identifier of the PSU with the merchant.
DeliveryAddress	0..1	CBSPaymentSubmission1/Risk/DeliveryAddress	Information that locates and identifies a specific address, as defined by postal services or in free format text.
AddressLine	0..2	CBSPaymentSubmission1/Risk/DeliveryAddress/AddressLine	Information that locates and identifies a specific address, as defined by postal services, that is presented in free format text.
StreetName	0..1	CBSPaymentSubmission1/Risk/DeliveryAddress/StreetName	Name of a street or thoroughfare.
BuildingNumber	0..1	CBSPaymentSubmission1/Risk/DeliveryAddress/BuildingNumber	Number that identifies the position of a building on a street.
PostCode	0..1	CBSPaymentSubmission1/Risk/DeliveryAddress/PostCode	Identifier consisting of a group of letters and/or numbers that is added to a postal address to assist the sorting of mail.
TownName	1..1	CBSPaymentSubmission1/Risk/DeliveryAddress/TownName	Name of a built-up area, with defined boundaries, and a local government.

CountrySubDivision	0..2	CBSPaymentSubmission1/Risk/DeliveryAddress/CountrySubDivision	Identifies a subdivision of a country, for instance state, region, county.
Country	1..1	CBSPaymentSubmission1/Risk/DeliveryAddress/Country	Nation with its own government, occupying a particular territory.

### Payment Submission - Response

The CBSPaymentSubmissionResponse1 object will be used for a response to a call to:

- POST /payment-submissions
- GET /payment-submissions/{PaymentSubmissionId}

### Notes

The Payment Submission POST **response** contains the:

- PaymentSubmissionId
- PaymentId
- Status of the payment-submission resource
- Date time the payment-submission resource was created

### Data Dictionary

Name	Cardinality	XPath	EnhancedDefinition
CBSPaymentSubmissionResponse1		CBSPaymentSubmissionResponse1	
Data	1..1	CBSPaymentSubmissionResponse1/Data	
PaymentSubmissionId	1..1	CBSPaymentSubmissionResponse1/Data/PaymentSubmissionId	Unique identification as assigned by CBS to uniquely identify the payment submission resource.
PaymentId	1..1	CBSPaymentSubmissionResponse1/Data/PaymentId	Unique identification as assigned by CBS to uniquely identify the payment setup resource.
Status	0..1	CBSPaymentSubmissionResponse1/Data/Status	Specifies the status of the payment submission resource.
CreationDateTime	1..1	CBSPaymentSubmissionResponse1/Data/CreationDateTime	Date and time at which the resource was created.

### Payments/{PaymentId}/funds-confirmation - Response

The CBSPaymentFundsConfirmationResponse1 object will be used for a response to a call to:

- GET /payments/{ConsentId}/funds-confirmation

### Notes

The confirmation of funds response contains the result of a funds availability check



### Data Dictionary

Name	Cardinality	XPath	EnhancedDefinition
CBSPaymentFundsConfirmationResponse1		CBSPaymentFundsConfirmationResponse1	
Data	1..1	CBSPaymentFundsConfirmationResponse1/Data	Result of a funds availability check.
FundsAvailableResult	0..1	CBSPaymentFundsConfirmationResponse1/Data/FundsAvailableResult	Unique identification as assigned by CBS to uniquely identify the payment submission resource.
FundsAvailableDateTime	1..1	CBSPaymentFundsConfirmationResponse1/Data/FundsAvailableDateTime	Date and time at which the funds availability check was generated.
FundsAvailable	1..1	CBSPaymentFundsConfirmationResponse1/Data/FundsAvailable	Yes or No Flag to indicate the availability of funds given the Amount in the consent request.

### Mandates setup - Request

The CBSMandateSetupRequest1 object will be used for the call to:

POST /mandates

### Notes

The PaymentType is used to specify if the mandate is for a single future dated payment or for a regular payment.

### Data Dictionary

Name	Cardinality	XPath	EnhancedDefinition
CBSMandateSetupRequest1		CBSMandateSetupRequest1	
Data	1..1	CBSMandateSetupRequest1/Data	Result of a funds availability check.
PaymentType	1..1	CBSMandateSetupRequest1/Data/PaymentType	CBSPaymentType1Code enumeration. Indicates whether the request is for a future dated or regular mandate.
Initiation	1..1	CBSMandateSetupRequest1/Data/Initiation	The Initiation payload is sent by the initiating party to the CBS. It is used to request movement of funds from the debtor account to a creditor.
InstructionIdentification	1..1	CBSMandateSetupRequest1/Data/Initiation/InstructionIdentification	Unique identification as assigned by an instructing party for an instructed party to unambiguously identify the instruction.

			Usage: the instruction identification is a point to point reference that can be used between the instructing party and the instructed party to refer to the individual instruction. It can be included in several messages related to the instruction.
EndToEndIdentification	1..1	CBSMandateSetupRequest1/Data/Initiation/EndToEndIdentification	<p>Unique identification assigned by the initiating party to unambiguously identify the transaction. This identification is passed on, unchanged, throughout the entire end-to-end chain.</p> <p>Usage: The end-to-end identification can be used for reconciliation or to link tasks relating to the transaction. It can be included in several messages related to the transaction.</p> <p>The Faster Payments Scheme can only access 31 characters for the EndToEndIdentification field.</p>
FuturePaymentDateTime	0..1	CBSMandateSetupRequest1/Data/Initiation/FuturePaymentDateTime	<p><b>Required if PaymentType = FutureDated</b></p> <p>Date at which the initiating party requests the clearing agent to process the payment.</p> <p>Usage: This is the date on which the debtor's account is to be debited.</p>
ValidFromDateTime	0..1	CBSMandateSetupRequest1/Data/Initiation/ValidFromDateTime	<p><b>Required if PaymentType = Regular</b></p> <p>The date on which the first payment for a Regular Payment will be made.</p>
ValidToDateTime	0..1	CBSMandateSetupRequest1/Data/Initiation/ValidToDateTime	The date on which the final payment for a Regular Payment will be made.
Frequency	0..1	CBSMandateSetupRequest1/Data/Initiation/Frequency	<p>CBSFrequency1Code Enumeration</p> <p>How often the regular payment is to be executed.</p>
InstructedAmount	1..1	CBSMandateSetupRequest1/Data/Initiation/InstructedAmount	Amount of money to be moved between the debtor and creditor, before deduction of charges, expressed in the currency as

			<p>ordered by the initiating party.</p> <p>Usage: This amount has to be transported unchanged through the transaction chain.</p>
Amount	1..1	CBSMandateSetupRequest1/Data/Initiation/InstructedAmount/Amount	A number of monetary units specified in an active currency where the unit of currency is explicit and compliant with ISO 4217.
Currency	1..1	CBSMandateSetupRequest1/Data/Initiation/InstructedAmount/Currency	A code allocated to a currency by a Maintenance Agency under an international identification scheme, as described in the latest edition of the international standard ISO 4217 "Codes for the representation of currencies and funds".
DebtorAgent	0..1	CBSMandateSetupRequest1/Data/Initiation/DebtorAgent	Financial institution servicing an account for the debtor.
SchemeName	1..1	CBSMandateSetupRequest1/Data/Initiation/DebtorAgent/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSMandateSetupRequest1/Data/Initiation/DebtorAgent/Identification	Unique and unambiguous identification of a person.
DebtorAccount	0..1	CBSMandateSetupRequest1/Data/Initiation/DebtorAccount	Unambiguous identification of the account of the debtor to which a debit entry will be made as a result of the transaction.
SchemeName	1..1	CBSMandateSetupRequest1/Data/Initiation/DebtorAccount/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSMandateSetupRequest1/Data/Initiation/DebtorAccount/Identification	Identification assigned by an institution to identify an account. This identification is known by the account owner.
Name	0..1	CBSMandateSetupRequest1/Data/Initiation/DebtorAccount/Name	<p>Name of the account, as assigned by the account servicing institution, in agreement with the account owner in order to provide an additional means of identification of the account.</p> <p>Usage: The account name is different from the account owner name. The account name is used in certain user communities to provide a means of identifying the account, in addition to the account</p>

			owner's identity and the account number.
SecondaryIdentification	0..1	CBSMandateSetupRequest1/Data/Initiation/DebtorAccount/SecondaryIdentification	This is secondary identification of the account, as assigned by the account servicing institution. This can be used by building societies to additionally identify accounts with a roll number (in addition to a sort code and account number combination).
CreditorAgent	0..1	CBSMandateSetupRequest1/Data/Initiation/CreditorAgent	Financial institution servicing an account for the creditor.
SchemeName	1..1	CBSMandateSetupRequest1/Data/Initiation/CreditorAgent/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSMandateSetupRequest1/Data/Initiation/CreditorAgent/Identification	Unique and unambiguous identification of a person.
CreditorAccount	1..1	CBSMandateSetupRequest1/Data/Initiation/CreditorAccount	Unambiguous identification of the account of the creditor to which a credit entry will be posted as a result of the payment transaction.
SchemeName	1..1	CBSMandateSetupRequest1/Data/Initiation/CreditorAccount/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSMandateSetupRequest1/Data/Initiation/CreditorAccount/Identification	Identification assigned by an institution to identify an account. This identification is known by the account owner.
Name	1..1	CBSMandateSetupRequest1/Data/Initiation/CreditorAccount/Name	Name of the account, as assigned by the account servicing institution, in agreement with the account owner in order to provide an additional means of identification of the account.  Usage: The account name is different from the account owner name. The account name is used in certain user communities to provide a means of identifying the account, in addition to the account owner's identity and the account number.  ASPSPs <b>may</b> carry out name validation for Confirmation of Payee, but it is not mandatory.
SecondaryIdentification	0..1	CBSMandateSetupRequest1/Data/Initiation/CreditorAccount/SecondaryIdentification	This is secondary identification of the account, as assigned by the account servicing institution. This can be used by building societies to additionally identify accounts with a roll number (in addition to a sort code and account number combination).

