Interface Agreement
Northbound TPP

Version 2.6

Distribuzione strettamente confidenziale.

Contents

1 Introduction

Scope of the document

The document contains the Interface Agreement of the PSD2 Orchestrator.

1.1 Format

The system accept only strings in UTF-8 format. Be sure to provide in input only characters in UTF-8 format.

The application protocol is based on REST architecture and data structure is based on JSON.

All the APIs respect the following parameters:

- **Protocol:** Restful
- **Method:** Post/Get/Put/Delete (it is specified in the api)
- **Content Type:** application/json

1.2 Assumptions

Each API requires a mandatory transaction ID as input. The API user MUST provide as input a unique transactionId for each API invocation. Though a formal validation of transactionId uniqueness is not available, the transactionId must be generated to avoid collisions with other transactionId. The transactionId MUST be created appending a UUID to the component abbreviation. The same transactionId received as input by an API is used as transactionId to invoke other components.

In case of error the APIs return an error message composed by and error code and an error description. The error description is returned if and only the component is configured in Debug mode, otherwise only the error code in returned as response.

The values of payload and query parameters you can find in the examples cannot be used for real testing but they must be replaced with values consistent with the database present in the environment being used.

During the APIs invocation, if an error occurs the response will be filled with "error management" object, otherwise it will contain only response parameters/objects.

In the APIs responses if the timezone is not specified it is assumed that the date time is in UTC format.

The field **PSU-ID** rapresents a code which identifies the PSU on the ASPSP system. The PSU-ID is not equal to the user identifier used by PSU for the authentication on the ASPSP channels.

For this reason this field is not supposed to be known by PSU himself.

Thus the TPP is requested to invoke the services made available by the PSD2 Gateway for authentication or for SCA to obtain this data. Overall TPP should never use a PSU-ID that is not provided by the ASPSP.

1.3 Pagination

API SHOULD support Pagination using Offset/limit based pattern:

GET /resources?offset=4&limit=100

Meaning: get 100 resources of the 4th page.

Default value of query parameter *offset* is 1, default value of *limit* is ASPSP dependent, if not otherwise specified in the Interface Agreement of the API.

API supporting pagination return the total count of resources and pages only when invoked with offset=1, that is when the first page is requested. The total count of resources and pages will be returned in custom headers:

- *cpaas-total-elements*
- *cpaas-total-pages*

Sample request:

GET /resources

Sample response:

HTTP Headers
cpaas-total-elements *= 250*
cpaas-total-pages *= 3*

If 250 exceeds the max *limit* defined into ASPSP system, the API should return the *n* resources of the 1st page, with *n=max limit*

Sample request:

GET /resources?offset=2&limit=32

Sample response:

The API return 32 resources of the 2th page

1.4 Sorting

The sort parameter is a comma-separated list of fields to sort. To indicate sorting direction, fields my prefixed with + (ascending) or - (descending, default), e.g. /configuration-sca-methods?sort=URLEncoded(+last_update_time).

1.5 General Notes

Following is the legend on the different types of mandatory input parameters that are used in the document:

- **M** (Mandatory)
- **O** (Optional)
- **M-Private** (Mandatory only for Private Interface, for public interface is not required)
- **O-Private** (Optional only for Private Interface, for public interface is not required)
- **M-Public** (Mandatory only for Public Interface, for private interface is not required)

2 Acronyms

Term	Description
PSD2	Payments Systems Directive 2
AISP	Account Information Service Provider
AMS	Accenture Mobility Services
ASPSP	Account Servicing Payment Service Providers
CSRM	Customer Subscription & Relation Management

ISO	International Organization for Standardization
PIISP	Payment Instrument Issuer Service Provider
PISP	Payment Initiation Service Provider
PSU	Payment Service User
SCA	Strong Customer Authentication
TPP	Third Party Providers
UUID	Universally unique identifier
AIS	Account Information Service
PIS	Payment Initiation Service
PIIS	Payment Instrument Issuing Service
QTSP	Qualified Trust Service Provider
JSON	JavaScript Object Notation
UTF	Unicode Transformation Format
BG	Berlin Group

Table 1 Acronyms

3 Reference

Attachment Id	Title	File
1	BG-PSD2-Implementation_1.1.pdf	https://docs.wixstatic.com/ugd/c2914b_5351b289bf844c6881e46ee3561d95bb.pdf

Table 2 - Documents Reference

4 establishConsentServices

In this section are described the APIs to manage the PSU account and the related consents.

API	Description	Visibility	Access Token
establishCosent	Creates an account information consent resource at the ASPSP regarding access to accounts specified in this request.	Public	Application
updateConsent	This API manage the process of PSU identification, PSU authentication and explicit authorisation of transactions by using SCA or the transfer data for SCA checks by the ASPSP.	Public	Application
getConsentStatus	This API check the status of an account information consent resource.	Public	Application
getConsent	Returns the content of an account information consent object.	Public	Application
deleteConsent	The TPP can delete an account information consent object if needed.	Public	Application

There’s an example of a JSON Request/Response below every API .

4.1 establishConsent

Creates an account information consent resource at the ASPSP regarding access to accounts specified in this request.

Description:

This API allows a TPP to start a process to gather from the PSU the consent to access data of the PSU payment accounts reachable by PSD2 XS2A interfaces.

The TPP can ask to the PSU the consent to access:

- the list of the reachable PSU accounts
- the details of a PSU specific payment account
- the balances of a PSU specific payment accounts
- the payment transactions on a PSU specific payment account
- the details of a specific payment transaction on a PSU specific payment

A consent can be one-off or recurring according to the value provided for the *recurringIndicator* input parameter of this API. It is possible to request a recurring consent only to access the balances or the payment transactions on a specific account, otherwise the request will be rejected.

The PSD2 Gateway defines the following constraint about recurring consents: a TPP can't have more than 1 valid recurring consent at a time for a given PSU. Since a consent becomes valid only after a successful completion of the SCA, at that moment the other recurring consent, if present, will be replaced by the new one.

The TPP must provide the certificate, issued by the competent National Authority and qualifying the TPP as a AISP, in order to access to this API.

Notes:

- when this Consent Request is a request where the "recurringIndicator" equals "true", and if it exists already a former consent for recurring access on account information for the addressed PSU, then the former consent automatically expires as soon as the new consent request is authorised by the PSU.
- A consent can be created by a TPP to be used only one-time or multiple times. At establish phase it is allowed to request a consent with “recurringIndicator=true” according the table below. Requests not compliant with these rules will be rejcted with a validation error.

Consent for	Recurring allowed
Read Account List	No
Read Account Details	No
Read Account Balances	Yes
Read Account Transactions List	Yes[1]
Read Transaction Details	No

[1] At estabish phase it is allowed to create a recurring consent to access Account Transactions List, but at consent usage time, a such consent (recurring) doesn’t allow to retrieve transactions over 90 days. To be able to retrieve transactions over 90 days it is mandatory to use a one-off consent.

Tags: information, consent, resource, request, aspsp

PROTOCOL	HTTP
PATH (Public Exposure)	https://<ASPSP_FQDN>/platform/enabler/psd2orchestrator/ais/1.0.0/consents
METHOD	POST

Parameter description

*At least a parameter is required.

INPUT				
HEADER PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***
authorization:Bearer	The value of the access token	M-Public	-	String
psu-authorization	This token is contained only, if an OAuth2 based authentication was performed in a pre-step or an OAuth2 based SCA was performed in an preceding AIS service in the same session. Reserved for future use.	O	255	String
x-request-id	ID of the request, unique to the call, as determined by the initiating party.	M	255	String
psu-id	The ID of the PSU in the ASPSP client interface. Mandatory if “psu-corporate-id” is valorized.	O	100	String
psu-id-type	Type of the PSU-ID, needed in scenarios where PSUs have several PSU-IDs as access possibility.	O	50	String
psu-corporate-id	Identification of a Corporate, only used in a corporate context	O	100	String
psu-corporate-id-type	This is describing the type of the identification needed by the ASPSP to identify the PSU-Corporate-ID content.	O	50	String
tpp-redirect-preferred	Only “true” or “false” value is accepted. If it equals “true”, the TPP prefers a redirect over an embedded SCA approach. If it equals “false”, the TPP prefers not to be redirected for SCA. The ASPSP will then choose between the Embedded or the Decoupled SCA approach, depending on the choice of the SCA procedure by the TPP/PSU. If the parameter is not used, it will be choosen the SCA approach depending on the SCA method chosen by the TPP/PSU.	O	5	String

tpp-redirect-uri	URI of the TPP, where the transaction flow shall be redirected to after a Redirect. **This redirect link must be contained, if the tpp-redirect-preferred flag is contained and equals "true"	O**	2048	String
tpp-nok-redirect-uri	If this URI is contained, the TPP is asking to redirect the transaction flow to this address instead of the TPP-Redirect-URI in case of a negative result of the redirect SCA method. This might be ignored by the ASPSP.	O	2048	String
tpp-authentication-redirect-uri	URI of the TPP, where the transaction flow shall be redirected to after the authentication of the PSU on the ASPSP system.	O	2048	String
digest	Is contained if and only if the “Signature” element is contained in the header of the request.	O	255	String
signature	A signature of the request by the TPP on application level. This might be mandated by ASPSP. This string contains the following fields separated by commas: - *keyId*: The ‘keyId‘ field is an opaque string that the server can use to look up the component they need to validate the signature. It could be an SSH key fingerprint, a URL to machine-readable key data, an LDAP DN, etc. Management of keys and assignment of ‘keyId‘ is out of scope for this document. Serial Number of the TPP's certificate included in the “Certificate” header of this request. Serial Number of the TPP's certificate included in the “TPP-Signature-Certificate” header of this request. It shall be formatted as follows: keyId=“SN=XXX,CA=YYYYYYYYYYYYYYYY ” where “XXX” is the serial number of the certificate in hexadecimal coding given in the TPP-Signature-Certificate-Header and “YYYYYYYYYYYYYYYY” is the full Distinguished Name of the Certification Authority having produced this certificate. - algorithm: The ‘algorithm’ parameter is used to specify the digital signature algorithm to use when generating the signature. The algorithm must identify the same algorithm for the signature as presented in the certificate (Element “TPP-Certificate”) of this Request.	O	1024	String

	The available values are: "rsa-sha256" or "rsa-sha512" - Headers: The 'headers' parameter is used to specify the list of HTTP headers included when generating the signature for the message. If specified, it should be a lowercased, quoted list of HTTP header fields, separated by a single space character. If not specified, implementations MUST operate as if the field were specified with a single value, the 'Date' header, in the list of HTTP headers. Note that the list order is important, and MUST be specified in the order the HTTP header field-value pairs are concatenated together during signing. Must include - "digest", - "x-request-id", - "psu-id" (if and only if "PSU-ID" is included as a header of the HTTP-Request). - "psu-corporate-id" (if and only if "psu-corporate-id" is included as a header of the HTTP-Request). - "Date" - "tpp-redirect-uri"(if and only if "tpp-redirect-uri" is included as a header of the HTTP-Request). No other entries may be included. - Signature: The 'signature' parameter is a base 64 encoded digital signature, as described in RFC 4648 [RFC4648], Section 4. The client uses the 'algorithm' and 'headers' signature parameters to form a canonicalised 'signing string'. This 'signing string' is then signed with the key associated with 'keyId' and the algorithm corresponding to 'algorithm'. The 'signature' parameter is then set to the base 64 encoding of the signature.			
tpp-signature-certificate	This is a X509 certificate that the TPP uses for signing the request, in base64 encoding. This certificate is in PEM format without the "-----BEGIN CERTIFICATE-----" and "-----END CERTIFICATE-----". Must be contained if a signature is contained, see above.	O	4096	String
aspsp-code	The ASPSP code	M	20	String
aspsp-product-code	ASPSP product code. It must be a string without spaces and special characters.	M	45	String
date	The date provided by the TPP. Format: EEE, dd MMM yyyy hh:mm:ss z	M	31	String

BODY				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***
access	The consent identification assigned to the created resource.	M	-	Object
- accounts	Is asking for detailed account information. *At most one of these parameters is permitted.	O	-	List<Object>
○ accountId	This identification is denoting the addressed account. The account-id is the UUID related to the account structure. Its value is constant at least throughout the lifecycle of a given consent.	M	100	String
○ iban	This is an identifier used internationally by financial institutions to uniquely identify the account of a customer at a financial institution (IBANIdentifier ISO 20022). According to ISO 13616: Pattern: [A-Z]{2}[0-9]{2}[A-Z0-9]{1,30}	O*	34	String
○ bban	This data elements is used for payment accounts which have no IBAN. Specifies the Basic Bank Account Number (BBANIdentifier ISO 20022), an Identifier used nationally by financial institutions, ie, in individual countries, generally as part of a National Account Numbering Scheme(s), which uniquely identifies the account of a customer. Pattern = "[a-zA-Z0-9]{1,30}"	O*	30	String
○ pan	Primary Account Number (PAN) of a card, can be tokenized by the ASPSP due to PCI DSS requirements. This data element can be used in the body of the Consent Request Message for retrieving account access consent from this card.	O*	19	String
○ maskedPan	Primary Account Number (PAN) of a card in masked form. This data element can be used in the body of the Consent Request Message for retrieving account access consent from this card.	O*	19	String
○ msisdn	An alias to access a payment account via a registered mobile phone number. This alias might be needed e.g. in the payment initiation service. The support of this alias must be explicitly documented by the ASPSP for the corresponding API Calls.	O*	15	String
○ currency	The currency code. Codes following ISO 4217 Alpha 3	O	3	String
- balances	Is asking for balances of the addressed accounts. *At most one of these parameters is permitted.	O	-	List<Object>
○ accountId	This identification is denoting the addressed account. The account-id is the UUID related to the account structure. Its value is constant at least	M	100	String

	throughout the lifecycle of a given consent.			
○ iban	This is an identifier used internationally by financial institutions to uniquely identify the account of a customer at a financial institution (IBANIdentifier ISO 20022). According to ISO 13616: Pattern: [A-Z]{2}[0-9]{2}[A-Z0-9]{1,30}	O*	34	String
○ bban	This data elements is used for payment accounts which have no IBAN. Specifies the Basic Bank Account Number (BBANIdentifier ISO 20022), an Identifier used nationally by financial institutions, ie, in individual countries, generally as part of a National Account Numbering Scheme(s), which uniquely identifies the account of a customer. Pattern = "[a-zA-Z0-9]{1,30}"	O*	30	String
○ pan	Primary Account Number (PAN) of a card, can be tokenized by the ASPSP due to PCI DSS requirements. This data element can be used in the body of the Consent Request Message for retrieving account access consent from this card.	O*	19	String
○ maskedPan	Primary Account Number (PAN) of a card in masked form. This data element can be used in the body of the Consent Request Message for retrieving account access consent from this card.	O*	19	String
○ msisdn	An alias to access a payment account via a registered mobile phone number. This alias might be needed e.g. in the payment initiation service. The support of this alias must be explicitly documented by the ASPSP for the corresponding API Calls.	O*	15	String
○ currency	The currency code. Codes following ISO 4217 Alpha 3	O	3	String
- transactions	Is asking for transactions of the addressed accounts. *At most one of these parameters is permitted.	O	-	List<Object>
○ accountId	This identification is denoting the addressed account. The account-id is the UUID related to the account structure. Its value is constant at least throughout the lifecycle of a given consent.	M	100	String
○ iban	This is an identifier used internationally by financial institutions to uniquely identify the account of a customer at a financial institution (IBANIdentifier ISO 20022). According to ISO 13616: Pattern: [A-Z]{2}[0-9]{2}[A-Z0-9]{1,30}	O*	34	String
○ bban	This data elements is used for payment accounts which have no IBAN. Specifies the Basic Bank Account Number (BBANIdentifier ISO 20022), an Identifier used nationally by financial institutions, ie, in individual countries,	O*	30	String

	generally as part of a National Account Numbering Scheme(s), which uniquely identifies the account of a customer. Pattern = "[a-zA-Z0-9]{1,30}"			
○ pan	Primary Account Number (PAN) of a card, can be tokenized by the ASPSP due to PCI DSS requirements. This data element can be used in the body of the Consent Request Message for retrieving account access consent from this card.	O*	19	String
○ maskedPan	Primary Account Number (PAN) of a card in masked form. This data element can be used in the body of the Consent Request Message for retrieving account access consent from this card.	O*	19	String
○ msisdn	An alias to access a payment account via a registered mobile phone number. This alias might be needed e.g. in the payment initiation service. The support of this alias must be explicitly documented by the ASPSP for the corresponding API Calls.	O*	15	String
○ currency	The currency code. Codes following ISO 4217 Alpha 3	O	3	String
- availableAccounts	Only the values "allAccounts", “all-accounts” or "allAccountsWithBalances" is admitted. When this field is present, isn’t allowed to use account, balances, transactions	O	23	String
- allPsd2	Only the value "allAccounts" or “all-accounts” is admitted. *Reserved for future use*	O	12	String
recurringIndicator	Only “true” or “false” value is accepted. "true", if the consent is for recurring access to the account data; "false", if the consent is for one access to the account data. RecurringIndicator “true“ is allowed only for certains access as decribed In the table at the beginning of the paragraph.	M	5	String
validUntil	This parameter is requesting a valid until date for the requested consent. Format: YYYY-MM-DD	M	10	String
frequencyPerDay	This field indicates the requested maximum frequency for an access per day. For a one-off access, this attribute is set to "1".	M	3	Integer
combinedServiceIndicator	Only “true” or “false” value is accepted. If "true" indicates that a payment initiation service will be addressed in the same "session".	M	5	String

OUTPUT			
HEADER PARAM			
Parameter	***Description***	***Mandatory / Optional***	***Type***
location	Location of the created resource (if created)	M	String

x-request-id	ID of the request, unique to the call, as determined by the initiating party.	M	String
aspsp-sca-approach	This data element must be contained, if the SCA Approach is already fixed. Possible values are: - EMBEDDED - DECOUPLED - REDIRECT The OAuth SCA approach will be subsumed by REDIRECT.	O	String
BODY			
Parameter	***Description***	***Mandatory / Optional***	***Type***
errorManagement	Object identifying the error	O	Object
○ errorCode	Code that identifies error occurred	O	String
○ errorDescription	Error description	O	String
consentStatus	Mandatory in case of Establish Consent Process. Accepted values: - received Appendix – Consent Status	M	String
consentId	Identification of the consent resource as it is used in the API structure. Shall be contained, if a consent resource was generated.	O	String
psuCredentials	PSU Credentials on the Bank system	O	Object
- aspspProductCode	Product Identification. Used to contextualize the credentials provided by the PSU for those ASPSP that need of it.	M	String
- credentialsDetails	Credentials Details	O	List<Object>
○ credentialDetailId	Credential Detail Identification	M	String
○ isSecret	If true, it indicates that the field is a password so it should be secreted.	M	String
○ labelList	The list of the labels to show to the end user. They are internationalized.	M	List<Object>
▪ label	The label associated to the credentials to show to the end user.	M	String
▪ language	Label internationalization. It specifies the language of the label.	M	String
scaMethods	This data element might be contained, if SCA is required and if the PSU has a choice between different authentication methods. Depending on the risk management of the ASPSP this choice might be offered before or after the PSU has been identified with the first relevant factor, or if an access token is transported. If this data element is contained, then there is also an hyperlink of type "selectAuthenticationMethods" contained in the response body. These methods shall be presented towards the PSU for selection by the TPP.	O	List<Object>
- authenticationType	The field describes the type of the authentication method. Example values are: ○ SMS_OTP ○ CHIP_OTP	M	String

	○ PHOTO_OTP ○ PUSH_OTP See Appendix - AuthenticationType.		
- authenticationVersion	Depending on the "authenticationType". This version can be used by differentiating authentication tools used within performing OTP generation in the same authentication type. This version can be referred to in the ASPSP's documentation.	O	String
- authenticationMethodId	An identification provided by the ASPSP for the later identification of the authentication method selection.	M	String
- name	This could be a description provided by the ASPSP like "SMS OTP on phone +49160 xxxxx 28". This name shall be used by the TPP when presenting a list of authentication methods to the PSU, if available.	O	String
- explanation	This is a description about the authentication method.	O	String
chosenScaMethod	This data element is only contained in the response if the APSPS has chosen the Embedded SCA Approach, if the PSU is already identified with the first relevant factor or alternatively an access token, if SCA is required and if the authentication method is implicitly selected. Appendix - AuthenticationObject	O	Object
- authenticationType	The field describes the type of the authentication method. Example values are: ○ SMS_OTP ○ CHIP_OTP ○ PHOTO_OTP ○ PUSH_OTP See Appendix - AuthenticationType.	M	String
- authenticationVersion	Depending on the "authenticationType". This version can be used by differentiating authentication tools used within performing OTP generation in the same authentication type. This version can be referred to in the ASPSP's documentation.	O	String
- authenticationMethodId	An identification provided by the ASPSP for the later identification of the authentication method selection.	M	String
- name	This is the name of the authentication method defined by the PSU in the Online Banking frontend of the ASPSP. Alternatively this could be a description provided by the ASPSP like "SMS OTP on phone +49160 xxxxx 28". This name shall be used by the TPP when presenting a list of authentication methods to the PSU, if available.	O	String
- explanation	This is a description about the authentication method.	O	String
challengeData	It is containded in addition to the data element chosenScaMethod if challenge data is needed for SCA. In rare cases this attribute is also used in the context of the updatePsuAuthentication link. Appendix - Challenge	O	Object
- image	PNG data (max. 512 kilobyte) to be displayed to the PSU, Base64 encoding , cp. [RFC 4648].	O	String

	This attribute is used only, when PHOTO_OTP or CHIP_OTP is the selected SCA method.		
- data	String challenge data.	O	String
- imageLink	A link where the ASPSP will provides the challenge image for the TPP.	O	String
- otpMaxLength	The maximal length for the OTP to be typed in by the PSU.	O	Integer
- otpFormat	The format type of the OTP to be typed in. The admitted values are: • characters • integer.	O	String
- additional Information	Additional explanation for the PSU to explain e.g. fallback mechanism for the chosen SCA method. The TPP is obliged to show this to the PSU.	O	String
_links	A list of hyperlinks to be recognised by the TPP.	M	Object
- updatePsuAuthenticationRedirect	A link to an ASPSP site where the PSU authentication is performed within the Redirect authentication approach. The authentication redirect URI will be provided to the TPP encoded according to the URL encoding process that consists in encoding only the single query parameters after "?".	O	Object
○ href	This field contains a link to a resource.	M	String
- scaRedirect	A link to an ASPSP site where SCA is performed within the Redirect SCA approach.	O	Object
○ href	This field contains a link to a resource.	M	String
- scaOAuth	The link refers to a JSON document specifying the OAuth details of the ASPSP's authorisation server. JSON document follows the definition given in https://tools.ietf.org/html/draft-ietf-oauth-discovery.	O	Object
○ href	This field contains a link to a resource.	M	String
- updatePsuAuthentication	The link to the payment initiation or account information resource, which needs to be updated by a PSU password and eventually the PSU identification if not delivered yet.	O	Object
○ href	This field contains a link to a resource.	M	String
- selectAuthenticationMethod	This is a link to a resource, where the TPP can select the applicable second factor authentication methods for the PSU, if there were several available authentication methods.	O	Object
○ href	This field contains a link to a resource.	M	String
○ self	The link to the payment initiation resource created by the request itself. This link can be used later to retrieve the transaction status of the payment initiation.	O	Object
○ href	This field contains a link to a resource.	M	String
- status	Status of the resource.	O	Object
○ href	This field contains a link to a resource.	M	String
psuMessage	Text to be displayed to the PSU, e.g. in a Decoupled SCA Approach.	O	String
tppMessages	List of messages to the TPP on operational issues.	O	List<Message>
○ category	Only "ERROR" or "WARNING" permitted	M	String
○ code	The code of the error. Refers to the list of possible error code (Message code)	M	String
○ path	The path of the element of the request message which provoked this error message	O	String

○ text	Additional explaining text (max 512 characters)	O	String

HTTP Code	Result Description
201	Created

Error management		
HTTP Code	**Error code**	**Error Description**
400	PSD2_01.000.A0002	Missing header parameter: {field name}
400	PSD2_01.000.A0003	Invalid header parameter: {field name}
400	PSD2_01.000.A0008	Custom bean validation error - {field name} {condition violated}
400	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
400	PSD2_01.190.A0036	Invalid ASPSP product code
400	PSD2_01.188.A0037	No SCA methods applicable
401	PSD2_01.000.A0009	Invalid signature
401	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
403	PSD2_01.001.A0004	Unknown ASPSP
403	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
404	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
406	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
429	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
500	PSD2_00.000.A0000	Generic Error

* In this case the error is provided by the ASPSP. The http code and the TPP-Messages are defined by using the BG specification. Refers to Message Code section for details.

Example of consentRequest

POST https://<ASPSP_FQDN>/platform/enabler/psd2orchestrator/ais/1.0.0/consents

Request:

```
HEADERS:
aspsp-code=12345
content-type: application/json
x-request-id: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
date = Wed, 27 Jun 2018 13:55:51 GMT
```

```
BODY:
{
        "access": {
                "accounts": [{
                        "accountId": "710f9c01-159a-4338-a544-5171e85ebf36",
                        "maskedPan": "************0194"
                }],
                "balances": [{
                        "accountId": "710f9c01-159a-4338-a544-5171e85ebf36",
                        "maskedPan": "************0194"
                }],
                "transactions": [{
                        "accountId": "710f9c01-159a-4338-a544-5171e85ebf36",
                        "maskedPan": "************0194"
```

```
            }]
      },
      "recurringIndicator": "false",
      "validUntil": "2018-11-01",
      "frequencyPerDay": "1",
      "combinedServiceIndicator": "false"
}
```

Response:

```
HTTP Status code: 200

HEADERS:
x-request-id: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
location: "consents/1234-wertiq-983"
ASPSP-SCA-Approach: REDIRECT
```

```
Response in case of a redirect
BODY:
{
      "consentStatus": "received",
      "consentId": "1234-wertiq-983",
      "_links": {
            "scaRedirect": {
                  "href": "https://www.testbank.com/authentication/1234-
wertiq-983"
            }
      }
}

Response in case of the OAuth2 approach:
{
      "consentStatus": "received",
      "consentId": "1234-wertiq-983",
      "_links": {
            "self": {
                  "href": "/consents/1234-wertiq-983"
            }
      }
}

Response in case of the Decoupled approach:

HEADERS:
ASPSP-SCA-Approach: DECOUPLED

BODY:
{
      "consentStatus": "received",
      "consentId": "1234-wertiq-983",
      "_links": {
            "updatePsuIdentification": {
```

```
                        "href": "/consents/1234-wertiq-983"
                }
        }
}

Response in case of the Embedded approach:

HEADERS:
ASPSP-SCA-Approach: EMBEDDED

BODY:
{
        "consentStatus": "received",
        "consentId": "1234-wertiq-983",
        "_links": {
                "updatePsuAuthentication": {
                        "href": "/consents/1234-wertiq-983"
                }
        }
}
```

4.2 updateConsent

This API manage the process of PSU identification, PSU authentication and explicit authorisation of transactions by using SCA or the transfer data for SCA checks by the ASPSP.

Description:

This API must be used by the TPP to move forward a consent establish flow started through establishConsent API.