RemittanceInformation	0..1	CBSMandateSetupRequest1/Data/Initiation/RemittanceInformation	Information supplied to enable the matching of an entry with the items that the transfer is intended to settle, such as commercial invoices in an accounts' receivable system.
Unstructured	0..1	CBSMandateSetupRequest1/Data/Initiation/RemittanceInformation/Unstructured	Information supplied to enable the matching/reconciliation of an entry with the items that the payment is intended to settle, such as commercial invoices in an accounts' receivable system, in an unstructured form.
Reference	0..1	CBSMandateSetupRequest1/Data/Initiation/RemittanceInformation/Reference	<p>Unique reference, as assigned by the creditor, to unambiguously refer to the payment transaction.</p> <p>Usage: If available, the initiating party should provide this reference in the structured remittance information, to enable reconciliation by the creditor upon receipt of the amount of money.</p> <p>If the business context requires the use of a creditor reference or a payment remit identification, and only one identifier can be passed through the end-to-end chain, the creditor's reference or payment remittance identification should be quoted in the end-to-end transaction identification.</p> <p>The Faster Payments Scheme can only accept 18 characters for the ReferenceInformation field - which is where this ISO field will be mapped.</p>
Risk	1..1	CBSMandateSetupRequest1/Risk	The Risk section is sent by the initiating party to the CBS. It is used to specify additional details for risk scoring for Payments.
PaymentContextCode	0..1	CBSMandateSetupRequest1/Risk/PaymentContextCode	Specifies the payment context
MerchantCategoryCode	0..1	CBSMandateSetupRequest1/Risk/MerchantCategoryCode	Category code conform to ISO 18245, related to the type of services or goods the merchant provides for the transaction.
MerchantCustomerIdentification	0..1	CBSMandateSetupRequest1/Risk/MerchantCustomerIdentification	The unique customer identifier of the PSU with the merchant.
DeliveryAddress	0..1	CBSMandateSetupRequest1/Risk/DeliveryAddress	Information that locates and identifies a specific address, as defined by postal services or in free format text.
AddressLine	0..2	CBSMandateSetupRequest1/Risk/DeliveryAddress/AddressLine	Information that locates and identifies a specific address, as

			defined by postal services that is presented in free format text.
StreetName	0..1	CBSMandateSetupRequest1/Risk/DeliveryAddress/StreetName	Name of a street or thoroughfare.
BuildingNumber	0..1	CBSMandateSetupRequest1/Risk/DeliveryAddress/BuildingNumber	Number that identifies the position of a building on a street.
PostCode	0..1	CBSMandateSetupRequest1/Risk/DeliveryAddress/PostCode	Identifier consisting of a group of letters and/or numbers that is added to a postal address to assist the sorting of mail.
TownName	1..1	CBSMandateSetupRequest1/Risk/DeliveryAddress/TownName	Name of a built-up area, with defined boundaries, and a local government.
CountrySubDivision	0..2	CBSMandateSetupRequest1/Risk/DeliveryAddress/CountrySubDivision	Identifies a subdivision of a country, for instance state, region, county.
Country	1..1	CBSMandateSetupRequest1/Risk/DeliveryAddress/Country	Nation with its own government, occupying a particular territory.

### Mandates setup – Response

The CBSMandateSetupResponse1 object will be used for a response to a call to:

- POST /mandates
- GET /mandates/{PaymentId}

### Notes

The Mandate Setup **response** contains the full **original** payload from the Mandate Setup POST **request** - with these additional elements:

- PaymentId.
- Status of the Payment resource.
- Date time the Payment resource was created.

### Data Dictionary

Name	Cardinality	XPath	EnhancedDefinition
CBSMandateSetupResponse1		CBSMandateSetupResponse1	
Data	1..1	CBSMandateSetupResponse1/Data	
PaymentId	1..1	CBSMandateSetupResponse1/Data/PaymentId	Unique identification as assigned by CBS to uniquely identify the payment setup resource.
PaymentType	1..1	CBSMandateSetupRequest1/Data/PaymentType	CBSPaymentType1Code enumeration. Indicates whether the

			request is for a future dated or regular mandate.
Status	0..1	CBSMandateSetupResponse1/Data/Status	Specifies the status of the payment resource.
CreationDateTime	1..1	CBSMandateSetupResponse1/Data/CreationDateTime	Date and time at which the resource was created.
Initiation	1..1	CBSMandateSetupResponse1/Data/Initiation	The Initiation payload is sent by the initiating party to the CBS. It is used to request movement of funds from the debtor account to a creditor.
InstructionIdentification	1..1	CBSMandateSetupResponse1/Data/Initiation/InstructionIdentification	<p>Unique identification as assigned by an instructing party for an instructed party to unambiguously identify the instruction.</p> <p>Usage: the instruction identification is a point to point reference that can be used between the instructing party and the instructed party to refer to the individual instruction. It can be included in several messages related to the instruction.</p>
EndToEndIdentification	1..1	CBSMandateSetupResponse1/Data/Initiation/EndToEndIdentification	<p>Unique identification assigned by the initiating party to unambiguously identify the transaction. This identification is passed on, unchanged, throughout the entire end-to-end chain.</p> <p>Usage: The end-to-end identification can be used for reconciliation or to link tasks relating to the transaction. It can be included in several messages related to the transaction. The Faster Payments Scheme can only access 31 characters for the EndToEndIdentification field.</p>
FuturePaymentDateTime	0..1	CBSMandateSetupResponse1/Data/Initiation/FuturePaymentDateTime	<p><b>Required if PaymentType = FutureDated</b></p> <p>Date at which the initiating party requests the clearing agent to process the payment.</p> <p>Usage: This is the date on which the debtor's account is to be debited.</p>
ValidFromDateTime	0..1	CBSMandateSetupResponse1/Data/Initiation/ValidFromDateTime	<p><b>Required if PaymentType = Regular</b></p> <p>The date on which the first payment for a Regular Payment will be made.</p>
ValidToDateTime	0..1	CBSMandateSetupResponse1/Data/Initiation/ValidToDateTime	The date on which the final payment for a Regular Payment will be made.

Frequency	0..1	CBSMandateSetupResponse1/Data/Initiation/Frequency	CBSFrequency1Code Enumeration How often the regular payment is to be executed.
InstructedAmount	1..1	CBSMandateSetupResponse1/Data/Initiation/InstructedAmount	Amount of money to be moved between the debtor and creditor, before deduction of charges, expressed in the currency as ordered by the initiating party.  Usage: This amount has to be transported unchanged through the transaction chain.
Amount	1..1	CBSMandateSetupResponse1/Data/Initiation/InstructedAmount/Amount	A number of monetary units specified in an active currency where the unit of currency is explicit and compliant with ISO 4217.
Currency	1..1	CBSMandateSetupResponse1/Data/Initiation/InstructedAmount/Currency	A code allocated to a currency by a Maintenance Agency under an international identification scheme, as described in the latest edition of the international standard ISO 4217 "Codes for the representation of currencies and funds".
DebtorAgent	0..1	CBSMandateSetupResponse1/Data/Initiation/DebtorAgent	Financial institution servicing an account for the debtor.
SchemeName	1..1	CBSMandateSetupResponse1/Data/Initiation/DebtorAgent/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSMandateSetupResponse1/Data/Initiation/DebtorAgent/Identification	Unique and unambiguous identification of a person.
DebtorAccount	0..1	CBSMandateSetupResponse1/Data/Initiation/DebtorAccount	Unambiguous identification of the account of the debtor to which a debit entry will be made as a result of the transaction.
SchemeName	1..1	CBSMandateSetupResponse1/Data/Initiation/DebtorAccount/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSMandateSetupResponse1/Data/Initiation/DebtorAccount/Identification	Identification assigned by an institution to identify an account. This identification is known by the account owner.
Name	0..1	CBSMandateSetupResponse1/Data/Initiation/DebtorAccount/Name	Name of the account, as assigned by the account servicing institution, in agreement with the account owner in



			<p>order to provide an additional means of identification of the account.</p> <p>Usage: The account name is different from the account owner name. The account name is used in certain user communities to provide a means of identifying the account, in addition to the account owner's identity and the account number.</p>
SecondaryIdentification	0..1	CBSMandateSetupResponse1/Data/Initiation/DebtorAccount/SecondaryIdentification	<p>This is secondary identification of the account, as assigned by the account servicing institution.</p> <p>This can be used by building societies to additionally identify accounts with a roll number (in addition to a sort code and account number combination).</p>
CreditorAgent	0..1	CBSMandateSetupResponse1/Data/Initiation/CreditorAgent	Financial institution servicing an account for the creditor.
SchemeName	1..1	CBSMandateSetupResponse1/Data/Initiation/CreditorAgent/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSMandateSetupResponse1/Data/Initiation/CreditorAgent/Identification	Unique and unambiguous identification of a person.
CreditorAccount	1..1	CBSMandateSetupResponse1/Data/Initiation/CreditorAccount	Unambiguous identification of the account of the creditor to which a credit entry will be posted as a result of the payment transaction.
SchemeName	1..1	CBSMandateSetupResponse1/Data/Initiation/CreditorAccount/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSMandateSetupResponse1/Data/Initiation/CreditorAccount/Identification	Identification assigned by an institution to identify an account. This identification is known by the account owner.
Name	1..1	CBSMandateSetupResponse1/Data/Initiation/CreditorAccount/Name	<p>Name of the account, as assigned by the account servicing institution, in agreement with the account owner in order to provide an additional means of identification of the account.</p> <p>Usage: The account name is different from the account owner name. The account name is used in certain user communities to provide a means of identifying the account, in addition to the account owner's identity and the account number.</p> <p>ASPSPs <b>may</b> carry out name validation for Confirmation of Payee, but it is not mandatory.</p>
SecondaryIdentification	0..1	CBSMandateSetupResponse1/Data/Initiation/CreditorAccount/	This is secondary identification of the account, as assigned by the account

		SecondaryIdentification	servicing institution. This can be used by building societies to additionally identify accounts with a roll number (in addition to a sort code and account number combination).
RemittanceInformation	0..1	CBSMandateSetupResponse1/Data/Initiation/RemittanceInformation	Information supplied to enable the matching of an entry with the items that the transfer is intended to settle, such as commercial invoices in an accounts' receivable system.
Unstructured	0..1	CBSMandateSetupResponse1/Data/Initiation/ RemittanceInformation/Unstructured	Information supplied to enable the matching/reconciliation of an entry with the items that the payment is intended to settle, such as commercial invoices in an accounts' receivable system, in an unstructured form.
Reference	0..1	CBSMandateSetupResponse1/Data/Initiation/RemittanceInformation/Reference	Unique reference, as assigned by the creditor, to unambiguously refer to the payment transaction.  Usage: If available, the initiating party should provide this reference in the structured remittance information, to enable reconciliation by the creditor upon receipt of the amount of money.  If the business context requires the use of a creditor reference or a payment remit identification, and only one identifier can be passed through the end-to-end chain, the creditor's reference or payment remittance identification should be quoted in the end-to-end transaction identification. The Faster Payments Scheme can only accept 18 characters for the ReferenceInformation field - which is where this ISO field will be mapped.
Risk	1..1	CBSMandateSetupResponse1/Risk	The Risk section is sent by the initiating party to the CBS. It is used to specify additional details for risk scoring for Payments.
PaymentContextCode	0..1	CBSMandateSetupResponse1/Risk/PaymentContextCode	Specifies the payment context
MerchantCategoryCode	0..1	CBSMandateSetupResponse1/Risk/MerchantCategoryCode	Category code conform to ISO 18245, related to the type of services or goods the merchant provides for the transaction.
MerchantCustomerIdentification	0..1	CBSMandateRequest1/Risk/MerchantCustomerIdentification	The unique customer identifier of the PSU with the merchant.
DeliveryAddress	0..1	CBSMandateSetupResponse1/Risk/DeliveryAddress	Information that locates and identifies a specific address, as defined by postal services or in free format text.
AddressLine	0..2	CBSMandateSetupResponse1/Risk/DeliveryAddress/AddressLine	Information that locates and identifies a specific address, as defined by

			postal services that is presented in free format text.
StreetName	0..1	CBSMandateSetupResponse1/Risk/DeliveryAddress/StreetName	Name of a street or thoroughfare.
BuildingNumber	0..1	CBSMandateSetupResponse1/Risk/DeliveryAddress/BuildingNumber	Number that identifies the position of a building on a street.
PostCode	0..1	CBSMandateSetupResponse1/Risk/DeliveryAddress/PostCode	Identifier consisting of a group of letters and/or numbers that is added to a postal address to assist the sorting of mail.
TownName	1..1	CBSMandateSetupResponse1/Risk/DeliveryAddress/TownName	Name of a built-up area, with defined boundaries, and a local government.
CountrySubDivision	0..2	CBSMandateSetupResponse1/Risk/DeliveryAddress/CountrySubDivision	Identifies a subdivision of a country, for instance state, region, county.
Country	1..1	CBSMandateSetupResponse1/Risk/DeliveryAddress/Country	Nation with its own government, occupying a particular territory.