TPP can use this API to manage following scenarios that may arise due to the execution of an establishConsent:

a) authenticate the PSU at the ASPSP system in case of the ASPSP requires an EMBEDDED authentication for the aspsp-product-code specified in the establishConsent request.

b) Select the SCA method to be used to strongly authenticate the PSU in case the ASPSP makes available more than one of these methods for the aspsp-product-code specified in the establishConsent request.

c) Authorise the consent finalization providing to the ASPSP the SCA authentication data to complete the strong customer authentication process in case the ASPSP requires an EMBEDDED SCA approach for the aspsp-product-code specified in the establishConsent request.

The TPP must provide the certificate, issued by the competent National Authority and qualifying the TPP as a AISP, in order to access to this API.

Among the mandatory parameters the TPP must provide to use this API, the main ones are:

- *consent-id:* to identify the payment resource;
- *operationName:* to identify the usage scenario;
- *psuCredentials*: to authenticate the PSU to the ASPSP system (scenario a);
- *authenticationMethodId*: to select the desired SCA method (scenario b);
- *scaAuthenticationData*: to authorize the payment booking (scenario c).

Tags: consent, update, psu, data, authentication, tpp, transaction, sca, aspsp

PROTOCOL	HTTP
PATH (Public Exposure)	https://<ASPSP_FQDN>/platform/enabler/psd2orchestrator/ais/1.0.0/consents/{consent-id}
METHOD	PUT

Parameter description

INPUT				
HEADER PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***

authorization:Bearer	The value of the access token	M-Public	-	String
psu-authorization	This token is contained only, if an OAuth2 based authentication was performed in a pre-step or an OAuth2 based SCA was performed in an preceding AIS service in the same session. Reserved for future use.	O	-	String
operation-name	Operation to execute. Accepted values are: - updatePsuData - transactionAuthorisation	M	-	String
x-request-id	ID of the request, unique to the call, as determined by the initiating party.	M	255	String
psu-id	The ID of the PSU in the ASPSP client interface. Mandatory if “psu-corporate-id” is valorized.	O	100	String
psu-id-type	Type of the PSU-ID, needed in scenarios where PSUs have several PSU-IDs as access possibility.	O	50	String
psu-corporate-id	Identification of a Corporate, only used in a corporate context	O	100	String
psu-corporate-id-type	This is describing the type of the identification needed by the ASPSP to identify the PSU-Corporate-ID content.	O	50	String
digest	Is contained if and only if the “Signature” element is contained in the header of the request.	O	255	String
signature	A signature of the request by the TPP on application level. This might be mandated by ASPSP. This string contains the following fields separated by commas: - *keyId*: The ‘keyId‘ field is an opaque string that the server can use to look up the component they need to validate the signature. It could be an SSH key fingerprint, a URL to machine-readable key data, an LDAP DN, etc. Management of keys and assignment of ‘keyId‘ is out of scope for this document. Serial Number of the TPP's certificate included in the “Certificate” header of this request. Serial Number of the TPP's certificate included in the “TPP-Signature-Certificate” header of this request. It shall be formatted as follows: keyId=“SN=XXX,CA=YYYYYYYYYYYYYYYY” where “XXX” is the serial number of the certificate in hexadecimal coding given in the TPP-Signature-Certificate-Header and “YYYYYYYYYYYYYYYY” is the full Distinguished Name of the Certification Authority having produced this certificate. - algorithm: The ‘algorithm’ parameter is used to specify the digital signature algorithm to use when generating the signature. The algorithm must identify the same algorithm for the signature as presented in	O	1024	String

	the certificate (Element “TPP-Certificate”) of this Request. The available values are: “rsa-sha256” or “rsa-sha512” - Headers: The ‘headers’ parameter is used to specify the list of HTTP headers included when generating the signature for the message. If specified, it should be a lowercased, quoted list of HTTP header fields, separated by a single space character. If not specified, implementations MUST operate as if the field were specified with a single value, the ‘Date’ header, in the list of HTTP headers. Note that the list order is important, and MUST be specified in the order the HTTP header field-value pairs are concatenated together during signing. Must include - “digest”, - “x-request-id”, - “psu-id” (if and only if “PSU-ID” is included as a header of the HTTP-Request). - “psu-corporate-id” (if and only if “psu-corporate-id” is included as a header of the HTTP-Request). - “Date” - "tpp-redirect-uri"(if and only if "tpp-redirect-uri" is included as a header of the HTTP-Request). No other entries may be included. - Signature: The ‘signature’ parameter is a base 64 encoded digital signature, as described in RFC 4648 [RFC4648], Section 4. The client uses the ‘algorithm’ and ‘headers’ signature parameters to form a canonicalised ‘signing string’. This ‘signing string’ is then signed with the key associated with ‘keyId’ and the algorithm corresponding to ‘algorithm’. The ‘signature’ parameter is then set to the base 64 encoding of the signature.			
tpp-signature-certificate	This is a X509 certificate that the TPP uses for signing the request, in base64 encoding. This certificate is in PEM format without the "-----BEGIN CERTIFICATE-----" and "-----END CERTIFICATE-----". Must be contained if a signature is contained, see above.	O	4096	String
aspsp-code	The ASPSP code	M	20	String
date	The date provided by the TPP. Format: EEE, dd MMM yyyy hh:mm:ss z	M	31	String
PATH PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***
consent-id	Resource Identification of the related payment initiation.	M	255	String

BODY				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***
psuCredentials	PSU Credentials on the Bank system	O**	-	Object
- productCode	Product Identification. Used to contextualize the credentials provided by the PSU for those ASPSP that need of it.	M	255	String
- credentialsDetails	Credentials Details	M	-	List<Object>
- credentialDetailId	Credential Detail Identification	M	50	string
- credentialValue	Credential Value	M	255	String
authenticationMethodId	The authentication method ID as provided by the ASPSP.	O**	255	String
scaAuthenticationData	SCA authentication data, depending on the chosen authentication method. If the data is binary, then it is base64 encoded.	O***	2048	String

** In case of operation-name=updatePsuData then one amongs psuCredentials and authenticationMethodId must be present.
*** In case of operation-name=transactionAuthorisation then scaAuthenticationData must be present.

OUTPUT			
HEADER PARAM			
Parameter	***Description***	***Mandatory / Optional***	***Type***
x-request-id	ID of the request, unique to the call, as determined by the initiating party.	M	String
aspsp-sca-approach	Possible values are: • EMBEDDED • DECOUPLED • REDIRECT	O	String
BODY			
Parameter	***Description***	***Mandatory / Optional***	***Type***
errorManagement	Object identifying the error	O	Object
○ errorCode	Code that identifies error occurred	O	String
○ errorDescription	Error description	O	String
psuId	The PSU identifier.	O	String
psuIdType	The PSU identifier type.	O	String
psuCorporateId	The PSU corporate identifier. This field is relevant only in a corporate context.	O	String
psuCorporateIdType	The PSU corporate identifier. Might be mandated by the ASPSP in addition if the PSU-Corporate-ID is contained.	O	String
chosenScaMethod	A definition of the provided SCA method is contained, if only one authentication method is available, and if the Embedded SCA approach is chosen by the ASPSP. challengeData Challenge Conditional Challenge data might be contained, if only one authentication method is available. Appendix - AuthenticationObject	O	Authentication Object
○ authenticationType	The field describes the type of the authentication method. Example values are: ○ SMS_OTP ○ CHIP_OTP ○ PHOTO_OTP ○ PUSH_OTP	M	String

	See Appendix - AuthenticationType.		
○ authenticationVersion	Depending on the "authenticationType". This version can be used by differentiating authentication tools used within performing OTP generation in the same authentication type. This version can be referred to in the ASPSP's documentation.	O	String
○ authenticationMethodId	An identification provided by the ASPSP for the later identification of the authentication method selection.	M	String
○ name	This is the name of the authentication method defined by the PSU in the Online Banking frontend of the ASPSP. Alternatively this could be a description provided by the ASPSP like "SMS OTP on phone +49160 xxxxx 28". This name shall be used by the TPP when presenting a list of authentication methods to the PSU, if available.	O	String
○ explanation	This is a description about the authentication method.	O	String
challengeData	Challenge data might be contained, if only one authentication method is available. Appendix - Challenge	O	Challenge
- image	PNG data (max. 512 kilobyte) to be displayed to the PSU, Base64 encoding , cp. [RFC 4648]. This attribute is used only, when PHOTO_OTP or CHIP_OTP is the selected SCA method.	O	String
- data	String challenge data.	O	String
- imageLink	A link where the ASPSP will provides the challenge image for the TPP.	O	String
- otpMaxLength	The maximal length for the OTP to be typed in by the PSU.	O	Integer
- otpFormat	The format type of the OTP to be typed in. The admitted values are: • characters • integer.	O	String
- additional Information	Additional explanation for the PSU to explain e.g. fallback mechanism for the chosen SCA method. The TPP is obliged to show this to the PSU.	O	String
scaMethods	Might be contained, if several authentication methods are available. (name, type).	O	List<Object>
○ authenticationType	The field describes the type of the authentication method. Example values are: ○ SMS_OTP ○ CHIP_OTP ○ PHOTO_OTP ○ PUSH_OTP See Appendix - AuthenticationType.	M	String
○ authenticationVersion	Depending on the "authenticationType". This version can be used by differentiating authentication tools used within performing OTP generation in the same authentication type. This version can be referred to in the ASPSP's documentation.	O	String
○ authenticationMethodId	An identification provided by the ASPSP for the later identification of the authentication method selection.	M	String
○ name	This is the name of the authentication method defined by the PSU in the Online Banking frontend of	O	String

	the ASPSP. Alternatively this could be a description provided by the ASPSP like "SMS OTP on phone +49160 xxxxx 28". This name shall be used by the TPP when presenting a list of authentication methods to the PSU, if available.		
○ explanation	This is a description about the authentication method.	O	String
_links	A list of hyperlinks to be recognised by the TPP.	O	Links
○ scaRedirect	A link to an ASPSP site where SCA is performed within the Redirect SCA approach.	O	Object
○ href	This field contains a link to a resource.	M	String
○ updatePsuAuthenticationRedirect	A link to an ASPSP site where the PSU authentication is performed within the Redirect authentication approach. The authentication redirect URI will be provided to the TPP encoded according to the URL encoding process that consists in encoding only the single query parameters after "?".	O	Object
○ href	This field contains a link to a resource.	M	String
○ scaOAuth	The link refers to a JSON document specifying the OAuth details of the ASPSP's authorisation server. JSON document follows the definition given in https://tools.ietf.org/html/draft-ietf-oauth-discovery.	O	Object
○ href	This field contains a link to a resource.	M	String
○ updatePsuAuthentication	The link to the payment initiation or account information resource, which needs to be updated by a PSU password and eventually the PSU identification if not delivered yet.	O	Object
○ href	This field contains a link to a resource.	M	String
○ selectAuthenticationMethod	This is a link to a resource, where the TPP can select the applicable second factor authentication methods for the PSU, if there were several available authentication methods.	O	Object
○ href	This field contains a link to a resource.	M	String
○ authoriseTransaction	The link to the payment initiation or consent resource, where the "Transaction Authorisation"Request" is sent to. This is the link to the resource which will authorise the payment or the consent by checking the SCA authentication data within the Embedded SCA approach.	O	Object
○ href	This field contains a link to a resource.	M	String
○ self	The link to the payment initiation resource created by the request itself. This link can be used later to retrieve the transaction status of the payment initiation.	O	Object
○ href	This field contains a link to a resource.	M	String
○ status	Status of the resource.	O	Object
○ href	This field contains a link to a resource.	M	String
○ account	A link to the resource providing the details of one account	O	Object
○ href	This field contains a link to a resource.	M	String
○ balances	A link to the resource providing the balance of a dedicated account.	O	Object
○ href	This field contains a link to a resource.	M	String

○ transactions	A link to the resource providing the transaction history of a dedicated account.	O	Object
○ href	This field contains a link to a resource.	M	String
○ transactionDetails	A link to the resource providing details of a dedicated transaction.	O	Object
○ href	This field contains a link to a resource.	M	String
○ first	Navigation link for paginated account reports.	O	Object
○ href	This field contains a link to a resource.	M	String
○ next	Navigation link for paginated account reports.	O	Object
○ href	This field contains a link to a resource.	M	String
○ previous	Navigation link for paginated account reports.	O	Object
○ href	This field contains a link to a resource.	M	String
○ last	Navigation link for paginated account reports.	O	Object
○ href	This field contains a link to a resource.	M	String
○ download	Download link for huge AIS data packages.	O	Object
○ href	This field contains a link to a resource.	M	String
transactionStatus	The values defined in the chapter Appendix – Transaction Status might be used.	O	String
consentStatus	Mandatory in case of Establish Consent Process. Accepted values: - received - rejected - valid Appendix – Consent Status	O	String
psuMessage	Message to the PSU	O	String
tppMessages	List of messages to the TPP on operational issues.	O	List<Message>
○ category	Only "ERROR" or "WARNING" permitted	M	String
○ code	The code of the error. Refers to the list of possible error code (Message code)	M	String
○ path	The path of the element of the request message which provoked this error message	O	String
○ text	Additional explaining text (max 512 characters)	O	String
psuCredentials	PSU Credentials on the Bank system	O	Object
○ aspspProductCode	Product Identification. Used to contextualize the credentials provided by the PSU for those ASPSP that need of it.	M	String
○ credentialsDetails	Credentials Details	O	List<Object>
○ credentialDetailId	Credential Detail Identification	M	String
○ isSecret	If true, it indicates that the field is a password so it should be secreted.	M	String
○ labelList	The list of the labels to show to the end user. They are internationalized.	M	List<Object>
▪ label	The label associated to the credentials to show to the end user.	M	String
▪ language	Label internationalization. It specifies the language of the label.	M	String

HTTP Code	Result Description
201	Created

Error management		
HTTP Code	**Error code**	**Error Description**
400	PSD2_01.000.A0002	Missing header parameter: {field name}
400	PSD2_01.000.A0003	Invalid header parameter: {field name}
400	PSD2_01.000.A0008	Custom bean validation error - {field name} {condition violated}
400	PSD2_01.190.A0018	Inconsistent consent resource status
400	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
401	PSD2_01.000.A0009	Invalid signature
401	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
403	PSD2_01.001.A0004	Unknown ASPSP
403	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
403	PSD2_01.188.A0024	Resource expired
404	PSD2_01.190.A0010	Entity not found
404	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
406	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
409	PSD2_01.000.A0001	Operation not allowed
429	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
500	PSD2_00.000.A0000	Generic Error

* In this case the error is provided by the ASPSP. The http code and the TPP-Messages are defined by using the BG specification. Refers to Message Code section for details.