### Mandate Submission - Request

The CBSMandateSubmission1 object will be used for a call to:

- POST /mandate-submissions

### Notes

The payment-submission request object contains the:

- PaymentId
- The full payload from the mandate setup request (including the Initiation and Risk sections)

The **Initiation** and **Risk** sections of the payment-submission request **must** match the **Initiation** and **Risk** sections of the corresponding payment setup request.

### Data Dictionary

Name	Card-inality	XPath	EnhancedDefinition
CBSMandateSubmission1		CBSMandateSubmission1	
Data	1..1	CBSMandateSubmission1/Data	
PaymentId	1..1	CBSMandateSubmission1/Data/PaymentId	Unique identification as assigned by CBS to uniquely identify the payment setup resource.
PaymentType	1..1	CBSMandateSetupRequest1/Data/PaymentType	CBSPaymentType1Code enumeration. Indicates whether the request is for a future dated or regular mandate.
Initiation	1..1	CBSMandateSubmission1/Data/Initiation	The Initiation payload is sent by the initiating party to the CBS. It is used to

			request movement of funds from the debtor account to a creditor.
InstructionIdentification	1..1	CBSMandateSubmission1/Data/Initiation/InstructionIdentification	<p>Unique identification as assigned by an instructing party for an instructed party to unambiguously identify the instruction.</p> <p>Usage: the instruction identification is a point to point reference that can be used between the instructing party and the instructed party to refer to the individual instruction. It can be included in several messages related to the instruction.</p>
EndToEndIdentification	1..1	CBSMandateSubmission1/Data/Initiation/EndToEndIdentification	<p>Unique identification assigned by the initiating party to unambiguously identify the transaction. This identification is passed on, unchanged, throughout the entire end-to-end chain.</p> <p>Usage: The end-to-end identification can be used for reconciliation or to link tasks relating to the transaction. It can be included in several messages related to the transaction. The Faster Payments Scheme can only access 31 characters for the EndToEndIdentification field.</p>
FuturePaymentDateTime	0..1	CBSMandateSubmission1/Data/Initiation/FuturePaymentDateTime	<p><b>Required if PaymentType = FutureDated</b></p> <p>Date at which the initiating party requests the clearing agent to process the payment.</p> <p>Usage: This is the date on which the debtor's account is to be debited.</p>
ValidFromDateTime	0..1	CBSMandateSubmission1/Data/Initiation/ValidFromDateTime	<p><b>Required if PaymentType = Regular</b></p> <p>The date on which the first payment for a Regular Payment will be made.</p>
ValidToDateTime	0..1	CBSMandateSubmission1/Data/Initiation/ValidToDateTime	<p>The date on which the final payment for a Regular Payment will be made.</p>
Frequency	0..1	CBSMandateSubmission1/Data/Initiation/Frequency	<p>CBSFrequency1Code Enumeration</p> <p>How often the regular payment is to be executed.</p>
InstructedAmount	1..1	CBSMandateSubmission1/Data/Initiation/InstructedAmount	<p>Amount of money to be moved between the debtor and creditor,</p>

			<p>before deduction of charges, expressed in the currency as ordered by the initiating party.</p> <p>Usage: This amount has to be transported unchanged through the transaction chain.</p>
Amount	1..1	CBSMandateSubmission1/Data/Initiation/InstructedAmount/Amount	A number of monetary units specified in an active currency where the unit of currency is explicit and compliant with ISO 4217.
Currency	1..1	CBSMandateSubmission1/Data/Initiation/InstructedAmount/Currency	A code allocated to a currency by a Maintenance Agency under an international identification scheme, as described in the latest edition of the international standard ISO 4217 "Codes for the representation of currencies and funds".
DebtorAgent	0..1	CBSMandateSubmission1/Data/Initiation/DebtorAgent	Financial institution servicing an account for the debtor.
SchemeName	1..1	CBSMandateSubmission1/Data/Initiation/DebtorAgent/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSMandateSubmission1/Data/Initiation/DebtorAgent/Identification	Unique and unambiguous identification of a person.
DebtorAccount	0..1	CBSMandateSubmission1/Data/Initiation/DebtorAccount	Unambiguous identification of the account of the debtor to which a debit entry will be made as a result of the transaction.
SchemeName	1..1	CBSMandateSubmission1/Data/Initiation/DebtorAccount/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSMandateSubmission1/Data/Initiation/DebtorAccount/Identification	Identification assigned by an institution to identify an account. This identification is known by the account owner.
Name	0..1	CBSMandateSubmission1/Data/Initiation/DebtorAccount/Name	<p>Name of the account, as assigned by the account servicing institution, in agreement with the account owner in order to provide an additional means of identification of the account.</p> <p>Usage: The account name is different from the account owner name. The account name is used in certain user communities to provide a means of identifying the account, in addition to the account owner's identity and the account number.</p>

SecondaryIdentification	0..1	CBSMandateSubmission1/Data/Initiation/DebtorAccount/SecondaryIdentification	This is secondary identification of the account, as assigned by the account servicing institution. This can be used by building societies to additionally identify accounts with a roll number (in addition to a sort code and account number combination).
CreditorAgent	0..1	CBSMandateSubmission1/Data/Initiation/CreditorAgent	Financial institution servicing an account for the creditor.
SchemeName	1..1	CBSMandateSubmission1/Data/Initiation/CreditorAgent/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSMandateSubmission1/Data/Initiation/CreditorAgent/Identification	Unique and unambiguous identification of a person.
CreditorAccount	1..1	CBSMandateSubmission1/Data/Initiation/CreditorAccount	Unambiguous identification of the account of the creditor to which a credit entry will be posted as a result of the payment transaction.
SchemeName	1..1	CBSMandateSubmission1/Data/Initiation/CreditorAccount/SchemeName	Name of the identification scheme, in a coded form as published in an external list.
Identification	1..1	CBSMandateSubmission1/Data/Initiation/CreditorAccount/Identification	Identification assigned by an institution to identify an account. This identification is known by the account owner.
Name	1..1	CBSMandateSubmission1/Data/Initiation/CreditorAccount/Name	Name of the account, as assigned by the account servicing institution, in agreement with the account owner in order to provide an additional means of identification of the account.  Usage: The account name is different from the account owner name. The account name is used in certain user communities to provide a means of identifying the account, in addition to the account owner's identity and the account number.  ASPSPs <b>may</b> carry out name validation for Confirmation of Payee, but it is not mandatory.
SecondaryIdentification	0..1	CBSMandateSubmission1/Data/Initiation/CreditorAccount/ SecondaryIdentification	This is secondary identification of the account, as assigned by the account servicing institution. This can be used by building societies to additionally identify accounts with a roll number (in addition to a sort code and account number combination).
RemittanceInformation	0..1	CBSMandateSubmission1/Data/Initiation/RemittanceInformation	Information supplied to enable the matching of an entry with the items that the transfer is intended to settle, such as commercial invoices in an accounts' receivable system.

Unstructured	0..1	CBSMandateSubmission1/Data/Initiation/ RemittanceInformation/Unstructured	Information supplied to enable the matching/reconciliation of an entry with the items that the payment is intended to settle, such as commercial invoices in an accounts' receivable system, in an unstructured form.
Reference	0..1	CBSMandateSubmission1/Data/Initiation/RemittanceInformation/Reference	<p>Unique reference, as assigned by the creditor, to unambiguously refer to the payment transaction.</p> <p>Usage: If available, the initiating party should provide this reference in the structured remittance information, to enable reconciliation by the creditor upon receipt of the amount of money.</p> <p>If the business context requires the use of a creditor reference or a payment remit identification, and only one identifier can be passed through the end-to-end chain, the creditor's reference or payment remittance identification should be quoted in the end-to-end transaction identification. The Faster Payments Scheme can only accept 18 characters for the ReferenceInformation field - which is where this ISO field will be mapped.</p>
Risk	1..1	CBSMandateSubmission1/Risk	The Risk section is sent by the initiating party to the CBS. It is used to specify additional details for risk scoring for Payments.
PaymentContextCode	0..1	CBSMandateSubmission1/Risk/PaymentContextCode	Specifies the payment context
MerchantCategoryCode	0..1	CBSMandateSubmission1/Risk/MerchantCategoryCode	Category code conform to ISO 18245, related to the type of services or goods the merchant provides for the transaction.
MerchantCustomerIdentification	0..1	CBSMandateSubmission1/Risk/MerchantCustomerIdentification	The unique customer identifier of the PSU with the merchant.
DeliveryAddress	0..1	CBSMandateSetupResponse1/Risk/DeliveryAddress	Information that locates and identifies a specific address, as defined by postal services or in free format text.
AddressLine	0..2	CBSMandateSubmission1/Risk/DeliveryAddress/AddressLine	Information that locates and identifies a specific address, as defined by postal services that is presented in free format text.
StreetName	0..1	CBSMandateSubmission1/Risk/DeliveryAddress/StreetName	Name of a street or thoroughfare.
BuildingNumber	0..1	CBSMandateSubmission1/Risk/DeliveryAddress/BuildingNumber	Number that identifies the position of a building on a street.
PostCode	0..1	CBSMandateSubmission1/Risk/DeliveryAddress/PostCode	Identifier consisting of a group of letters and/or numbers that is added to a postal address to assist the sorting of mail.

TownName	1..1	CBSMandateSubmission1/Risk/DeliveryAddress/TownName	Name of a built-up area, with defined boundaries, and a local government.
CountrySubDivision	0..2	CBSMandateSubmission1/Risk/DeliveryAddress/CountrySubDivision	Identifies a subdivision of a country, for instance state, region, county.
Country	1..1	CBSMandateSubmission1/Risk/DeliveryAddress/Country	Nation with its own government, occupying a particular territory.

## Mandate Submission - Response

The CBSMandateSubmissionResponse1 object will be used for a response to a call to:

- POST /mandate-submissions
- GET /mandate-submissions/{PaymentSubmissionId}

## Notes

The Payment Submission POST **response** contains the:

- PaymentSubmissionId
- PaymentId
- Status of the mandate-submission resource
- Date time the mandate-submission resource was created

## Data Dictionary

Name	Cardinality	XPath	EnhancedDefinition
CBSMandateSubmissionResponse1		CBSMandateSubmissionResponse1	
Data	1..1	CBSMandateSubmissionResponse1/Data	
PaymentSubmissionId	1..1	CBSMandateSubmissionResponse1/Data/PaymentSubmissionId	Unique identification as assigned by CBS to uniquely identify the payment submission resource.
PaymentId	1..1	CBSMandateSubmissionResponse1/Data/PaymentId	Unique identification as assigned by CBS to uniquely identify the payment setup resource.
Status	0..1	CBSMandateSubmissionResponse1/Data/Status	Specifies the status of the payment submission resource.
CreationDateTime	1..1	CBSMandateSubmissionResponse1/Data/CreationDateTime	Date and time at which the resource was created.

## Data Payload – Error Response Structure

### Example

```
{
  "Code": "...",
```



```

    "Id": "...",
    "Message": "...",
    "Errors": [
      {
        "ErrorCode": "...",
        "Message": "...",
        "Path": "...",
        "Url": "..."
      }
    ]
  }
}

```

### Data Dictionary

Name	Card-inality	Data Type	Notes
ErrorResponse			An array of detail error codes, and messages, and URLs to documentation to help remediation.
ErrorResponse /Code	1..1		High level textual error code, to help categorize the errors.
ErrorResponse /Message	1..1	Max128Text	Brief Error message, e.g., 'There is something wrong with the request parameters provided'
ErrorResponse /Errors	0..1		
ErrorResponse /Errors/ErrorCode	1..1	ISODateTime	Low level textual error code
ErrorResponse /Errors/Message	1..n		A description of the error that occurred

### Data Payload - Enumerations

This section gives the definitions for enumerations used in the Payment APIs.

Code Class	Name	Definition
------------	------	------------

OBExternalFinancialInstitutionIdentification2Code	BICFI	Valid BICs for financial institutions are registered by the ISO 9362 Registration Authority in the BIC directory, and consist of eight (8) or eleven (11) contiguous characters.
OBExternalPaymentContext1Code	BillPayment	The context of the payment initiation is a bill payment.
OBExternalPaymentContext1Code	EcommerceGoods	The context of the payment initiation is a for goods via an ecommerce channel.
OBExternalPaymentContext1Code	EcommerceServices	The context of the payment initiation is a for services via an ecommerce channel.
OBExternalPaymentContext1Code	PersonToPerson	The context of the payment initiation is a person to person payment.
OBExternalPaymentContext1Code	Other	The context of the payment initiation is of an other type.
OBTransactionIndividualStatus1Code	AcceptedCustomerProfile	Preceding check of technical validation was successful. Customer profile check was also successful.
OBTransactionIndividualStatus1Code	AcceptedSettlementCompleted	<p>Settlement on the debtor's account has been completed.</p> <p>Usage : this can be used by the first agent to report to the debtor that the transaction has been completed. Warning : this status is provided for transaction status reasons, not for financial information. It can only be used after bilateral agreement.</p> <p>PISPs <b>must not</b> use this status as confirmation that settlement is complete on the creditor's account.</p>
OBTransactionIndividualStatus1Code	AcceptedSettlementInProgress	All preceding checks such as technical validation and customer profile were successful and therefore the payment initiation has been accepted for execution.
OBTransactionIndividualStatus1Code	AcceptedTechnicalValidation	Authentication and syntactical and semantic validation are successful.
OBTransactionIndividualStatus1Code	Pending	Payment initiation or individual transaction included in the payment initiation is pending. Further checks and status update will be performed.
OBTransactionIndividualStatus1Code	Rejected	Payment initiation or individual transaction included in the payment initiation has been rejected.
CBSPaymentType1Code	FutureDated	The mandate is for a future dated one-off payment
CBSPaymentType1Code	Regular	The mandate is for a regular payment
CBSFrequency1Code	Weekly	The regularity of regular payments
CBSFrequency1Code	Fortnightly	The regularity of regular payments

CBSFrequency1Code	Monthly	The regularity of regular payments
CBSFrequency1Code	Quarterly	The regularity of regular payments
CBSFrequency1Code	Half Yearly	The regularity of regular payments
CBSFrequency1Code	Annually	The regularity of regular payments

## Identifier Fields

This section describes the identifiers used through the Payment API flows - the direction of flow through the system, and how they are used.

The standard definitions for the elements in the API payloads are described in the Data Payload section. However, this table gives further detail on the business meaning, and how they are used.

Generated	Identifier	Business Description
Merchant/PISP Sent in API Payload	EndToEndIdentification	The EndToEndIdentification reference is a reference that can be populated by the debtor (or merchant in the ecommerce space). This reference is important to the debtor (could be an internal reference Id against the transaction), it is NOT the reference information that will be primarily populated on the statement of the creditor (beneficiary).
Merchant/PISP Sent in API Payload	InstructionIdentification	PISP generates the InstructionIdentification which is a unique transaction Id and passes it to CBS (this is mandatory), but this doesn't have to go any further in the payment flow. The flow of this identifier needs to align with payment scheme rules.  The expectation is that this is unique indefinitely across all time periods. The PISP can ensure this is indefinitely unique by including a date or date time element to the field, or by inserting a unique Id.
Merchant/PISP Sent in API Payload	RemittanceInformation	The RemittanceInformation is the reference information that creditor (or beneficiary) will need to reconcile (e.g. Invoice 123).
CBS / API System	PaymentId	Unique identification as assigned by CBS to uniquely identify the payment setup resource.
CBS / API System	PaymentSubmissionId	Unique identification as assigned by CBS to uniquely identify the payment-submission resource.

The tables below identify the actor that initially creates each of the message identifiers and their transmission and visibility to other actors.

These flows are indicative - and will be dependent on what payment schemes or agencies are able to support.

Key:

O indicates the actor that creates the identifier.

=> downstream direction of flow

<= upstream direction of flow

## Merchant Flow

Identifier	PSU	Merchant	PISP	CBS	Payment Scheme	Beneficiary
EndToEndIdentification		0	=>	=>	=>	=>
RemittanceInformation		0	=>	=>	=>	=>
InstructionIdentification			0	=>		
PaymentId			<=	0		
PaymentSubmissionId			<=	0		

## Person to Person Flow

Identifier	PSU	Merchant	PISP	CBS	Payment Scheme	Beneficiary
EndToEndIdentification			0	=>	=>	=>
RemittanceInformation	0		=>	=>	=>	=>
InstructionIdentification			0	=>		
PaymentId			<=	0		
PaymentSubmissionId			<=	0		

## Mapping to Schemes & Standards

### Transaction Status

A principle has been agreed to adhere to the ISO Transaction Status codes.

The ISO Transaction Status has been split into statuses for the **payment** setup resource, and the **payment-submission** resource.



## Usage Examples

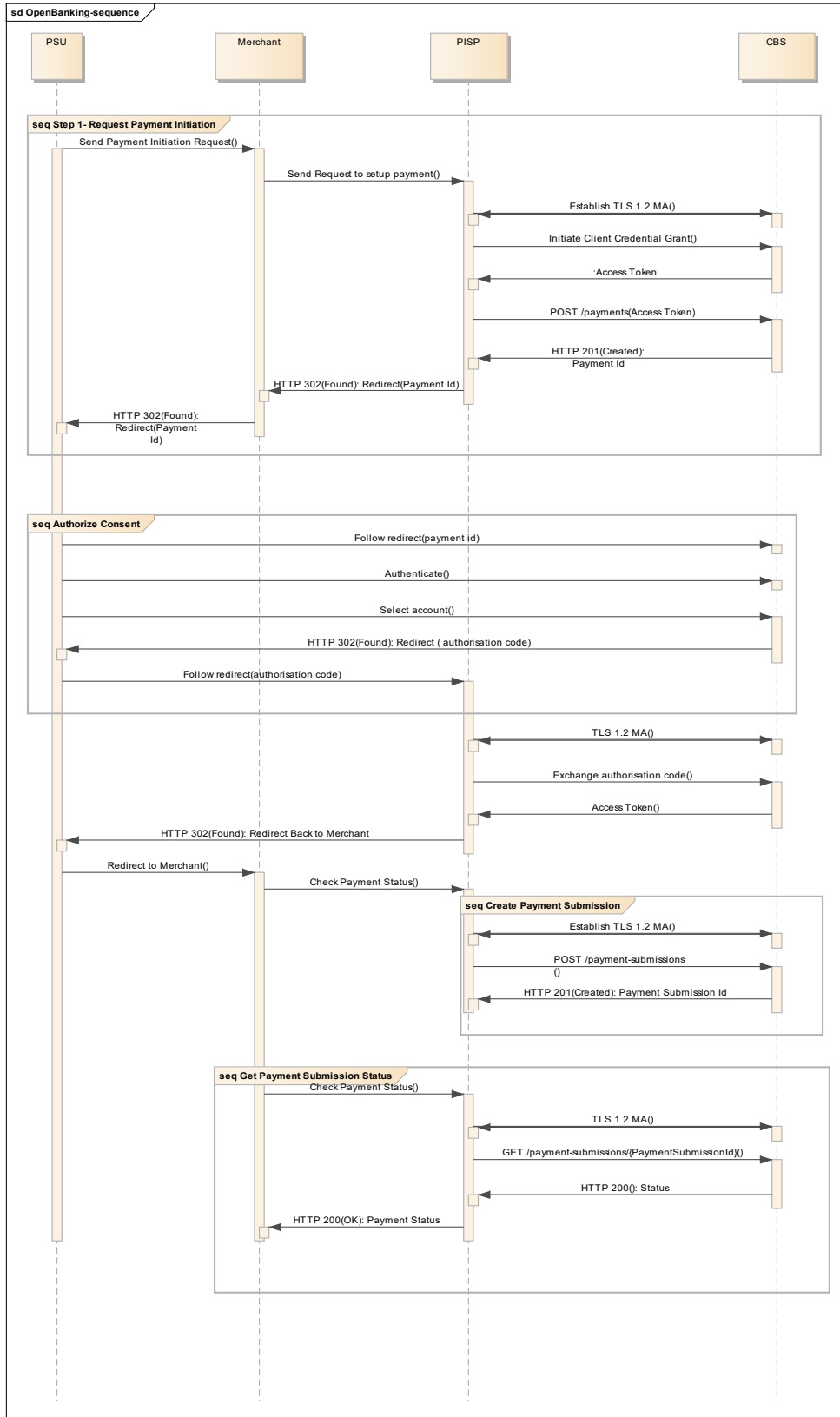
### Merchant

This example set of flows and payload examples are for a payment initiated by a merchant via a PISP.