Example of updateConsentResource

PUT https://<IAM_DNS>/platform/enabler/psd2orchestrator/ais/1.0.0/consents/qwer3456tzui7890

Request:

```
HEADERS:
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
aspsp-code=12345
PSU-ID: PSU-1234
date = Wed, 27 Jun 2018 13:55:51 GMT
```

```
Request in case of updatePsuData.

HEADERS:
operation-name: updatePsuData

BODY:
{
        "psuData": {
                "password": "start12"
        }
}

Request in case of transactionAuthorisation.

HEADERS:
operation-name: transactionAuthorisation

BODY:
{
      "scaAuthenticationData": "123456"
}
```

Response:

```
HTTP Status code: 201

HEADERS:
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
ASPSP-SCA-Approach: EMBEDED
```

```
Response in case of updatePsuData.
BODY:
{
      “transactionStatus”: “ACTC”,
      “_links”:{
            “authoriseTransaction”: {
                  “href”: “/v1/payments/sepa-credit-transfers/1234-wertiq-983”
            }
      }
}

Response in case of transactionAuthorisation.
BODY:
{
      “consentStatus”: “valid
}
```

4.3 getConsentStatus

This API check the status of an account information consent resource.

Description:
Using this API, the TPP can retrieve the consent status of a previously established consent. The API gives back also the SCA status and the PSU Authentication Status of the related consent resource managed by the PSD2 Gateway.
This API may be used by a TPP especially in cases where the consent was directly managed between the ASPSP and the PSU e.g.: in a re-direct SCA approach.

The TPP must provide the certificate, issued by the competent National Authority and qualifying the TPP as a AISP, in order to access to this API.

Among the mandatory parameters the TPP must provide to use this API, the main ones are:

- *consent-id:* to identify the consent resource.

Tags: account, information, consent, status

PROTOCOL	HTTP
PATH (Public Exposure)	https://<ASPSP_FQDN>/platform/enabler/psd2orchestrator/ais/1.0.0/consents/{consent-id}/status
METHOD	GET

Parameter description

INPUT				
HEADER PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***
authorization:Bearer	The value of the access token	M-Public	-	String
psu-authorization	This token is contained only, if an OAuth2 based authentication was performed in a pre-step or an OAuth2 based SCA was performed in an preceding AIS service in the same session. Reserved for future use.	O	-	String
x-request-id	ID of the request, unique to the call, as determined by the initiating party.	M	255	String
digest	Is contained if and only if the "Signature" element is contained in the header of the request.	O	255	String
signature	A signature of the request by the TPP on application level. This might be mandated by ASPSP. This string contains the following fields separated by commas:	O	1024	String

	- *keyId*: The ‘keyId‘ field is an opaque string that the server can use to look up the component they need to validate the signature. It could be an SSH key fingerprint, a URL to machine-readable key data, an LDAP DN, etc. Management of keys and assignment of ‘keyId‘ is out of scope for this document. Serial Number of the TPP's certificate included in the “Certificate” header of this request. Serial Number of the TPP's certificate included in the “TPP-Signature-Certificate” header of this request. It shall be formatted as follows: keyId=“SN=XXX,CA=YYYYYYYYYYYYYYYY” where “XXX” is the serial number of the certificate in hexadecimal coding given in the TPP-Signature-Certificate-Header and “YYYYYYYYYYYYYYYY” is the full Distinguished Name of the Certification Authority having produced this certificate. - algorithm: The ‘algorithm’ parameter is used to specify the digital signature algorithm to use when generating the signature. The algorithm must identify the same algorithm for the signature as presented in the certificate (Element “TPP-Certificate”) of this Request. The available values are: “rsa-sha256” or “rsa-sha512” - Headers: The ‘headers’ parameter is used to specify the list of HTTP headers included when generating the signature for the message. If specified, it should be a lowercased, quoted list of HTTP header fields, separated by a single space character. If not specified, implementations MUST operate as if the field were specified with a single value, the ‘Date’ header, in the list of HTTP headers. Note that the list order is important, and MUST be specified in the order the HTTP header field-value pairs are concatenated together during signing. Must include - “digest”, - “x-request-id”, - “psu-id” (if and only if “PSU-ID” is included as a header of the HTTP-Request). - “psu-corporate-id” (if and only if “psu-corporate-id” is included as a header of the HTTP-Request). - “Date” - "tpp-redirect-uri"(if and only if "tpp-redirect-uri" is included as a header of the HTTP-Request). No other entries may be included.			

	- Signature: The 'signature' parameter is a base 64 encoded digital signature, as described in RFC 4648 [RFC4648], Section 4. The client uses the 'algorithm' and 'headers' signature parameters to form a canonicalised 'signing string'. This 'signing string' is then signed with the key associated with 'keyId' and the algorithm corresponding to 'algorithm'. The 'signature' parameter is then set to the base 64 encoding of the signature.			
tpp-signature-certificate	This is a X509 certificate that the TPP uses for signing the request, in base64 encoding. This certificate is in PEM format without the "-----BEGIN CERTIFICATE-----" and "-----END CERTIFICATE-----". Must be contained if a signature is contained, see above.	O	4096	String
aspsp-code	The ASPSP code	M	20	String
date	The date provided by the TPP. Format: EEE, dd MMM yyyy hh:mm:ss z	M	31	String
PATH PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***
consent-id	The consent identification assigned to the created resource.	M	255	String

OUTPUT			
HEADER PARAM			
Parameter	***Description***	***Mandatory / Optional***	***Type***
x-request-id	ID of the request, unique to the call, as determined by the initiating party.	M	String
psu-id	The PSU identifier.	O	String
psu-id-type	The PSU identifier type.	O	String
psu-corporate-id	The PSU corporate identifier. This field is relevant only in a corporate context.	O	String
psu-corporate-id-type	The PSU corporate identifier. Might be mandated by the ASPSP in addition if the PSU-Corporate-ID is contained.	O	String
BODY			
Parameter	***Description***	***Mandatory / Optional***	***Type***
errorManagement	Object identifying the error	O	Object
○ errorCode	Code that identifies error occurred	O	String
○ errorDescription	Error description	O	String
consentStatus	Mandatory in case of Establish Consent Process. Accepted values: - received - rejected - valid - revokedByPsu - expired - terminatedByTpp - replaced - invalidated - pendingExpired	M	String

	Appendix – Consent Status		
scaStatus	This data element is containing information about the status of the SCA method applied.	O	String
psuAuthenticationStatus	his data element is containing information about the status of the authentication of the PSU. Allowed values: - 'IDENTIFICATION_REQUIRED' (The authentication is required in order to identify the PSU and to retrieve the PSU-ID. Typically this state occurs when the TPP doesn't send the PSU-ID as input parameter into the paymentInitiation Request) - 'AUTHENTICATION_REQUIRED' (The psu authentication is requested in order to proceed the execution of the payment request) - 'AUTHENTICATED' (PSU successfully authenticated) 'AUTHENTICATION_FAILED' (PSU authentication failed)	O	String
psuMessage	Message to the PSU	O	String
tppMessages	List of messages to the TPP on operational issues.	O	List<Message>
○ category	Only "ERROR" or "WARNING" permitted	M	String
○ code	The code of the error. Refers to the list of possible error code (Message code)	M	String
○ path	The path of the element of the request message which provoked this error message	O	String
○ text	Additional explaining text (max 512 characters)	O	String

HTTP Code	Result Description
200	Service executed successfully

Error management		
HTTP Code	Error code	Error Description
400	PSD2_01.000.A0002	Missing header parameter: {field name}
400	PSD2_01.000.A0003	Invalid header parameter: {field name}
400	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
401	PSD2_01.000.A0009	Invalid signature
401	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
403	PSD2_01.001.A0004	Unknown ASPSP
403	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
404	PSD2_01.190.A0010	Entity not found
404	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
406	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
409	PSD2_01.000.A0001	Operation not allowed
429	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
500	PSD2_00.000.A0000	Generic Error

* In this case the error is provided by the ASPSP. The http code and the TPP-Messages are defined by using the BG specification. Refers to Message Code section for details.

Example of getStatusRequestForConsent

GET https://<ASPSP_FQDN>/platform/enabler/psd2orchestrator/ais/1.0.0/consents/qwer3456tzui7890/status

Request:

```
HEADERS:
aspsp-code=12345
content-type: application/json
x-request-id: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
date = Wed, 27 Jun 2018 13:55:51 GMT
```

```
BODY:
N/A
```

Response:

```
HTTP Status code: 200

HEADERS:
x-request-id: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
```

```
BODY:
{
      “consentStatus” : “valid”
}
```

4.4 getConsent

Returns the content of an account information consent object.

Description:
Using this API, the TPP can retrieve the data received by the PSD2 Gateway in the establishConsent used to generate the consent identified by the consentId given as input in this API.

The TPP must provide the certificate, issued by the competent National Authority and qualifying the TPP as a AISP, in order to access to this API.

Among the mandatory parameters the TPP must provide to use this API, the main ones are:

- *consent-id: to identify the consent resource.*

Tags: get, consent, request, aspsp, tpp, psu

PROTOCOL	HTTP
PATH (Public Exposure)	https://<ASPSP_FQDN>/platform/enabler/psd2orchestrator/ais/1.0.0/consents/{consent-id}
METHOD	GET

Parameter description

INPUT				
HEADER PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***
authorization:Bearer	The value of the access token	M-Public	-	String
psu-authorization	This token is contained only, if an OAuth2 based authentication was performed in a pre-step or an OAuth2 based SCA was performed in an preceding AIS service in the same session. Reserved for future use.	O	-	String
x-request-id	ID of the request, unique to the call, as determined by the initiating party.	M	255	String
digest	Is contained if and only if the "Signature" element is contained in the header of the request.	O	255	String
signature	A signature of the request by the TPP on application level. This might be mandated by ASPSP. This string contains the following fields separated by commas: - *keyId*: The 'keyId' field is an opaque string that the server can use to look up the component they need to validate the	O	1024	String

	signature. It could be an SSH key fingerprint, a URL to machine-readable key data, an LDAP DN, etc. Management of keys and assignment of 'keyId' is out of scope for this document. Serial Number of the TPP's certificate included in the "Certificate" header of this request. Serial Number of the TPP's certificate included in the "TPP-Signature-Certificate" header of this request. It shall be formatted as follows: keyId="SN=XXX,CA=YYYYYYYYYYYYYYYY" where "XXX" is the serial number of the certificate in hexadecimal coding given in the TPP-Signature-Certificate-Header and "YYYYYYYYYYYYYYYY" is the full Distinguished Name of the Certification Authority having produced this certificate. - algorithm: The 'algorithm' parameter is used to specify the digital signature algorithm to use when generating the signature. The algorithm must identify the same algorithm for the signature as presented in the certificate (Element "TPP-Certificate") of this Request. The available values are: "rsa-sha256" or "rsa-sha512" - Headers: The 'headers' parameter is used to specify the list of HTTP headers included when generating the signature for the message. If specified, it should be a lowercased, quoted list of HTTP header fields, separated by a single space character. If not specified, implementations MUST operate as if the field were specified with a single value, the 'Date' header, in the list of HTTP headers. Note that the list order is important, and MUST be specified in the order the HTTP header field-value pairs are concatenated together during signing. Must include - "digest", - "x-request-id", - "psu-id" (if and only if "PSU-ID" is included as a header of the HTTP-Request). - "psu-corporate-id" (if and only if "psu-corporate-id" is included as a header of the HTTP-Request). - "Date" - "tpp-redirect-uri"(if and only if "tpp-redirect-uri" is included as a header of the HTTP-Request). No other entries may be included. - Signature: The 'signature' parameter is a base 64 encoded digital signature, as			

	described in RFC 4648 [RFC4648], Section 4. The client uses the 'algorithm' and 'headers' signature parameters to form a canonicalised 'signing string'. This 'signing string' is then signed with the key associated with 'keyId' and the algorithm corresponding to 'algorithm'. The 'signature' parameter is then set to the base 64 encoding of the signature.			
tpp-signature-certificate	This is a X509 certificate that the TPP uses for signing the request, in base64 encoding. This certificate is in PEM format without the "-----BEGIN CERTIFICATE-----" and "-----END CERTIFICATE-----". Must be contained if a signature is contained, see above.	O	4096	String
aspsp-code	The ASPSP code	M	20	String
date	The date provided by the TPP. Format: EEE, dd MMM yyyy hh:mm:ss z	M	31	String
PATH PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***
consent-id	The consent identification as returned by an Account Information Consent Request	M	255	String