In this scenario:

- The merchant has **not** specified the Debtor Account details for the the PSU - the PSU will select their account during the authorisation of consent
- The merchant's account is a building society account - with a roll number specified in the SecondaryIdentification field

# Sequence Diagram



## Illustrative Interactions

### POST /payments request

Request Payload	Response Payload
<pre>POST /payments HTTP/1.1 Authorization: Bearer 2YotnFZFEjr1zCsicMWpAA x-idempotency-key: FRESCO.21302.GFX.20 x-fapi-financial-id: CBSOpenBanking x-fapi-customer-last-logged-time: Sun, 10 Sep 2017 19:43:31 UTC x-fapi-customer-ip-address: 104.25.212.99 x-fapi-interaction-id: 93bac548-d2de-4546-b106- 880a5018460d X-Client-Id: tppclientid Content-Type: application/json Accept: application/json  {   "Data": {     "Initiation": {       "InstructionIdentification": "ACME412",       "EndToEndIdentification": "FRESCO.21302.GFX.22",       "InstructedAmount": {         "Amount": "1.43",         "Currency": "GBP"       },       "CreditorAccount": {         "SchemeName": "SortCodeAccountNumber",         "Identification": "40230341298607",         "Name": "MR R E DEELEY",         "SecondaryIdentification": "0002"       },       "RemittanceInformation": {         "Reference": "Immediate-Payment",         "Unstructured": "Internal ops code 5120101"       }     }   },   "Risk": {     "PaymentContextCode": "EcommerceGoods",     "MerchantCategoryCode": "5967",     "MerchantCustomerIdentification": "053598653254",     "DeliveryAddress": {       "AddressLine": [         "Flat 7",         "Acacia Lodge"       ],       "StreetName": "Acacia Avenue",       "BuildingNumber": "27",       "PostCode": "GU31 2ZZ",       "TownName": "Sparsholt",     }   } }</pre>	<pre>HTTP/1.1 201 Created x-fapi-interaction-id: 93bac548-d2de-4546-b106- 880a5018460d Content-Type: application/json  {   "Data": {     "PaymentId": "86e15e3f-fb80-4b45-b1ab- c87621d59b00",     "CreationDateTime": "2018-04- 06T14:06:00+01:00",     "Initiation": {       "InstructionIdentification": "ACME412",       "EndToEndIdentification": "FRESCO.21302.GFX.22",       "InstructedAmount": {         "Amount": "1.43",         "Currency": "GBP"       },       "CreditorAccount": {         "SchemeName": "SortCodeAccountNumber",         "Identification": "40230341298607",         "Name": "MR R E DEELEY",         "SecondaryIdentification": "0002"       },       "RemittanceInformation": {         "Unstructured": "Internal ops code 5120101",         "Reference": "Immediate-Payment"       }     },     "Status": "AcceptedTechnicalValidation"   },   "Risk": {     "PaymentContextCode": "EcommerceGoods",     "MerchantCategoryCode": "5967",     "MerchantCustomerIdentification": "053598653254",     "DeliveryAddress": {       "StreetName": "Acacia Avenue",       "BuildingNumber": "27",       "PostCode": "GU31 2ZZ",       "TownName": "Sparsholt",       "Country": "UK",       "AddressLine": [         "Flat 7",         "Acacia Lodge"       ]     }   } }</pre>

```

    "CountrySubDivision": [
      "Wessex"
    ],
    "Country": "UK"
  }
}

```

```

    ],
    "CountrySubDivision": ["Wessex"]
  }
},
"Links": {"Self": "/open-banking/v2.0/payments"},
"Meta": {}
}

```

## POST /payment-submissions Request

Request Payload
<pre> POST /payment-submissions HTTP/1.1 Authorization: Bearer Jhingapulaav x-idempotency-key: FRESNO.1317.GFX.22 x-fapi-financial-id: CBSOpenBanking x-fapi-customer-last-logged-time: Sun, 10 Sep 2017 19:43:31 UTC x-fapi-customer-ip-address: 104.25.212.99 x-fapi-interaction-id: 93bac548-d2de-4546-b106- 880a5018460d X-Client-Id: tppclientid Content-Type: application/json Accept: application/json  {   "Data":   {     "PaymentId": "86e15e3f-fb80-4b45-b1ab- c87621d59b00",     "Initiation": {       "CreditorAccount": {         "Identification": "87878UFUHR",         "Name": "MR KELVIN",         "SchemeName": "SortCodeAccountNumber",         "SecondaryIdentification": "HFHHFHVBHFJE"       },       "CreditorAgent": {         "Identification": "HHU5UU5",         "SchemeName": "BICFI"       },       "DebtorAccount": {         "Identification": "7T575YC7NT",         "Name": "MRS KEL",         "SchemeName": "SortCodeAccountNumber",         "SecondaryIdentification": "YJB5Y7875X3Y7YN"       },       "DebtorAgent": {         "Identification": "454556G65",         "SchemeName": "BICFI"       }     }   } } </pre>

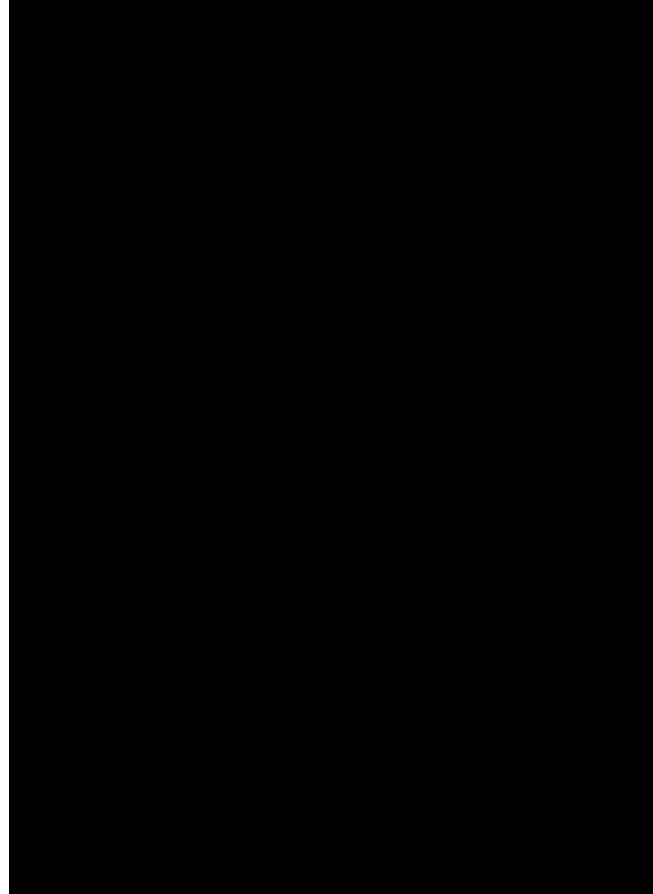
Response Payload
<pre> HTTP/1.1 201 Created x-fapi-interaction-id: 93bac548-d2de-4546-b106- 880a5018460d Content-Type: application/json  {   "Data":   {     "CreationDateTime": "2017-08-08T15:23:02Z",     "PaymentId": "86e15e3f-fb80-4b45-b1ab- c87621d59b00",     "PaymentSubmissionId": "GGDGHVG",     "Status": "AcceptedSettlementInProgress"   },   "Links":   {     "Self": "/open-banking/v2.0/payment-submissions"   },   "Meta":   {} } </pre>



```

},
"EndToEndIdentification": "FYUJU74754UYU4U",
"InstructedAmount": {
"Amount": "100",
"Currency": "GBP"
},
"InstructionIdentification": "GUG7457R7F4U",
"RemittanceInformation": {
"Reference": "YUY5YY45N",
"Unstructured": "7Y4X7R7484RU834"
}
},
"Risk": {
"DeliveryAddress": {
"AddressLine": ["GHGHH", "HHHHHJKKFK"],
"BuildingNumber": "12",
"Country": "UK",
"CountrySubDivision": ["HHFJRHTJ", "RGYGYTTHH"],
"PostCode": "Y7457Y75",
"StreetName": "8HF45HUFCUUIHI",
"TownName": "8Y8Y54UYHU4U5U"
},
"MerchantCategoryCode": "GRTT",
"MerchantCustomerIdentification": "DCTT5768",
"PaymentContextCode": "EcommerceGoods"
}
}

```



### GET /payments Request

#### Request Payload

```

GET /payments/86e15e3f-fb80-4b45-b1ab-
c87621d59b00 HTTP/1.1
Authorization: Bearer Jhingapulaav
x-fapi-financial-id: CBSOpenBanking
x-fapi-customer-last-logged-time: Sun, 10 Sep 2017
19:43:31 UTC
x-fapi-customer-ip-address: 104.25.212.99
x-fapi-interaction-id: 93bac548-d2de-4546-b106-
880a5018460d
X-Client-Id: tppclientid
Accept: application/json

```

#### Response Payload

```

HTTP/1.1 200 OK
x-fapi-interaction-id: 93bac548-d2de-4546-b106-
880a5018460d
Content-Type: application/json
{
  "Data": {
    "PaymentId": "86e15e3f-fb80-4b45-b1ab-
c87621d59b00",
    "Status": "AcceptedTechnicalValidation",
    "CreationDateTime": "2017-06-05T15:15:19Z",
    "Initiation": {
      "InstructionIdentification": "ACME412",
      "EndToEndIdentification":
"FRESCO.21302.GFX.20",
      "InstructedAmount": {
        "Amount": "165.88",
        "Currency": "GBP"
      },
    },
    "CreditorAgent": {

```

```
"SchemeName": "BICFI",
"Identification": "SC080800"
},
"CreditorAccount": {
  "Identification": "21325698",
  "Name": "ACME Inc",
  "SchemeName": "SortCodeAccountNumber",
  "SecondaryIdentification": "0002"
},
  "DebtorAccount": {
    "Identification": "1254653",
    "Name": "Prince",
    "SchemeName":
"SortCodeAccountNumber",
"SecondaryIdentification": "2221"
  },
    "DebtorAgent": {
      "Identification": "54667",
      "SchemeName": "BICFI"
    },
    "RemittanceInformation": {
      "Reference": "FRESCO-101",
      "Unstructured": "Internal ops code 5120101"
    }
  },
  "Risk": {
    "PaymentContextCode": "EcommerceGoods",
    "MerchantCategoryCode": "5967",
    "MerchantCustomerIdentification":
"053598653254",
    "DeliveryAddress": {
      "AddressLine": [
        "Flat 7",
        "Acacia Lodge"
      ],
      "StreetName": "Acacia Avenue",
      "BuildingNumber": "27",
      "PostCode": "GU31 2ZZ",
      "TownName": "Sparsholt",
      "CountySubDivision": [
        "Wessex", "kessel"
      ],
      "Country": "UK"
    }
  },
  "Links": {
    "Self": "/open-
banking/v2.0/payments/86e15e3f-fb80-4b45-b1ab-
c87621d59b00"
  },
  "Meta": { }
}
```

## GET /payment-submissions Request

### Request Payload

```
GET /payment-submissions/GGDGHVG HTTP/1.1
Authorization: Bearer 2YotnFZFEjr1zCsicMWpAA
x-fapi-financial-id: CBSOpenBanking
x-fapi-customer-last-logged-time: Sun, 10 Sep 2017
19:43:31 UTC
x-fapi-customer-ip-address: 104.25.212.99
x-fapi-interaction-id: 93bac548-d2de-4546-b106-
880a5018460d
X-Client-Id: tppclientid
Accept: application/json
```

### Response Payload

```
HTTP/1.1 200 OK
Content-Type: application/json

{ {
  "Data": {
    "PaymentSubmissionId":
    "GGDGHVG",
    "PaymentId": "86e15e3f-fb80-4b45-b1ab-
c87621d59b00",
    "Status":
    "AcceptedSettlementInProgress",
    "CreationDateTime":
    "2017-06-05T15:15:22+00:00"
  },
  "Links": {
    "Self":
    "/open-banking/v2.0/payment-submiss
ions/GGDGHVG"
  },
  "Meta": {}
}
```

## GET /payments/{PaymentId}/funds-confirmation

### Funds Available example

### Request Payload

```
GET /payments/86e15e3f-fb80-4b45-b1ab-
c87621d59b00/funds-confirmation HTTP/1.1
Authorization: Bearer 2YotnFZFEjr1zCsicMWpAA
x-fapi-financial-id: CBSOpenBanking
x-fapi-customer-last-logged-time: Sun, 10 Sep 2017
19:43:31 UTC
x-fapi-customer-ip-address: 104.25.212.99
x-fapi-interaction-id: 93bac548-d2de-4546-b106-
880a5018460d
X-Client-Id: tppclientid
Accept: application/json
```

### Response Payload

```
HTTP/1.1 200 OK
Content-Type: application/json

{ {
  "Data": {
    "FundsAvailableResult": {
      "FundsAvailableDateTime": 2017-06-
05T15:15:22+00:00",
      "FundsAvailable": "Yes"
    },
    "Links": {
      "Self":
      "/open-banking/v2.0/payments86e15e3f-fb80-4b45-
b1ab-c87621d59b00/funds-confirmation"
    },
    "Meta": {}
  }
}
```

### **Funds Not Available Example**

Request Payload
GET /payments/86e15e3f-fb80-4b45-b1ab-c87621d59b00/funds-confirmation HTTP/1.1 Authorization: Bearer 2YotnFZFEjr1zCsicMWpAA x-fapi-financial-id: CBSOpenBanking x-fapi-customer-last-logged-time: Sun, 10 Sep 2017 19:43:31 UTC x-fapi-customer-ip-address: 104.25.212.99 x-fapi-interaction-id: 93bac548-d2de-4546-b106-880a5018460d Accept: application/json

Response Payload
HTTP/1.1 200 OK Content-Type: application/json <pre>{ {   "Data": {     "FundsAvailableResult": {       "FundsAvailableDateTime": "2017-06-05T15:15:22+00:00",       "FundsAvailable": "No"     },     "Links": {       "Self":         "/open-banking/v2.0/payments86e15e3f-fb80-4b45-b1ab-c87621d59b00/funds-confirmation"     },     "Meta": {}   } }</pre>

### **POST /mandates Request**

#### **Future Dated Example**

Request Payload
POST /mandates HTTP/1.1 Authorization: Bearer 2YotnFZFEjr1zCsicMWpAA x-idempotency-key: FRESCO.21302.GFX.20 x-fapi-financial-id: CBSOpenBanking x-fapi-customer-last-logged-time: Sun, 10 Sep 2017 19:43:31 UTC x-fapi-customer-ip-address: 104.25.212.99 x-fapi-interaction-id: 93bac548-d2de-4546-b106-880a5018460d X-Client-Id: tppclientid Content-Type: application/json Accept: application/json <pre>{   "Data": {     "PaymentType": "FutureDated",     "Initiation": {       "InstructionIdentification": "ACME412",       "EndToEndIdentification":         "FRESCO.21302.GFX.22",       "FuturePaymentDateTime": "2017-12-25",       "InstructedAmount": {         "Amount": "1.43",         "Currency": "GBP"       }     }   } }</pre>

Response Payload
HTTP/1.1 201 Created x-fapi-interaction-id: 93bac548-d2de-4546-b106-880a5018460d Content-Type: application/json <pre>{   "Data": {     "PaymentId": "86e15e3f-fb80-4b45-b1ab-c87621d59b00",     "CreationDateTime": "2018-04-06T14:06:00+01:00",     "PaymentType": "FutureDated",     "Initiation": {       "InstructionIdentification": "ACME412",       "EndToEndIdentification":         "FRESCO.21302.GFX.22",       "FuturePaymentDateTime": "2017-12-25",       "InstructedAmount": {         "Amount": "1.43",         "Currency": "GBP"       },       "CreditorAccount": {         "SchemeName":           "SortCodeAccountNumber", </pre>

```

    },
    "CreditorAccount": {
      "SchemeName": "SortCodeAccountNumber",
      "Identification": "40230341298607",
      "Name": " MR R E DEELEY ",
      "SecondaryIdentification": "0002"
    },
    "RemittanceInformation": {
      "Reference": "Future-Payment",
      "Unstructured": "Internal ops code 5120101"
    }
  }
},
"Risk": {
  "PaymentContextCode": "EcommerceGoods",
  "MerchantCategoryCode": "5967",
  "MerchantCustomerIdentification": "053598653254",
  "DeliveryAddress": {
    "AddressLine": [
      "Flat 7",
      "Acacia Lodge"
    ],
    "StreetName": "Acacia Avenue",
    "BuildingNumber": "27",
    "PostCode": "GU31 2ZZ",
    "TownName": "Sparsholt",
    "CountrySubDivision": [
      "Wessex"
    ],
    "Country": "UK"
  }
}
}
}

```

```

    "Identification": "40230341298607",
    "Name": "MR R E DEELEY",
    "SecondaryIdentification": "0002"
  },
  "RemittanceInformation": {
    "Unstructured": "Internal ops code
5120101",
    "Reference": "Future-Payment"
  }
},
"Status": "AcceptedTechnicalValidation"
},
"Risk": {
  "PaymentContextCode": "EcommerceGoods",
  "MerchantCategoryCode": "5967",
  "MerchantCustomerIdentification":
"053598653254",
  "DeliveryAddress": {
    "StreetName": "Acacia Avenue",
    "BuildingNumber": "27",
    "PostCode": "GU31 2ZZ",
    "TownName": "Sparsholt",
    "Country": "UK",
    "AddressLine": [
      "Flat 7",
      "Acacia Lodge"
    ],
    "CountrySubDivision": ["Wessex"]
  }
},
"Links": {"Self": "/open-banking-
extensions/v2.0/mandates"},
"Meta": {}
}
}

```

### Regular Example

#### Request Payload

```

POST /mandates HTTP/1.1
Authorization: Bearer 2YotnFZFEjr1zCsicMWpAA
x-idempotency-key: FRESCO.21302.GFX.20
x-fapi-financial-id: CBSOpenBanking
x-fapi-customer-last-logged-time: Sun, 10 Sep 2017
19:43:31 UTC
x-fapi-customer-ip-address: 104.25.212.99
x-fapi-interaction-id: 93bac548-d2de-4546-b106-
880a5018460d
Content-Type: application/json
Accept: application/json

{
  "Data": {
    "PaymentType": "Regular",

```

#### Response Payload

```

HTTP/1.1 201 Created
x-fapi-interaction-id: 93bac548-d2de-4546-b106-
880a5018460d
Content-Type: application/json

{
  "Data": {
    "PaymentId": "86e15e3f-fb80-4b45-b1ab-
c87621d59b00",
    "CreationDateTime": "2018-04-
06T14:06:00+01:00",
    "PaymentType": "Regular",
    "Initiation": {
      "InstructionIdentification": "ACME412",

```

```

"Initiation": {
  "InstructionIdentification": "ACME412",
  "EndToEndIdentification":
  "FRESCO.21302.GFX.22",
  "ValidFromDate": "2017-12-
25T00:00:00+00:00",
  "ValidToDate": "2018-12-
25T00:00:00+00:00",
  "Frequency": "Monthly",
  "InstructedAmount": {
    "Amount": "1.43",
    "Currency": "GBP"
  },
  "CreditorAccount": {
    "SchemeName": "SortCodeAccountNumber",
    "Identification": "40230341298607",
    "Name": "MR R E DEELEY ",
    "SecondaryIdentification": "0002"
  },
  "RemittanceInformation": {
    "Reference": "Regular-Payment",
    "Unstructured": "Internal ops code 5120101"
  }
},
"Risk": {
  "PaymentContextCode": "EcommerceGoods",
  "MerchantCategoryCode": "5967",
  "MerchantCustomerIdentification": "053598653254",
  "DeliveryAddress": {
    "AddressLine": [
      "Flat 7",
      "Acacia Lodge"
    ],
    "StreetName": "Acacia Avenue",
    "BuildingNumber": "27",
    "PostCode": "GU31 2ZZ",
    "TownName": "Sparsholt",
    "CountrySubDivision": [
      "Wessex"
    ],
    "Country": "UK"
  }
}
}

```

```

"EndToEndIdentification":
"FRESCO.21302.GFX.22",
"ValidFromDate": "2017-12-
25T00:00:00+00:00",
"ValidToDate": "2018-12-
25T00:00:00+00:00",
"Frequency": "Monthly",
"Amount": {
  "Amount": "1.43",
  "Currency": "GBP"
},
"CreditorAccount": {
  "SchemeName":
"SortCodeAccountNumber",
  "Identification": "40230341298607",
  "Name": "MR R E DEELEY",
  "SecondaryIdentification": "0002"
},
"RemittanceInformation": {
  "Unstructured": "Internal ops code
5120101",
  "Reference": " Regular-Payment"
}
},
"Status": "AcceptedTechnicalValidation"
},
"Risk": {
  "PaymentContextCode": "EcommerceGoods",
  "MerchantCategoryCode": "5967",
  "MerchantCustomerIdentification":
"053598653254",
  "DeliveryAddress": {
    "StreetName": "Acacia Avenue",
    "BuildingNumber": "27",
    "PostCode": "GU31 2ZZ",
    "TownName": "Sparsholt",
    "Country": "UK",
    "AddressLine": [
      "Flat 7",
      "Acacia Lodge"
    ],
    "CountrySubDivision": ["Wessex"]
  }
},
"Links": {"Self": "/open-banking-
extensions/v2.0/mandates"},
"Meta": {}
}

```

**POST /mandate-submissions Request**

*Future Dated Mandate Example*

### Request Payload

```
POST /mandate-submissions HTTP/1.1
Authorization: Bearer Jhingapulaav
x-idempotency-key: FRESNO.1317.GFX.22
x-fapi-financial-id: CBSOpenBanking
x-fapi-customer-last-logged-time: Sun, 10 Sep 2017
19:43:31 UTC
x-fapi-customer-ip-address: 104.25.212.99
x-fapi-interaction-id: 93bac548-d2de-4546-b106-
880a5018460d
X-Client-Id: tppclientid
Content-Type: application/json
Accept: application/json

{
  "Data": {
    "PaymentId": "86e15e3f-fb80-4b45-b1ab-
c87621d59b00",
    "PaymentType": "FutureDated",
    "Initiation": {
      "InstructionIdentification": "ACME412",
      "EndToEndIdentification":
"FRESCO.21302.GFX.22",
      "FuturePaymentDateTime": "2017-12-
25T00:00:00+00:00",
      "InstructedAmount": {
        "Amount": " 1.43",
        "Currency": "GBP"
      },
      "CreditorAccount": {
        "SchemeName": "SortCodeAccountNumber",
        "Identification": "40230341298607",
        "Name": " MR R E DEELEY ",
        "SecondaryIdentification": "0002"
      },
      "RemittanceInformation": {
        "Reference": "Future-Payment",
        "Unstructured": "Internal ops code 5120101"
      }
    }
  },
  "Risk": {
    "PaymentContextCode": "EcommerceGoods",
    "MerchantCategoryCode": "5967",
    "MerchantCustomerIdentification": "053598653254",
    "DeliveryAddress": {
      "AddressLine": [
        "Flat 7",
        "Acacia Lodge"
      ],
      "StreetName": "Acacia Avenue",
      "BuildingNumber": "27",
      "PostCode": "GU31 2ZZ",
      "TownName": "Sparsholt",
      "CountrySubDivision": [
        "Wessex"
      ]
    }
  }
}
```

### Response Payload

```
HTTP/1.1 201 Created
x-fapi-interaction-id: 93bac548-d2de-4546-b106-
880a5018460d
Content-Type: application/json

{
  "Data": {
    "CreationDateTime": "2017-08-08T15:23:02Z",
    "PaymentId": "86e15e3f-fb80-4b45-b1ab-
c87621d59b00",
    "PaymentSubmissionId": "GGDGHVG",
    "Status": "AcceptedSettlementInProgress"
  },
  "Links": {
    "Self": "/open-banking-extensions/v2.0/mandate-
submissions"
  },
  "Meta": {}
}
```

```
  ],
  "Country": "UK"
}
}
```