OUTPUT			
HEADER PARAM			
Parameter	***Description***	***Mandatory / Optional***	***Type***
x-request-id	ID of the request, unique to the call, as determined by the initiating party.	M	String
BODY			
Parameter	***Description***	***Mandatory / Optional***	***Type***
errorManagement	Object identifying the error	O	Object
○ errorCode	Code that identifies error occurred	O	String
○ errorDescription	Error description	O	String
access	Requested access services.	M	Object
○ accounts	Is asking for detailed account information. If the array is empty, the TPP is asking for an accessible account list. This may be restricted in a PSU/ASPSP authorization dialogue.	O	List<Object>
○ iban	This is an identifier used internationally by financial institutions to uniquely identify the account of a customer at a financial institution (IBANIdentifier ISO 20022). According to ISO 13616: Pattern: [A-Z]{2}[0-9]{2}[A-Z0-9]{1,30}	O	String
○ bban	This data elements is used for payment accounts which have no IBAN. Specifies the Basic Bank Account Number (BBANIdentifier ISO 20022), an Identifier used nationally by financial institutions, ie, in individual countries, generally as part of a National Account Numbering Scheme(s), which uniquely identifies the account of a customer. Pattern = "[a-zA-Z0-9]{1,30}"	O	String
○ pan	Primary Account Number (PAN) of a card, can be tokenized by the ASPSP due to PCI DSS requirements. This data element can be used in	O	String

	the body of the Consent Request Message for retrieving account access consent from this card.		
○ maskedPan	Primary Account Number (PAN) of a card in masked form. This data element can be used in the body of the Consent Request Message for retrieving account access consent from this card.	O	String
○ msisdn	An alias to access a payment account via a registered mobile phone number. This alias might be needed e.g. in the payment initiation service. The support of this alias must be explicitly documented by the ASPSP for the corresponding API Calls.	O	String
○ currency	The currency code. Codes following ISO 4217 Alpha 3	O	String
○ balances	Is asking for balances of the addressed accounts. If the array is empty, the TPP is asking for the balances of all accessible account lists. This may be restricted in a PSU/ASPSP authorization dialogue.	O	List<Object>
○ iban	This is an identifier used internationally by financial institutions to uniquely identify the account of a customer at a financial institution (IBANIdentifier ISO 20022). According to ISO 13616: Pattern: [A-Z]{2}[0-9]{2}[A-Z0-9]{1,30}	O	String
○ bban	This data elements is used for payment accounts which have no IBAN. Specifies the Basic Bank Account Number (BBANIdentifier ISO 20022), an Identifier used nationally by financial institutions, ie, in individual countries, generally as part of a National Account Numbering Scheme(s), which uniquely identifies the account of a customer. Pattern = "[a-zA-Z0-9]{1,30}"	O	String
○ pan	Primary Account Number (PAN) of a card, can be tokenized by the ASPSP due to PCI DSS requirements. This data element can be used in the body of the Consent Request Message for retrieving account access consent from this card.	O	String
○ maskedPan	Primary Account Number (PAN) of a card in masked form. This data element can be used in the body of the Consent Request Message for retrieving account access consent from this card.	O	String
○ msisdn	An alias to access a payment account via a registered mobile phone number. This alias might be needed e.g. in the payment initiation service. The support of this alias must be explicitly documented by the ASPSP for the corresponding API Calls.	O	String
○ currency	The currency code. Codes following ISO 4217 Alpha 3	O	String
○ transactions	Is asking for transactions of the addressed accounts. If the array is empty, the TPP is asking for the transactions of all accessible account lists. This	O	List<Object>

	may be restricted in a PSU/ASPSP authorization dialogue.		
○ iban	This is an identifier used internationally by financial institutions to uniquely identify the account of a customer at a financial institution (IBANIdentifier ISO 20022). According to ISO 13616: Pattern: [A-Z]{2}[0-9]{2}[A-Z0-9]{1,30}	O	String
○ bban	This data elements is used for payment accounts which have no IBAN. Specifies the Basic Bank Account Number (BBANIdentifier ISO 20022), an Identifier used nationally by financial institutions, ie, in individual countries, generally as part of a National Account Numbering Scheme(s), which uniquely identifies the account of a customer. Pattern = "[a-zA-Z0-9]{1,30}"	O	String
○ pan	Primary Account Number (PAN) of a card, can be tokenized by the ASPSP due to PCI DSS requirements. This data element can be used in the body of the Consent Request Message for retrieving account access consent from this card.	O	String
○ maskedPan	Primary Account Number (PAN) of a card in masked form. This data element can be used in the body of the Consent Request Message for retrieving account access consent from this card.	O	String
○ msisdn	An alias to access a payment account via a registered mobile phone number. This alias might be needed e.g. in the payment initiation service. The support of this alias must be explicitly documented by the ASPSP for the corresponding API Calls.	O	String
○ currency	The currency code. Codes following ISO 4217 Alpha 3	O	String
○ availableAccounts	Only the values "allAccounts", “all-accounts” or "allAccountsWithBalances" is admitted.	O	String
○ allPsd2	Only the value "allAccounts" or “all-accounts” is admitted.	O	String
recurringIndicator	"true", if the consent is for recurring access to the account data; "false", if the consent is for one access to the account data.	M	Boolean
validUntil	This parameter is requesting a valid until date for the requested consent. Format: YYYY-MM-DD	M	String
frequencyPerDay	This field indicates the requested maximum frequency for an access per day. For a one-off access, this attribute is set to "1".	M	Integer
lastActionDate	This date is containing the date of the last action on the consent object either through the XS2A interface or the PSU/ASPSP interface having an impact on the status. Format: YYYY-MM-DD	M	String
consentStatus	The status of the consent resource. Accepted vales: - received - rejected - valid	M	String

	- revokedByPsu - expired - terminatedByTpp - replaced - invalidated - pendingExpired Appendix – Consent Status		
scaStatus	Status information of the SCA in Decoupled or Redirect SCA Approach.	O	String
tppMessages	List of messages to the TPP on operational issues.	O	List<Message>
○ category	Only "ERROR" or "WARNING" permitted	M	String
○ code	The code of the error. Refers to the list of possible error code (Message code)	M	String
○ path	The path of the element of the request message which provoked this error message	O	String
○ text	Additional explaining text (max 512 characters)	O	String

* One of the creditor/debtor account is mandatory.

HTTP Code	Result Description
200	Service executed successfully

Error management		
HTTP Code	**Error code**	**Error Description**
400	PSD2_01.000.A0002	Missing header parameter: {field name}
400	PSD2_01.000.A0003	Invalid header parameter: {field name}
400	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
401	PSD2_01.000.A0009	Invalid signature
401	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
403	PSD2_01.001.A0004	Unknown ASPSP
403	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
404	PSD2_01.190.A0010	Entity not found
409	PSD2_01.000.A0001	Operation not allowed
500	PSD2_00.000.A0000	Generic Error

* In this case the error is provided by the ASPSP. The http code and the TPP-Messages are defined by using the BG specification.

Refers to Message Code section for details.

Example of getConsentRequest

GET https://<IAM_DNS>/platform/enabler/psd2orchestrator/ais/1.0.0/consents/qwer3456tzui7890

Request:

```
HEADERS:
aspsp-code=12345
content-type: application/json
x-request-id: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
date = Wed, 27 Jun 2018 13:55:51 GMT
```

```
BODY:
N/A
```

Response:

```
HTTP Status code: 200

HEADERS:
x-request-id: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
```

```
BODY:
{
    "access": {
        "accounts": [
            {
                "msisdn": "34755667789"
            }
        ],
        "balances": [
            {
                "iban": "DE40100100103307118608",
                "currency": "EUR"
            }
        ],
        "transactions": [
            {
                "iban": "DE40100100103307118608"
            },
            {
                "maskedPan": "1234********1234",
                "currency": "USD"
            }
        ]
    },
    "recurringIndicator": "true",
    "validUntil": "2017-01-02",
    "frequencyPerDay": "4",
    "lastActionDate": "2018-08-21",
    "scaStatus": "EXEMPT"
}
```

4.5 deleteConsent

The TPP can delete an account information consent object if needed.

Description:
Using this API, the TPP can delete a previously established consent through the establishConsent API. A TPP can delete a consent just after its creation and until its expiration unless if the consent wouldn't be revoked by the PSU through the ASPSP provided interfaces.

The TPP must provide the certificate, issued by the competent National Authority and qualifying the TPP as a AISP, in order to access to this API.

Among the mandatory parameters the TPP must provide to use this API, the main ones are:

- *consent-id: to identify the consent resource.*

Tags: delete, consent

PROTOCOL	HTTP
PATH (Public Exposure)	https://<ASPSP_FQDN>/platform/enabler/psd2orchestrator/ais/1.0.0/consents/{consent-id}
METHOD	DELETE

Parameter description

INPUT				
HEADER PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***
psu-authorization	This token is contained only, if an OAuth2 based authentication was performed in a pre-step or an OAuth2 based SCA was performed in an preceding AIS service in the same session. Reserved for future use.	O	-	String
authorization:Bearer	The value of the access token	M	-	String
x-request-id	ID of the request, unique to the call, as determined by the initiating party.	M	255	String
aspsp-code	The ASPSP code	M	20	String
PATH PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***
consent-id	Contains the resource-ID of the consent to be deleted.	M	255	String

OUTPUT			
HEADER PARAM			
Parameter	***Description***	***Mandatory / Optional***	***Type***
x-request-id	ID of the request, unique to the call, as determined by the initiating party.	M	String
BODY			
Parameter	***Description***	***Mandatory / Optional***	***Type***
errorManagement	Object identifying the error	O	Object
○ errorCode	Code that identifies error occurred	O	String
○ errorDescription	Error description	O	String
tppMessages	List of messages to the TPP on operational issues.	O	List<Message>
○ category	Only "ERROR" or "WARNING" permitted	M	String
○ code	The code of the error. Refers to the list of possible error code (Message code)	M	String
○ path	The path of the element of the request message which provoked this error message	O	String
○ text	Additional explaining text (max 512 characters)	O	String

HTTP Code	Result Description
204	Service executed successfully

Error management		
HTTP Code	**Error code**	**Error Description**
400	PSD2_01.000.A0002	Missing header parameter: {field name}
400	PSD2_01.000.A0003	Invalid header parameter: {field name}
400	PSD2_01.000.A0008	Custom bean validation error - {field name} {condition violated}
400	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
401	PSD2_01.000.A0009	Invalid signature
401	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
403	PSD2_01.001.A0004	Unknown ASPSP
403	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
403	PSD2_01.188.A0024	Resource expired
404	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
406	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
409	PSD2_01.000.A0001	Operation not allowed
429	PSD2_00.190.A0005*	ASPSP provided error. Reference to the TPP-messages.
500	PSD2_00.000.A0000	Generic Error

* In this case the error is provided by the ASPSP. The http code and the TPP-Messages are defined by using the BG specification. Refers to Message Code section for details.

Example of deleteConsent

DELETE https://<ASPSP_FQDN>/platform/enabler/psd2orchestrator/ais/1.0.0/consents/qwer3456tzui7890

Request:

```
HEADERS:
aspsp-code=12345
```

```
Content-Type: application/json
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
date: Sun, 13 Aug 2017 17:05:37 GMT
```

```
BODY:
N/A
```

Response:

```
HTTP Status code: 204

HEADERS:
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
```

```
BODY:
N/A
```

5 cardAccountInformationServices

In this section are described the APIs to manage the PSU account and the related consents.

API	Description	Visibility	Access Token
readCardAccountList	Reads a list of card accounts with additional information, e.g. balance information. It is assumed that a consent of the PSU to this access is already given and stored on the ASPSP system.	Public	Application
readCardAccountDetails	Reads details about a card account. It is assumed that a consent of the PSU to this access is already given and stored on the ASPSP system.	Public	Application
readCardAccountBalance	Reads balance data from a given card account addressed by *"account-id".*	Public	Application
readCardAccountTransactionList	Reads account data from a given account addressed by *account-id*.	Public	Application

There's an example of a JSON Request/Response below every API .

5.1 readCardAccountList

Reads a list of card accounts with additional information, e.g. balance information. It is assumed that a consent of the PSU to this access is already given and stored on the ASPSP system.

Description:

This API allows a TPP to let the list of the PSU card accounts be reachable by PSD2 XS2A interfaces. Through this API it is also possible to know the balance of the accounts, in case the consent used to invoke the API allows it and the ASPSP supports this feature.

The consent needed to use this API must be a one-off consent, so it can be used both in attended or unattended mode but only once in total. In case the card accounts list would exceed the maximum allowed number of accounts contained in a response page, the TPP must ask the PSU for a new consent to access the rest of the accounts.

The response pagination is a feature that each ASPSP can choose to support or not.

The TPP must provide the certificate, issued by the competent National Authority and qualifying the TPP as a AISP, in order to access to this API.

Among the mandatory parameters the TPP must provide to use this API, the main ones are:

- *consent-id*: to identify the Account Information Consent the TPP wants use to access this service.