### Regular Mandate Example

#### Request Payload

```
POST /mandate-submissions HTTP/1.1
Authorization: Bearer Jhingapulaav
x-idempotency-key: FRESNO.1317.GFX.22
x-fapi-financial-id: CBSOpenBanking
x-fapi-customer-last-logged-time: Sun, 10 Sep 2017
19:43:31 UTC
x-fapi-customer-ip-address: 104.25.212.99
x-fapi-interaction-id: 93bac548-d2de-4546-b106-
880a5018460d
Content-Type: application/json
Accept: application/json

{
  "Data": {
    "PaymentId": "86e15e3f-fb80-4b45-b1ab-
c87621d59b00",
    "PaymentType": "Regular",
    "Initiation": {
      "InstructionIdentification": "ACME412",
      "EndToEndIdentification":
"FRESCO.21302.GFX.22",
      "ValidFromDate": "2017-12-
25T00:00:00+00:00",
      "ValidToDate": "2018-12-
25T00:00:00+00:00",
      "Frequency": "Monthly",
      "InstructedAmount": {
        "Amount": "1.43",
        "Currency": "GBP"
      },
      "CreditorAccount": {
        "SchemeName": "SortCodeAccountNumber",
        "Identification": "40230341298607",
        "Name": "MR R E DEELEY ",
        "SecondaryIdentification": "0002"
      },
      "RemittanceInformation": {
        "Reference": "Regular-Payment",
        "Unstructured": "Internal ops code 5120101"
      }
    }
  },
  "Risk": {
    "PaymentContextCode": "EcommerceGoods",
    "MerchantCategoryCode": "5967",
```

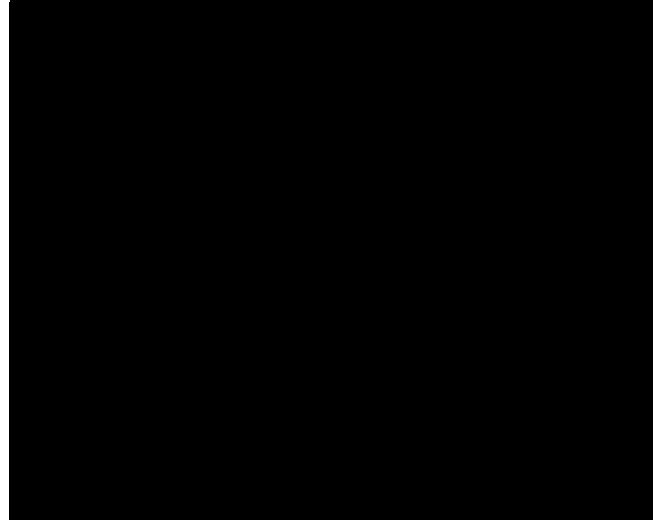
#### Response Payload

```
HTTP/1.1 201 Created
x-fapi-interaction-id: 93bac548-d2de-4546-b106-
880a5018460d
Content-Type: application/json

{
  "Data": {
    "CreationDateTime": "2017-08-08T15:23:02Z",
    "PaymentId": "86e15e3f-fb80-4b45-b1ab-
c87621d59b00",
    "PaymentSubmissionId": "GGDGHVG",
    "Status": "AcceptedSettlementInProgress"
  },
  "Links": {
    "Self": "/open-banking-extensions/v2.0/mandates"
  },
  "Meta": {
  }
}
```



```
"MerchantCustomerIdentification": "053598653254",
"DeliveryAddress": {
  "AddressLine": [
    "Flat 7",
    "Acacia Lodge"
  ],
  "StreetName": "Acacia Avenue",
  "BuildingNumber": "27",
  "PostCode": "GU31 2ZZ",
  "TownName": "Sparsholt",
  "CountrySubDivision": [
    "Wessex"
  ],
  "Country": "UK"
}
}
```



## GET /mandates Request

### *Future Dated Mandate Example*

#### Request Payload

```
GET /mandates/86e15e3f-fb80-4b45-b1ab-
c87621d59b00" HTTP/1.1
Authorization: Bearer Jhingapulaav
x-fapi-financial-id: CBSOpenBanking
x-fapi-customer-last-logged-time: Sun, 10 Sep 2017
19:43:31 UTC
x-fapi-customer-ip-address: 104.25.212.99
x-fapi-interaction-id: 93bac548-d2de-4546-b106-
880a5018460d
X-Client-Id: tppclientid
Accept: application/json
```

#### Response Payload

```
HTTP/1.1 200 OK
x-fapi-interaction-id: 93bac548-d2de-4546-b106-
880a5018460d
Content-Type: application/json

{
  "Data": {
    "PaymentId": "86e15e3f-fb80-4b45-b1ab-
c87621d59b00",
    "Status": "AcceptedTechnicalValidation",
    "CreationDateTime": "2017-06-05T15:15:19Z",
    "PaymentType": "FutureDated",
    "Initiation": {
      "InstructionIdentification": "ACME412",
      "EndToEndIdentification":
"FRESCO.21302.GFX.20",
      "FuturePaymentDateTime": "2017-12-25",
      "InstructedAmount": {
        "Amount": "165.88",
        "Currency": "GBP"
      },
    },
    "CreditorAgent": {
      "SchemeName": "BICFI",
      "Identification": "SC080800"
    },
    "CreditorAccount": {
      "Identification": "21325698",
      "Name": "ACME Inc",
      "SchemeName": "SortCodeAccountNumber",
```

```
"SecondaryIdentification": "0002"
},
  "DebtorAccount": {
    "Identification": "1254653",
    "Name": "Prince",
    "SchemeName":
"SortCodeAccountNumber",
"SecondaryIdentification": "2221"
  },
  "DebtorAgent": {
    "Identification": "54667",
    "SchemeName": "BICFI"
  },
  "RemittanceInformation": {
    "Reference": "FRESCO-101",
    "Unstructured": "Internal ops code 5120101"
  }
},
"Risk": {
  "PaymentContextCode": "EcommerceGoods",
  "MerchantCategoryCode": "5967",
  "MerchantCustomerIdentification":
"053598653254",
  "DeliveryAddress": {
    "AddressLine": [
      "Flat 7",
      "Acacia Lodge"
    ],
    "StreetName": "Acacia Avenue",
    "BuildingNumber": "27",
    "PostCode": "GU31 2ZZ",
    "TownName": "Sparsholt",
    "CountySubDivision": [
      "Wessex", "kessel"
    ],
    "Country": "UK"
  }
},
"Links": {
  "Self": "/open-banking-
extensions/v2.0/mandates/86e15e3f-fb80-4b45-
b1ab-c87621d59b00"
},
"Meta": { }
}
```

## Regular Mandate Example

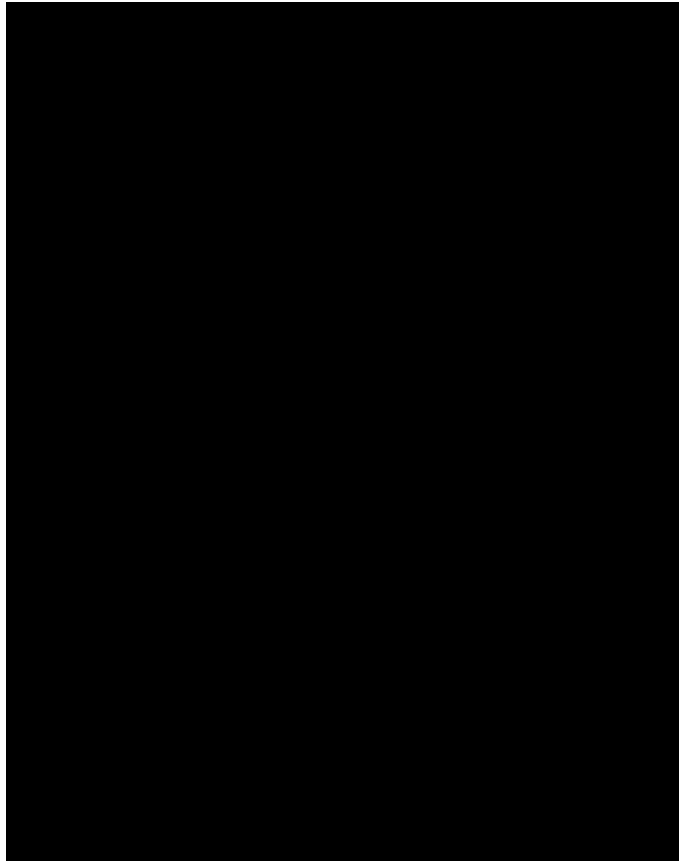
### Request Payload

```
GET /mandates/86e15e3f-fb80-4b45-b1ab-c87621d59b00 HTTP/1.1
Authorization: Bearer Jhingapulaav
x-fapi-financial-id: CBSOpenBanking
x-fapi-customer-last-logged-time: Sun, 10 Sep 2017 19:43:31 UTC
x-fapi-customer-ip-address: 104.25.212.99
x-fapi-interaction-id: 93bac548-d2de-4546-b106-880a5018460d
Accept: application/json
```

### Response Payload

```
HTTP/1.1 200 OK
x-fapi-interaction-id: 93bac548-d2de-4546-b106-880a5018460d
Content-Type: application/json

{
  "Data": {
    "PaymentId": "86e15e3f-fb80-4b45-b1ab-c87621d59b00",
    "Status": "AcceptedTechnicalValidation",
    "CreationDateTime": "2017-06-05T15:15:19Z",
    "PaymentType": "Regular",
    "Initiation": {
      "InstructionIdentification": "ACME412",
      "EndToEndIdentification": "FRESCO.21302.GFX.20",
      "ValidFromDateTime": "2017-12-25T00:00:00+00:00",
      "ValidToDateTime": "2018-12-25T00:00:00+00:00",
      "Frequency": "Monthly",
      "InstructedAmount": {
        "Amount": "165.88",
        "Currency": "GBP"
      },
      "CreditorAgent": {
        "SchemeName": "BICFI",
        "Identification": "SC080800"
      },
      "CreditorAccount": {
        "Identification": "21325698",
        "Name": "ACME Inc",
        "SchemeName": "SortCodeAccountNumber",
        "SecondaryIdentification": "0002"
      },
      "DebtorAccount": {
        "Identification": "1254653",
        "Name": "Prince",
        "SchemeName": "SortCodeAccountNumber",
        "SecondaryIdentification": "2221"
      },
      "DebtorAgent": {
        "Identification": "54667",
        "SchemeName": "BICFI"
      },
      "RemittanceInformation": {
        "Reference": "FRESCO-101",
        "Unstructured": "Internal ops code 5120101"
      }
    }
  }
}
```



```

},
"Risk": {
  "PaymentContextCode": "EcommerceGoods",
  "MerchantCategoryCode": "5967",
  "MerchantCustomerIdentification":
"053598653254",
  "DeliveryAddress": {
    "AddressLine": [
      "Flat 7",
      "Acacia Lodge"
    ],
    "StreetName": "Acacia Avenue",
    "BuildingNumber": "27",
    "PostCode": "GU31 2ZZ",
    "TownName": "Sparsholt",
    "CountySubDivision": [
      "Wessex", "kessel"
    ],
    "Country": "UK"
  }
},
"Links": {
  "Self": "/open-banking-
extensions/v2.0/mandates/86e15e3f-fb80-4b45-
b1ab-c87621d59b00"
},
"Meta": { }
}

```

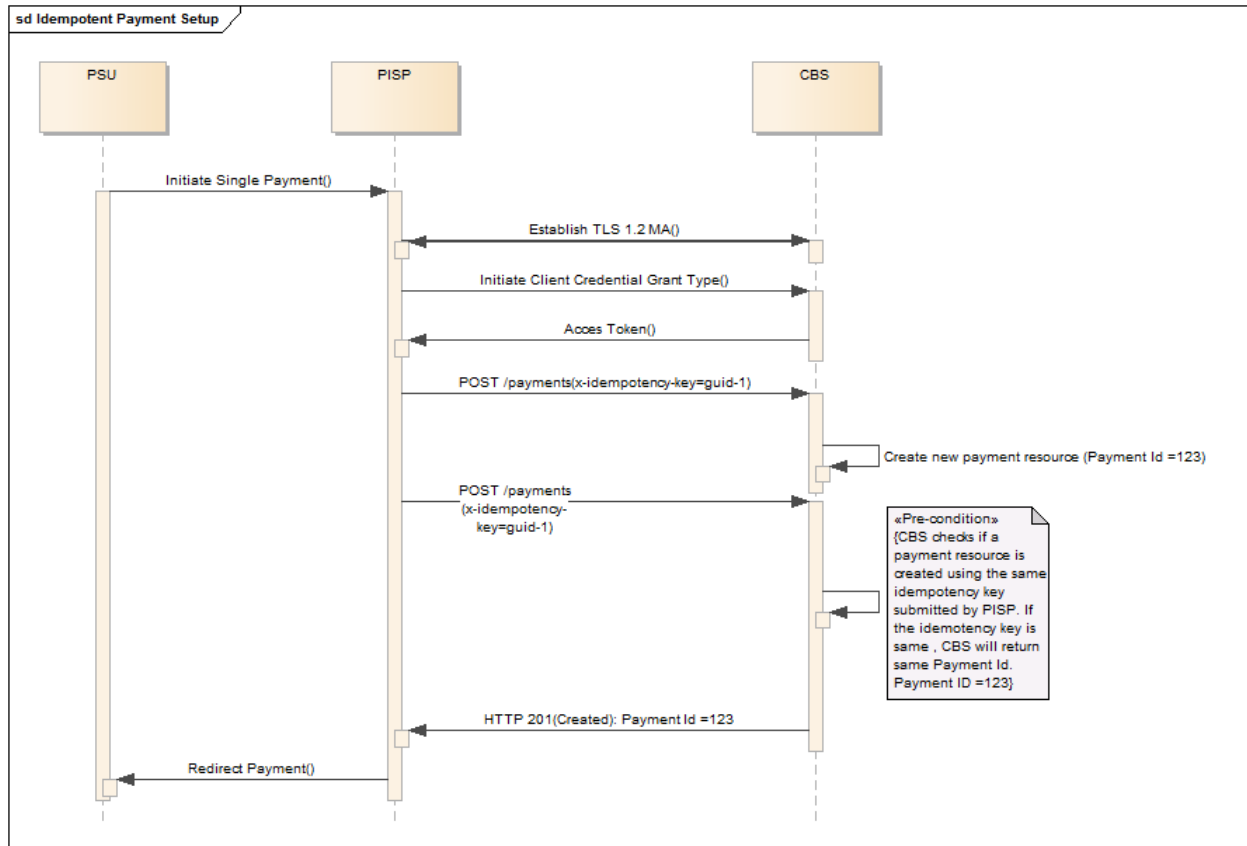
**GET /payment-submissions Request**

Request Payload
<pre> GET /payment-submissions/GGDGHVG HTTP/1.1 Authorization: Bearer 2YotnFZFEjr1zCsicMWpAA x-fapi-financial-id: CBSOpenBanking x-fapi-customer-last-logged-time: Sun, 10 Sep 2017 19:43:31 UTC x-fapi-customer-ip-address: 104.25.212.99 x-fapi-interaction-id: 93bac548-d2de-4546-b106- 880a5018460d X-Client-Id: tppclientid Accept: application/json </pre>

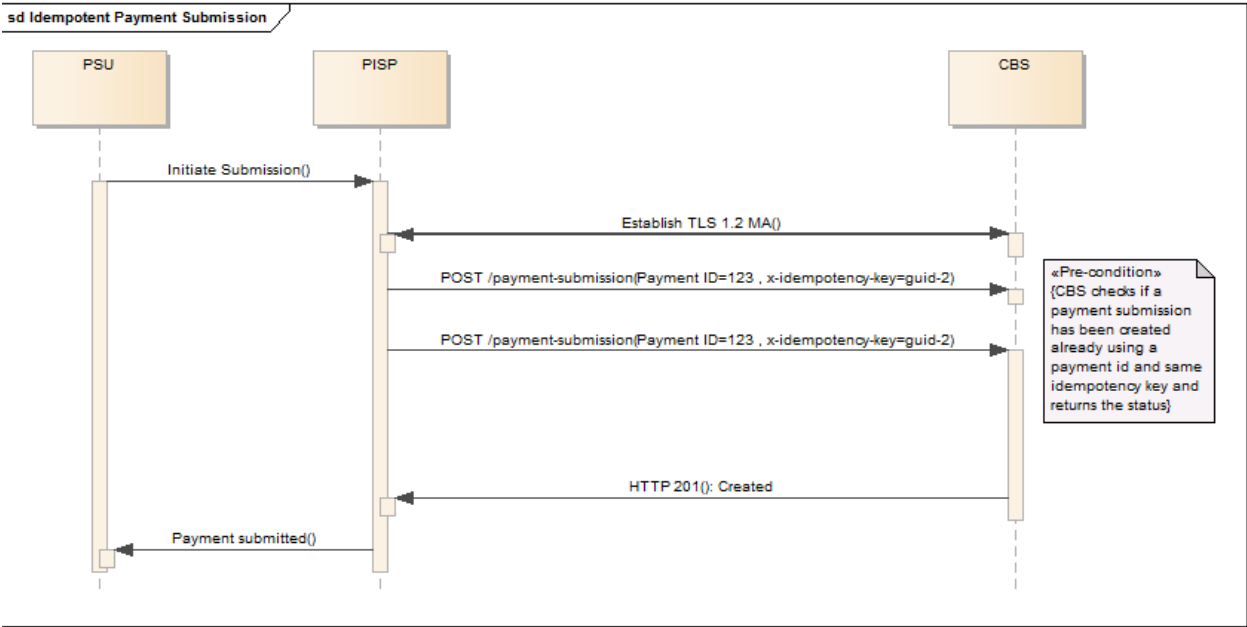
Response Payload
<pre> HTTP/1.1 200 OK Content-Type: application/json  { {   "Data": {     "PaymentSubmissionId": " GGDGHVG ",     "PaymentId": "58923",     "Status": "AcceptedSettlementInProgress",     "CreationDateTime": "2017-06-05T15:15:22+00:00"   },   "Links": {     "Self": "/ope n-banking/open-banking-extensions/v2.0/mandate- submissions/GGDGHVG "   },   "Meta": { } } </pre>

# Alternative and Error Flows

## Idempotent Payment Setup



# Idempotent Payment Submission



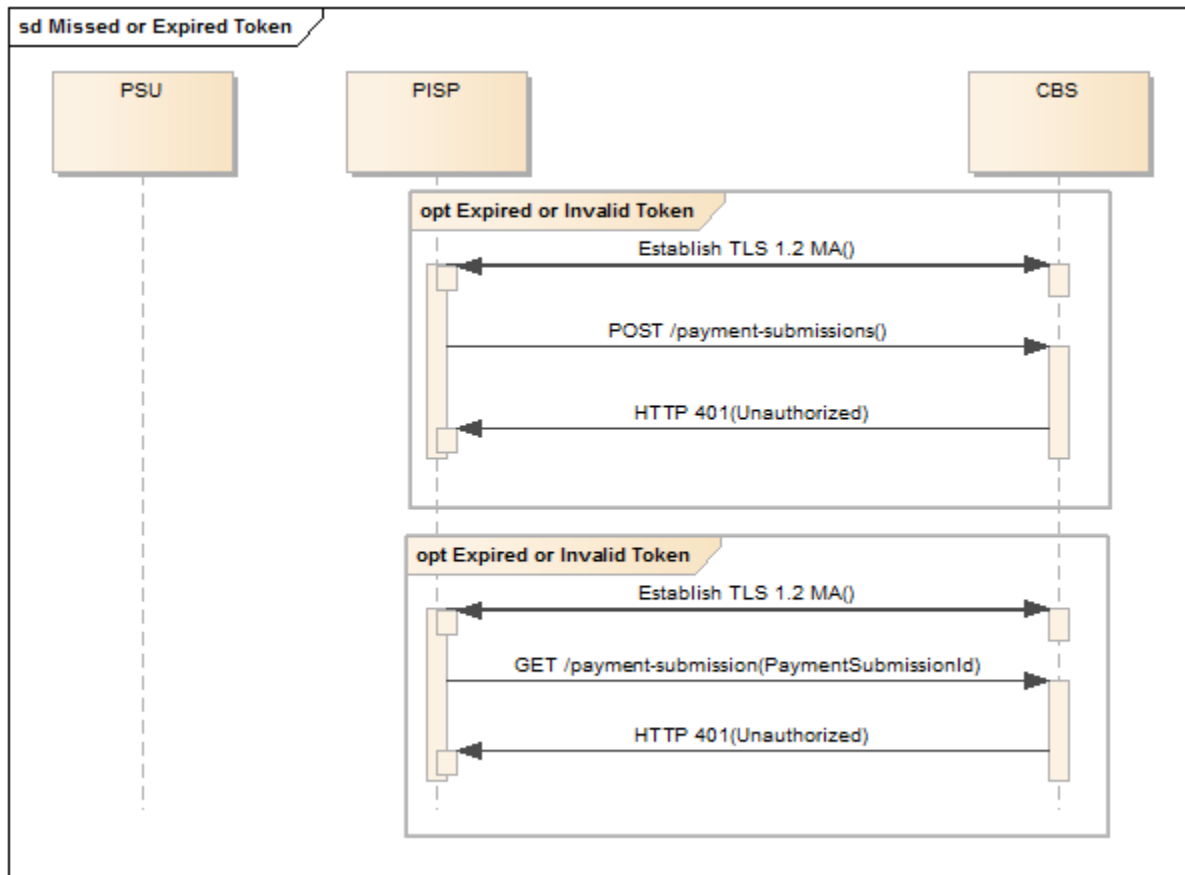
## Payment Submission Conflict - Sequence Diagram

### Missing or Expired Access Token

This flow assumes that the following Steps have been completed successfully:

- Step 1: Request Payment Initiation
- Step 2: Setup Single Payment Initiation
- Step 3: Authorize Consent

The PISP attempts to provide an expired or missing access token to CBS in an attempt to Create a Payment Submission or Get a Payment Submission Status



## Incomplete or Malformed Request Payload

This flow assumes that the following Steps have been completed successfully:

- Step 1: Request Payment Initiation
- Step 2: Setup Single Payment Initiation
- Step 3: Authorize Consent

The PISP provides a malformed request Payload to CBS in an attempt to Create a Payment Submission