Tags: read, bank, card, account, data, balance, get

PROTOCOL	HTTP
PATH (Public Exposure)	https://<ASPSP_FQDN>/platform/enabler/psd2orchestrator/ais/1.0.0/card-accounts
METHOD	GET

Parameter description

INPUT				
HEADER PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***
psu-authorization	This token is contained only, if an OAuth2 based authentication was performed in a pre-step or an OAuth2 based SCA was performed in an preceding AIS service in the same session. Reserved for future use.	O	-	String
authorization:Bearer	The value of the access token	M-Public	-	String
x-request-id	ID of the request, unique to the call, as determined by the initiating party.	M	255	String

consent-id	Shall be contained since "Establish Consent Transaction" was performed via this API before.	M	255	String
psu-ip-address	The forwarded IP Address header field consists of the corresponding http request IP Address field between PSU and TPP.	O	40	String
digest	Is contained if and only if the "Signature" element is contained in the header of the request.	O	255	String
signature	A signature of the request by the TPP on application level. This might be mandated by ASPSP. This string contains the following fields separated by commas: - *keyId*: The 'keyId' field is an opaque string that the server can use to look up the component they need to validate the signature. It could be an SSH key fingerprint, a URL to machine-readable key data, an LDAP DN, etc. Management of keys and assignment of 'keyId' is out of scope for this document. Serial Number of the TPP's certificate included in the "Certificate" header of this request. Serial Number of the TPP's certificate included in the "TPP-Signature-Certificate" header of this request. It shall be formatted as follows: keyId="SN=XXX,CA=YYYYYYYYYYYYYYYY" where "XXX" is the serial number of the certificate in hexadecimal coding given in the TPP-Signature-Certificate-Header and "YYYYYYYYYYYYYYYY" is the full Distinguished Name of the Certification Authority having produced this certificate. - algorithm: The 'algorithm' parameter is used to specify the digital signature algorithm to use when generating the signature. The algorithm must identify the same algorithm for the signature as presented in the certificate (Element "TPP-Certificate") of this Request. The available values are: "rsa-sha256" or "rsa-sha512" - Headers: The 'headers' parameter is used to specify the list of HTTP headers included when generating the signature for the message. If specified, it should be a lowercased, quoted list of HTTP header fields, separated by a single space character. If not specified, implementations MUST operate as if the field were specified with a single value, the 'Date' header, in the list of HTTP headers. Note that the list order is important, and MUST be specified in the order the HTTP	O	1024	String

	header field-value pairs are concatenated together during signing. Must include - “digest”, - “x-request-id”, - “psu-id” (if and only if “PSU-ID” is included as a header of the HTTP-Request). - “psu-corporate-id” (if and only if “psu-corporate-id” is included as a header of the HTTP-Request). - “Date” - "tpp-redirect-uri"(if and only if "tpp-redirect-uri" is included as a header of the HTTP-Request). No other entries may be included. - Signature: The ‘signature’ parameter is a base 64 encoded digital signature, as described in RFC 4648 [RFC4648], Section 4. The client uses the ‘algorithm’ and ‘headers’ signature parameters to form a canonicalised ‘signing string’. This ‘signing string’ is then signed with the key associated with ‘keyId’ and the algorithm corresponding to ‘algorithm’. The ‘signature’ parameter is then set to the base 64 encoding of the signature.			
tpp-signature-certificate	This is a X509 certificate that the TPP uses for signing the request, in base64 encoding. This certificate is in PEM format without the "-----BEGIN CERTIFICATE-----" and "-----END CERTIFICATE-----". Must be contained if a signature is contained, see above.	O	4096	String
aspsp-code	The ASPSP code	M	20	String
date	The date provided by the TPP. Format: EEE, dd MMM yyyy hh:mm:ss z	M	31	String
QUERY PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***
limit	Max elements per page to be returned. Only positive integers allowed.	O	-	String
offset	Requested page number. Only positive integers allowed. **Mandatory if limit param is provided	O	-	String

OUTPUT			
HEADER PARAM			
Parameter	***Description***	***Mandatory / Optional***	***Type***
x-request-id	ID of the request, unique to the call, as determined by the initiating party.	M	String
cpaas-total-elements	If query param offset=1, this field contains the total elements of the query executed on backend, before the pagination	O	String
cpaas-total-pages	If query param offset=1, this field contains the number of pages provided by the query executed on backend, before the pagination	O	String
BODY			
Parameter	***Description***	***Mandatory / Optional***	***Type***
errorManagement	Object identifying the error	O	Object
○ errorCode	Code that identifies error occurred	O	String
○ errorDescription	Error description	O	String
cardAccounts	Array of card account detail objects. At least a parameter is required.	M	List<Object>
○ resourceId	This is the data element to be used in the path when retrieving data from a dedicated account. This shall be filled, if addressable resource are created by the ASPSP.	O	String
○ maskedPan	Primary Account Number (PAN) of a card in a masked form (some digits are masked by a star). This data element can be used in the body of the Consent Request Message for retrieving account access consent from this card.	O	String
○ currency	The currency code. Codes following ISO 4217 Alpha 3	M	String
○ name	Name of the account given by the bank or the PSU in Online-Banking	O	String
○ product	Product Name of the Bank for this account, proprietary definition	O	String
○ status	Account status. The value is one of the following: - "enabled": account is available - "deleted": account is terminated - "blocked": account is blocked e.g. for legal reasons If this field is not used, than the account is available in the sense of this specification.	O	String
○ usage	Specifies the usage of the account - PRIV: private personal account - ORGA: professional account	O	String
○ details	Specifications that might be provided by the ASPSP - characteristics of the account - characteristics of the relevant card	O	String
○ creditLimit	Defines the credit limit of the PSU for all cards related to this card account in total.	O	Amount
▪ currency	The currency code. Codes following ISO 4217 Alpha 3	M	String
▪ amount	The amount given with fractional digits, where fractions must be compliant to the currency definition. Up to 14 significant figures. Negative amounts are signed by minus. The decimal separator is a dot. Example: Valid representations for EUR with up to two decimals are: • 1056 • 5768.2 • -1.50 • 5877.78	M	String

○ balances	List of account balances	O	List<Object>
▪ balanceAmount	Balance amount details	M	Amount
• currency	The currency code. Codes following ISO 4217 Alpha 3	M	String
• amount	The amount given with fractional digits, where fractions must be compliant to the currency definition. Up to 14 significant figures. Negative amounts are signed by minus. The decimal separator is a dot. Example: Valid representations for EUR with up to two decimals are: • 1056 • 5768.2 • -1.50 • 5877.78	M	String
▪ balanceType	Type of the balance. See possible type parameters in Appendix - BalanceType	M	String
▪ lastChangeDateTime	This data element might be used to indicate e.g. with the expected or booked balance that no action is known on the account, which is not yet booked. Format: ISO 8601 YYYY-MM-DDTHH:mm:ss.sssZ	O	String
▪ referenceDate	Reference date of the balance. Format: YYYY-MM-DD	O	String
▪ lastCommittedTransaction	EntryReference of the last commited transaction to support the TPP in identifying whether all PSU transactions are already known.	O	String
○ _links	Links to the account, which can be directly used for retrieving account information from this dedicated account. Links to "balances" and/or "transactions"	O	Links
○ account	A link to the resource providing the details of one account	O	Object
▪ href	This field contains a link to a resource.	M	String
○ balances	A link to the resource providing the balance of a dedicated account.	O	Object
▪ href	This field contains a link to a resource.	M	String
○ transactions	A link to the resource providing the transaction history of a dedicated account.	O	Object
▪ href	This field contains a link to a resource.	M	String
tppMessages	List of messages to the TPP on operational issues.	O	List<Message>
○ category	Only "ERROR" or "WARNING" permitted	M	String
○ code	The code of the error. Refers to the list of possible error code (Message code)	M	String
○ path	The path of the element of the request message which provoked this error message	O	String
○ text	Additional explaining text (max 512 characters)	O	String

HTTP Code	Result Description
200	Service executed successfully

Error management		
HTTP Code	**Error code**	**Error Description**
400	PSD2_01.000.A0002	Missing header parameter: {field name}
400	PSD2_01.000.A0003	Invalid header parameter: {field name}
400	PSD2_01.190.A0030	Entity not found
400	PSD2_01.190.A0018	Inconsistent consent resource status
400	PSD2_01.000.A0017	Repetition of query param not admitted: {param}
400	PSD2_00.189.A0005*	ASPSP provided error. Reference to the TPP-messages.
401	PSD2_01.000.A0009	Invalid signature
401	PSD2_00.189.A0005*	ASPSP provided error. Reference to the TPP-messages.
401	PSD2_01.190.A0032	Invalid consent resource
401	PSD2_01.190.A0034	The consent was created by this TPP but has expired and needs to be renewed.
401	PSD2_01.190.A0035	The consent has been invalidated by the ASPSP
403	PSD2_01.001.A0004	Unknown ASPSP
403	PSD2_01.001.A0020	Unknown TPP
403	PSD2_00.189.A0005*	ASPSP provided error. Reference to the TPP-messages.
404	PSD2_00.189.A0005*	ASPSP provided error. Reference to the TPP-messages.
406	PSD2_00.189.A0005*	ASPSP provided error. Reference to the TPP-messages.
409	PSD2_01.000.A0001	Operation not allowed
429	PSD2_01.190.A0033	The access on the account has been exceeding the consented multiplicity per day.
500	PSD2_00.000.A0000	Generic Error

* In this case the error is provided by the ASPSP. The http code and the TPP-Messages are defined by using the BG specification. Refers to Message Code section for details.

Example of readCardAccountList

GET https://<ASPSP_FQDN>/platform/enabler/psd2orchestrator/ais/1.0.0/card-accounts

Request:

```
HEADERS:
aspsp-code=12345
Content-Type: application/json
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
date = Wed, 27 Jun 2018 13:55:51 GMT
```

```
BODY:
N/A
```

Response:

```
HTTP Status code: 200

HEADERS:
```

```
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
```

```
BODY:
{
      “cardAccounts”: [{
            “resourceId”: “3dc3d5b3-7023-4848-9853-f5400a64e80f”,
            “maskedPan”: “4242*****4242”,
            “currency”: “EUR”,
            “product”: “Girokonto”,
            “name”: “Main Account”,
            “_links”: {
                  “balances”: {
                        “href”: “/accounts/3dc3d5b3-7023-4848-9853-
f5400a64e80f/balances”
                  },
                  “transactions”: {
                        “href”: “/accounts/3dc3d5b3-7023-4848-9853-
f5400a64e80f/transactions”
                  }
            }
      },
      {
            “resourceId”: “3dc3d5b3-7023-4848-9853-f5400a64e81g”,
            “maskedPan”: “5252****5252”,
            “currency”: “USD”,
            “product”: “Fremdwährungskonto”,
            “name”: “US Dollar Account”,
            “_links”: {
                  “balances”: {
                        “href”: “/accounts/3dc3d5b3-7023-4848-9853-
f5400a64e81g/balances”
                  }
            }
      }]
}
```

5.2 readCardAccountDetails

Reads details about a card account.

Description:

This API allows a TPP to get details of a specific card account reachable by PSD2 XS2A interfaces. Through this API it is also possible to know the balance of the account in case the consent used to invoke the API allows it and the ASPSP supports this feature.

The consent needed to use this API can be both a one-off consent or a recurring one. In case of a recurring consent, the usage of the consent to access this API is granted only if the consent is never already used. Under the above-mentioned conditions the usage can be both in attended or unattended mode.

In order to access to a card-account details, the consent can be asked by the TPP putting the account identifier in at least one of the following arrays of "access" parameter in the establishAccountInformationConsent API:

- accounts
- balances
- transactions

The TPP must provide the certificate, issued by the competent National Authority and qualifying the TPP as a AISP, in order to access to this API.

Among the mandatory parameters the TPP must provide to use this API, the main ones are:

- *account-id:* to identify the selected account to know the details. This identifier can be retrieved through the retrieveAccountList API;
- *consent-id:* to identify the Account Information Consent the TPP wants to use to access this service.

Tags: read, bank, card, account, details, data, balance, get

PROTOCOL	HTTP
PATH (Public Exposure)	https://<ASPSP_FQDN>/platform/enabler/psd2orchestrator/ais/1.0.0/card-accounts/{account-id}
METHOD	GET

Parameter description

INPUT				
HEADER PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***
psu-authorization	This token is contained only, if an OAuth2 based authentication was performed in a pre-	O	-	String

	step or an OAuth2 based SCA was performed in an preceding AIS service in the same session. Reserved for future use.			
authorization:Bearer	The value of the access token	M-Public	-	String
x-request-id	ID of the request, unique to the call, as determined by the initiating party.	M	255	String
consent-id	Shall be contained since “Establish Consent Transaction” was performed via this API before.	M	255	String
psu-ip-address	The forwarded IP Address header field consists of the corresponding http request IP Address field between PSU and TPP.	O	40	String
digest	Is contained if and only if the “Signature” element is contained in the header of the request.	O	255	String
signature	A signature of the request by the TPP on application level. This might be mandated by ASPSP. This string contains the following fields separated by commas: - *keyId*: The ‘keyId‘ field is an opaque string that the server can use to look up the component they need to validate the signature. It could be an SSH key fingerprint, a URL to machine-readable key data, an LDAP DN, etc. Management of keys and assignment of ‘keyId‘ is out of scope for this document. Serial Number of the TPP's certificate included in the “Certificate” header of this request. Serial Number of the TPP's certificate included in the “TPP-Signature-Certificate” header of this request. It shall be formatted as follows: keyId=“SN=XXX,CA=YYYYYYYYYYYYYYYY” where “XXX” is the serial number of the certificate in hexadecimal coding given in the TPP-Signature-Certificate-Header and “YYYYYYYYYYYYYYYY” is the full Distinguished Name of the Certification Authority having produced this certificate. - algorithm: The ‘algorithm’ parameter is used to specify the digital signature algorithm to use when generating the signature. The algorithm must identify the same algorithm for the signature as presented in the certificate (Element “TPP-Certificate”) of this Request. The available values are: “rsa-sha256” or “rsa-sha512” - Headers: The ‘headers’ parameter is used to specify the list of HTTP headers included when generating the signature for the message. If specified, it should be a lowercased, quoted list of HTTP header fields, separated by a single space character. If not	O	1024	String

	specified, implementations MUST operate as if the field were specified with a single value, the 'Date' header, in the list of HTTP headers. Note that the list order is important, and MUST be specified in the order the HTTP header field-value pairs are concatenated together during signing. Must include - "digest", - "x-request-id", - "psu-id" (if and only if "PSU-ID" is included as a header of the HTTP-Request). - "psu-corporate-id" (if and only if "psu-corporate-id" is included as a header of the HTTP-Request). - "Date" - "tpp-redirect-uri"(if and only if "tpp-redirect-uri" is included as a header of the HTTP-Request). No other entries may be included. - Signature: The 'signature' parameter is a base 64 encoded digital signature, as described in RFC 4648 [RFC4648], Section 4. The client uses the 'algorithm' and 'headers' signature parameters to form a canonicalised 'signing string'. This 'signing string' is then signed with the key associated with 'keyId' and the algorithm corresponding to 'algorithm'. The 'signature' parameter is then set to the base 64 encoding of the signature.			
tpp-signature-certificate	This is a X509 certificate that the TPP uses for signing the request, in base64 encoding. This certificate is in PEM format without the "-----BEGIN CERTIFICATE-----" and "-----END CERTIFICATE-----". Must be contained if a signature is contained, see above.	O	4096	String
aspsp-code	The ASPSP code	M	20	String
date	The date provided by the TPP. Format: EEE, dd MMM yyyy hh:mm:ss z	M	31	String
PATH PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***
account-id	This identification is denoting the addressed account. The account-id is the UUID related to the account structure. Its value is constant at least throughout the lifecycle of a given consent.	M	100	String

OUTPUT			
HEADER PARAM			
Parameter	***Description***	***Mandatory / Optional***	***Type***
x-request-id	ID of the request, unique to the call, as determined by the initiating party.	M	String

BODY			
Parameter	***Description***	***Mandatory / Optional***	***Type***
errorManagement	Object identifying the error	O	Object
○ errorCode	Code that identifies error occurred	O	String
○ errorDescription	Error description	O	String
cardAccounts	Card account detail objects.	M	CardAccount
○ resourceId	This is the data element to be used in the path when retrieving data from a dedicated account. This shall be filled, if addressable resource are created by the ASPSP.	O	String
○ maskedPan	Primary Account Number (PAN) of a card in a masked form (some digits are masked by a star). This data element can be used in the body of the Consent Request Message for retrieving account access consent from this card.	O	String
○ currency	The currency code. Codes following ISO 4217 Alpha 3	M	String
○ name	Name of the account given by the bank or the PSU in Online-Banking	O	String
○ product	Product Name of the Bank for this account, proprietary definition	O	String
○ status	Account status. The value is one of the following: - "enabled": account is available - "deleted": account is terminated - "blocked": account is blocked e.g. for legal reasons If this field is not used, than the account is available in the sense of this specification.	O	String
○ usage	Specifies the usage of the account - PRIV: private personal account - ORGA: professional account	O	String
○ details	Specifications that might be provided by the ASPSP - characteristics of the account - characteristics of the relevant card	O	String
○ creditLimit	Defines the credit limit of the PSU for all cards related to this card account in total.	O	Amount
▪ currency	The currency code. Codes following ISO 4217 Alpha 3	M	String
▪ amount	The amount given with fractional digits, where fractions must be compliant to the currency definition. Up to 14 significant figures. Negative amounts are signed by minus. The decimal separator is a dot. Example: Valid representations for EUR with up to two decimals are: • 1056 • 5768.2 • -1.50 • 5877.78	M	String
○ balances	List of account balances	O	List<Object>
▪ balanceAmount	Balance amount details	M	Amount
• currency	The currency code. Codes following ISO 4217 Alpha 3	M	String

• amount	The amount given with fractional digits, where fractions must be compliant to the currency definition. Up to 14 significant figures. Negative amounts are signed by minus. The decimal separator is a dot. Example: Valid representations for EUR with up to two decimals are: • 1056 • 5768.2 • -1.50 • 5877.78	M	String
▪ balanceType	Type of the balance. See possible type parameters in Appendix - BalanceType	M	String
▪ lastChangeDateTime	This data element might be used to indicate e.g. with the expected or booked balance that no action is known on the account, which is not yet booked. Format: ISO 8601 YYYY-MM-DDTHH:mm:ss.sssZ	O	String
▪ referenceDate	Reference date of the balance. Format: YYYY-MM-DD	O	String
▪ lastCommittedTransaction	EntryReference of the last commited transaction to support the TPP in identifying whether all PSU transactions are already known.	O	String
○ _links	Links to “balances” and/or “transactions”	O	Links
○ balances	A link to the resource providing the balance of a dedicated account.	O	Object
▪ href	This field contains a link to a resource.	M	String
○ transactions	A link to the resource providing the transaction history of a dedicated account.	O	Object
▪ href	This field contains a link to a resource.	M	String
tppMessages	List of messages to the TPP on operational issues.	O	List<Message>
○ category	Only "ERROR" or "WARNING" permitted	M	String
○ code	The code of the error. Refers to the list of possible error code (Message code)	M	String
○ path	The path of the element of the request message which provoked this error message	O	String
○ text	Additional explaining text (max 512 characters)	O	String

HTTP Code	Result Description
200	Service executed successfully

Error management		
HTTP Code	Error code	Error Description
400	PSD2_01.000.A0002	Missing header parameter: {field name}
400	PSD2_01.000.A0003	Invalid header parameter: {field name}
400	PSD2_01.190.A0030	Entity not found
400	PSD2_01.190.A0018	Inconsistent consent resource status
400	PSD2_01.000.A0017	Repetition of query param not admitted: {param}
400	PSD2_00.189.A0005*	ASPSP provided error. Reference to the TPP-messages.
401	PSD2_01.000.A0009	Invalid signature
401	PSD2_00.189.A0005*	ASPSP provided error. Reference to the TPP-messages.
401	PSD2_01.190.A0032	Invalid consent resource

Error management		
HTTP Code	**Error code**	**Error Description**
401	PSD2_01.190.A0034	The consent was created by this TPP but has expired and needs to be renewed.
401	PSD2_01.190.A0035	The consent has been invalidated by the ASPSP
403	PSD2_01.001.A0004	Unknown ASPSP
403	PSD2_01.001.A0020	Unknown TPP
404	PSD2_00.189.A0005*	ASPSP provided error. Reference to the TPP-messages.
406	PSD2_00.189.A0005*	ASPSP provided error. Reference to the TPP-messages.
409	PSD2_01.000.A0001	Operation not allowed
429	PSD2_01.190.A0033	The access on the account has been exceeding the consented multiplicity per day.
500	PSD2_00.000.A0000	Generic Error

* In this case the error is provided by the ASPSP. The http code and the TPP-Messages are defined by using the BG specification. Refers to Message Code section for details.

Example of readCardAccountDetails

GET https://<IAM_DNS>/platform/enabler/psd2orchestrator/ais/1.0.0/card-accounts/3dc3d5b3-7023-4848-9853-f5400a64e80f

Request:

```
HEADERS:
aspsp-code=12345
Consent-ID = tbd
Content-Type: application/json
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
date = Wed, 27 Jun 2018 13:55:51 GMT
```

```
BODY:
N/A
```

Response:

```
HTTP Status code: 200

HEADERS:
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
```

```
BODY:
{
      “cardAccount”: {
            “resourceId”: “3dc3d5b3-7023-4848-9853-f5400a64e80f”,
            “maskedPan”: “525412******3241”,
            “name”: “Main”,
            “currency”: “EUR”,
            “product”: “Multicurrency Account”,
            “status”: “enabled”,
            “creditLimit”: { “currency”: “EUR”, “amount”: 15000 },
            “_links”: {
                  “balances”: {
                        “href”: “/accounts/3dc3d5b3-7023-4848-9853-
f5400a64e80f/balances”
                  },
                  “transactions”: {
                        “href”: “/accounts/3dc3d5b3-7023-4848-9853-
f5400a64e80f/transactions”
                  }
            }
      }
}
```

5.3 readCardAccountBalance

Reads balance data from a given card account addressed by "*account-id*".

Description:

This API allows a TPP to get balances of a specific card account reachable by PSD2 XS2A interfaces. The possible balance types the API can give in response are the following:

- closingBooked
- expected
- authorised
- openingBooked
- interimAvailable
- forwardAvailable

Each ASPSP has to specify which of these balance types are supported.

The consent needed to use this API can be both a one-off consent or a recurring one. The access to this API is allowed both in attended or unattended mode. In case of unattended usage, the maximum daily allowed usage is 4 times.

In order to access to the card-account balances, the consent can be asked by the TPP putting the account identifier in the “balances” array of "access" parameter in the establishAccountInformationConsent API.

The TPP must provide the certificate, issued by the competent National Authority and qualifying the TPP as a AISP, in order to access to this API.

Among the mandatory parameters the TPP must provide to use this API, the main ones are:

- *account-id:* to identify the selected account to know the details. This identifier can be retrieved through the retrieveAccountList API
- *consent-id:* to identify the Account Information Consent the TPP wants to use to access this service

Tags: read, card, account, data, balance, get

PROTOCOL	HTTP
PATH (Public Exposure)	https://<ASPSP_FQDN>/platform/enabler/psd2orchestrator/ais/1.0.0/card-accounts/{account-id}/balances
METHOD	GET

Parameter description

INPUT				
HEADER PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***

psu-authorization	This token is contained only, if an OAuth2 based authentication was performed in a pre-step or an OAuth2 based SCA was performed in an preceding AIS service in the same session. Reserved for future use.	O	-	String
authorization:Bearer	The value of the access token	M-Public	-	String
x-request-id	ID of the request, unique to the call, as determined by the initiating party.	M	255	String
consent-id	Shall be contained since “Establish Consent Transaction” was performed via this API before.	M	255	String
psu-ip-address	The forwarded IP Address header field consists of the corresponding http request IP Address field between PSU and TPP.	O	40	String
digest	Is contained if and only if the “Signature” element is contained in the header of the request.	O	255	String
signature	A signature of the request by the TPP on application level. This might be mandated by ASPSP. This string contains the following fields separated by commas: - *keyId*: The ‘keyId‘ field is an opaque string that the server can use to look up the component they need to validate the signature. It could be an SSH key fingerprint, a URL to machine-readable key data, an LDAP DN, etc. Management of keys and assignment of ‘keyId‘ is out of scope for this document. Serial Number of the TPP's certificate included in the “Certificate” header of this request. Serial Number of the TPP's certificate included in the “TPP-Signature-Certificate” header of this request. It shall be formatted as follows: keyId=“SN=XXX,CA=YYYYYYYYYYYYYYYY” where “XXX” is the serial number of the certificate in hexadecimal coding given in the TPP-Signature-Certificate-Header and “YYYYYYYYYYYYYYYY” is the full Distinguished Name of the Certification Authority having produced this certificate. - algorithm: The ‘algorithm’ parameter is used to specify the digital signature algorithm to use when generating the signature. The algorithm must identify the same algorithm for the signature as presented in the certificate (Element “TPP-Certificate”) of this Request. The available values are: “rsa-sha256” or “rsa-sha512” - Headers: The ‘headers’ parameter is used to specify the list of HTTP headers included when generating the signature for the message. If specified, it should be a	O	1024	String

	lowercased, quoted list of HTTP header fields, separated by a single space character. If not specified, implementations MUST operate as if the field were specified with a single value, the 'Date' header, in the list of HTTP headers. Note that the list order is important, and MUST be specified in the order the HTTP header field-value pairs are concatenated together during signing. Must include - "digest", - "x-request-id", - "psu-id" (if and only if "PSU-ID" is included as a header of the HTTP-Request). - "psu-corporate-id" (if and only if "psu-corporate-id" is included as a header of the HTTP-Request). - "Date" - "tpp-redirect-uri"(if and only if "tpp-redirect-uri" is included as a header of the HTTP-Request). No other entries may be included. - Signature: The 'signature' parameter is a base 64 encoded digital signature, as described in RFC 4648 [RFC4648], Section 4. The client uses the 'algorithm' and 'headers' signature parameters to form a canonicalised 'signing string'. This 'signing string' is then signed with the key associated with 'keyId' and the algorithm corresponding to 'algorithm'. The 'signature' parameter is then set to the base 64 encoding of the signature.			
tpp-signature-certificate	This is a X509 certificate that the TPP uses for signing the request, in base64 encoding. This certificate is in PEM format without the "-----BEGIN CERTIFICATE-----" and "-----END CERTIFICATE-----". Must be contained if a signature is contained, see above.	O	4096	String
aspsp-code	The ASPSP code	M	20	String
date	The date provided by the TPP. Format: EEE, dd MMM yyyy hh:mm:ss z	M	31	String
PATH PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***
account-id	This identification is denoting the addressed card account. The account-id is retrieved by using a "Read Account List" call. The account-id is the "resourceId" attribute of the account structure. Its value is constant at least throughout the lifecycle of a given consent.	M	100	String

OUTPUT			
HEADER PARAM			
Parameter	***Description***	***Mandatory / Optional***	***Type***
x-request-id	ID of the request, unique to the call, as determined by the initiating party.	M	String
BODY			
Parameter	***Description***	***Mandatory / Optional***	***Type***
errorManagement	Object identifying the error	O	Object
○ errorCode	Code that identifies error occurred	O	String
○ errorDescription	Error description	O	String
account	Identifier of the addressed card account. *Exactly one of these parameters is required.	O	Object
○ iban	This is an identifier used internationally by financial institutions to uniquely identify the account of a customer at a financial institution (IBANIdentifier ISO 20022). According to ISO 13616: Pattern: [A-Z]{2}[0-9]{2}[A-Z0-9]{1,30}	O	String
○ bban	This data elements is used for payment accounts which have no IBAN.	O*	String
○ pan	Primary Account Number (PAN) of a card, can be tokenised by the ASPSP due to PCI DSS requirements.	O*	String
○ maskedPan	Primary Account Number (PAN) of a card in a masked form (some digits are masked by a star).	O*	String
○ msisdn	An alias to access a payment account via a registered mobile phone number.	O*	String
○ currency	The currency code. Codes following ISO 4217 Alpha 3	O	String
balances	A list of balances regarding this card account, e.g. the current balance, the last booked balance.	M	List<Balance>
○ balanceAmount	Balance amount details	M	Amount
• currency	The currency code. Codes following ISO 4217 Alpha 3	M	String
• amount	The amount given with fractional digits, where fractions must be compliant to the currency definition. Up to 14 significant figures. Negative amounts are signed by minus. The decimal separator is a dot. Example: Valid representations for EUR with up to two decimals are: • 1056 • 5768.2 • -1.50 • 5877.78	M	String
○ balanceType	Type of the balance. Accepted values: - closingBooked - expected - authorised - openingBooked - interimAvailable - forwardAvailable See possible type parameters in Appendix - BalanceType	M	String
○ lastChangeDateTime	This data element might be used to indicate e.g. with the expected or booked balance that no action is known on the account, which is not yet booked. Format: ISO 8601 YYYY-MM-DDTHH:mm:ss.sssZ	O	String

○ referenceDate	Reference date of the balance. Format: YYYY-MM-DD	O	String
○ lastCommittedTransaction	EntryReference of the last commited transaction to support the TPP in identifying whether all PSU transactions are already known.	O	String
tppMessages	List of messages to the TPP on operational issues.	O	List<Message>
○ category	Only "ERROR" or "WARNING" permitted	M	String
○ code	The code of the error. Refers to the list of possible error code (Message code)	M	String
○ path	The path of the element of the request message which provoked this error message	O	String
○ text	Additional explaining text (max 512 characters)	O	String

HTTP Code	Result Description
200	Service executed successfully

Error management		
HTTP Code	Error code	Error Description
400	PSD2_01.000.A0002	Missing header parameter: {field name}
400	PSD2_01.000.A0003	Invalid header parameter: {field name}
400	PSD2_01.000.A0017	Repetition of query param not admitted: {param}
400	PSD2_00.189.A0005*	ASPSP provided error. Reference to the TPP-messages.
400	PSD2_01.190.A0030	Entity not found
400	PSD2_01.190.A0018	Inconsistent consent resource status
401	PSD2_01.000.A0009	Invalid signature
401	PSD2_00.189.A0005*	ASPSP provided error. Reference to the TPP-messages.
401	PSD2_01.190.A0032	Invalid consent resource
401	PSD2_01.190.A0034	The consent was created by this TPP but has expired and needs to be renewed.
401	PSD2_01.190.A0035	The consent has been invalidated by the ASPSP
403	PSD2_01.001.A0004	Unknown ASPSP
403	PSD2_01.001.A0020	Unknown TPP
403	PSD2_00.189.A0005*	ASPSP provided error. Reference to the TPP-messages.
404	PSD2_00.189.A0005*	ASPSP provided error. Reference to the TPP-messages.
406	PSD2_00.189.A0005*	ASPSP provided error. Reference to the TPP-messages.
409	PSD2_01.000.A0001	Operation not allowed
429	PSD2_01.190.A0033	The access on the account has been exceeding the consented multiplicity per day.
500	PSD2_00.000.A0000	Generic Error

* In this case the error is provided by the ASPSP. The http code and the TPP-Messages are defined by using the BG specification. Refers to Message Code section for details.

Example of readCardAccountBalance

GET https://<ASPSP_FQDN>/platform/enabler/psd2orchestrator/ais/1.0.0/card-accounts/3dc3d5b3-7023-4848-9853-f5400a64e80f/balances

Request:

```
HEADERS :
```

```
aspsp-code=12345
Consent-ID = tbd
Content-Type: application/json
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
date = Wed, 27 Jun 2018 13:55:51 GMT
```

```
BODY:
N/A
```

Response:

```
HTTP Status code: 200

HEADERS:
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
```

```
BODY:
{
        "cardAccount": {
                "maskedPan": "424242******4343"
        },
        "balances": [{
                "balanceType": "closingBooked",
                "balanceAmount": {
                        "currency": "EUR",
                        "amount": "500.00"
                },
                "referenceDate": "2017-10-25"
        },
        {
                "balanceType": "expected",
                "balanceAmount": {
                        "currency": "EUR",
                        "amount": "900.00"
                },
                "lastChangeDateTime": "2017-10-25T15:30:35.035Z"
        },
        {
                "balanceType": "closingBooked",
                "balanceAmount": {
                        "currency": "USD",
                        "amount": "350.00"
                },
                "referenceDate": "2017-10-25"
        },
        {
                "balanceType": "expected",
                "balanceAmount": {
                        "currency": "USD",
                        "amount": "350.00"
                },
                "lastChangeDateTime": "2017-10-24T14:30:21Z"
        }]
}
```

5.4 readCardAccountTransactionList

Reads account data from a given card account addressed by *account-id*.

Description:

This API allows a TPP to get transactions list of a specific bank account reachable by PSD2 XS2A interfaces. In general, it is possible to ask for transactions according to their booking status, but each ASPSP can specify if this feature is supported or not.

The consent needed to use this API can be both a one-off consent or a recurring one. The access to this API is allowed both in attended or unattended mode. In case of unattended usage, the maximum daily allowed usage is 4 times.

In order to access to the account transactions list, the consent can be asked by the TPP putting the account identifier in the “transactions” array of "access" parameter in the establishAccountInformationConsent API.

The check on consent validity to access to this API is managed according to input parameter used by the TPP:

- in case of usage of date_from and date_to query parameters, the validity check of the consent is up to the PSD2 Gateway. The validity check is based on following rules:
  - when the *date_from* is within 90 days in the past, the consent can be both one-off and recurring and in this latter case can be already used.
  - When the date_from is over 90 days in the past, the consent must be a one-off consent or a recurring one never used.
- in case of usage of *delta_list* query parameter, the validity check of the consent is up to the ASPSP.

In case of the number of transactions retrieved requires pagination, each access to the pages increases the consent usage counter.

The TPP must provide the certificate, issued by the competent National Authority and qualifying the TPP as a AISP, in order to access to this API.

Among the mandatory parameters the TPP must provide to use this API, the main ones are:

- *account-id:* to identify the selected account to know the details. This identifier can be retrieved through the retrieveAccountList API;
- *consent-id:* to identify the Account Information Consent the TPP wants to use to access this service
- *date_from, date_to:* to define the search time interval;
- *delta_list:* to ask transactions after the last.

Tags: read, bank, transaction, list, data, get

PROTOCOL	HTTP
PATH (Public Exposure)	https://<ASPSP_FQDN>/platform/enabler/psd2orchestrator/ais/1.0.0/card-accounts/{account-id}/transactions
METHOD	GET

Parameter description

INPUT				
HEADER PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***
psu-authorization	This token is contained only, if an OAuth2 based authentication was performed in a pre-step or an OAuth2 based SCA was performed in an preceding AIS service in the same session. Reserved for future use.	O	-	String
authorization:Bearer	The value of the access token	M-Public	-	String
x-request-id	ID of the request, unique to the call, as determined by the initiating party.	M	255	String
consent-id	Shall be contained since “Establish Consent Transaction” was performed via this API before.	M	255	String
psu-ip-address	The forwarded IP Address header field consists of the corresponding http request IP Address field between PSU and TPP.	O	40	String
digest	Is contained if and only if the “Signature” element is contained in the header of the request.	O	255	String
signature	A signature of the request by the TPP on application level. This might be mandated by ASPSP. This string contains the following fields separated by commas: - *keyId*: The ‘keyId‘ field is an opaque string that the server can use to look up the component they need to validate the signature. It could be an SSH key fingerprint, a URL to machine-readable key data, an LDAP DN, etc. Management of keys and assignment of ‘keyId‘ is out of scope for this document. Serial Number of the TPP's certificate included in the “Certificate” header of this request. Serial Number of the TPP's certificate included in the “TPP-Signature-Certificate” header of this request. It shall be formatted as follows: keyId=“SN=XXX,CA=YYYYYYYYYYYYYYYY” where “XXX” is the serial number of the certificate in hexadecimal coding given in the TPP-Signature-Certificate-Header and “YYYYYYYYYYYYYYYY” is the full Distinguished Name of the Certification Authority having produced this certificate. - algorithm: The ‘algorithm’ parameter is used to specify the digital signature algorithm to use when generating the signature.	O	1024	String

	The algorithm must identify the same algorithm for the signature as presented in the certificate (Element “TPP-Certificate”) of this Request. The available values are: “rsa-sha256” or “rsa-sha512” - Headers: The ‘headers’ parameter is used to specify the list of HTTP headers included when generating the signature for the message. If specified, it should be a lowercased, quoted list of HTTP header fields, separated by a single space character. If not specified, implementations MUST operate as if the field were specified with a single value, the ‘Date’ header, in the list of HTTP headers. Note that the list order is important, and MUST be specified in the order the HTTP header field-value pairs are concatenated together during signing. Must include - “digest”, - “x-request-id”, - “psu-id” (if and only if “PSU-ID” is included as a header of the HTTP-Request). - “psu-corporate-id” (if and only if “psu-corporate-id” is included as a header of the HTTP-Request). - “Date” - "tpp-redirect-uri"(if and only if "tpp-redirect-uri" is included as a header of the HTTP-Request). No other entries may be included. - Signature: The ‘signature’ parameter is a base 64 encoded digital signature, as described in RFC 4648 [RFC4648], Section 4. The client uses the ‘algorithm’ and ‘headers’ signature parameters to form a canonicalised ‘signing string’. This ‘signing string’ is then signed with the key associated with ‘keyId’ and the algorithm corresponding to ‘algorithm’. The ‘signature’ parameter is then set to the base 64 encoding of the signature.			
tpp-signature-certificate	This is a X509 certificate that the TPP uses for signing the request, in base64 encoding. This certificate is in PEM format without the "-----BEGIN CERTIFICATE-----" and "-----END CERTIFICATE-----". Must be contained if a signature is contained, see above.	O	4096	String
aspsp-code	The ASPSP code	M	20	String
date	The date provided by the TPP. Format: EEE, dd MMM yyyy hh:mm:ss z	M	31	String
PATH PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***

account-id	This identification is denoting the addressed card account. The account-id is retrieved by using a "Read Card Account List" call. The account-id is the UUID related to the account structure. Its value is constant at least throughout the lifecycle of a given consent.	M	100	String
QUERY PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***
date_from	Starting date (inclusive the date dateFrom) of the transaction list, mandated if no delta access is required. Format: YYYY-MM-DD	O	-	String
date_to	End date (inclusive the data dateTo) of the transaction list, default is now if not given. Format: YYYY-MM-DD	O	-	String
booking_status	Permitted codes are "booked", "pending" and "both". "booked" shall be supported by the ASPSP. To support the "pending" and "both" feature is optional for the ASPSP, Error code if not supported in the online banking frontend.	M	-	String
delta_list	Only “true” or “false” value is accepted. This data attribute is indicating that the AISP is in favour to get all transactions after the last report access for this PSU on the addressed account. This is another implementation of a delta access-report. This delta indicator might be rejected by the ASPSP if this function is not supported.	O*	5	String
limit	Max elements per page to be returned. Only positive integers allowed.	O	-	String
offset	Requested page number. Only positive integers allowed. **Mandatory if limit param is provided	O	-	String

*If is valorized only the parameter *delta_list* and not *date_to/date_from*, the validity consent must be checked by ASPSP

OUTPUT			
HEADER PARAM			
Parameter	***Description***	***Mandatory / Optional***	***Type***
x-request-id	ID of the request, unique to the call, as determined by the initiating party.	M	String
cpaas-total-elements	If query param offset=1, this field contains the total elements of the query executed on backend, before the pagination	O	String
cpaas-total-pages	If query param offset=1, this field contains the number of pages provided by the query executed on backend, before the pagination	O	String

BODY			
Parameter	***Description***	***Mandatory / Optional***	***Type***
errorManagement	Object identifying the error	O	Object
○ errorCode	Code that identifies error occurred	O	String
○ errorDescription	Error description	O	String
account	Identifier of the addressed card account.	O	Object

	*Exactly one of these parameters is required.		
○ iban	This is an identifier used internationally by financial institutions to uniquely identify the account of a customer at a financial institution (IBANIdentifier ISO 20022). According to ISO 13616: Pattern: [A-Z]{2}[0-9]{2}[A-Z0-9]{1,30}	O*	String
○ bban	This data elements is used for payment accounts which have no IBAN. Specifies the Basic Bank Account Number (BBANIdentifier ISO 20022), an Identifier used nationally by financial institutions, ie, in individual countries, generally as part of a National Account Numbering Scheme(s), which uniquely identifies the account of a customer. Pattern = "[a-zA-Z0-9]{1,30}"	O*	String
○ pan	Primary Account Number (PAN) of a card, can be tokenized by the ASPSP due to PCI DSS requirements. This data element can be used in the body of the Consent Request Message for retrieving account access consent from this card.	O*	String
○ maskedPan	Primary Account Number (PAN) of a card in a masked form (some digits are masked by a star). This data element can be used in the body of the Consent Request Message for retrieving account access consent from this card.	O*	String
○ msisdn	An alias to access a payment account via a registered mobile phone number. This alias might be needed e.g. in the payment initiation service. The support of this alias must be explicitly documented by the ASPSP for the corresponding API Calls.	O*	String
○ currency	The currency code. Codes following ISO 4217 Alpha 3	O	String
balances	A list of balances regarding this account, which might be restricted to the current balance.	O	List<Object>
○ balanceAmount	Balance amount details	M	Object
▪ currency	The currency code. Codes following ISO 4217 Alpha 3	M	String
▪ amount	The amount given with fractional digits, where fractions must be compliant to the currency definition. Up to 14 significant figures. Negative amounts are signed by minus. The decimal separator is a dot. Example: Valid representations for EUR with up to two decimals are: • 1056 • 5768.2 • -1.50 • 5877.78	M	String
○ balanceType	Type of the balance. Accepted values: - closingBooked - expected - authorised - openingBooked - interimAvailable - forwardAvailable	M	String

	See Appendix - BalanceType		
○ lastChangeDateTime	This data element might be used to indicate e.g. with the expected or booked balance that no action is known on the account, which is not yet booked. Format: ISO 8601 YYYY-MM-DDTHH:mm:ss.sssZ	O	String
○ referenceDate	Reference date of the balance. Format: YYYY-MM-DD	O	String
○ lastCommittedTransaction	EntryReference of the last commited transaction to support the TPP in identifying whether all PSU transactions are already known.	O	String
transactions	Is asking for transactions of the addressed card accounts. If the array is empty, the TPP is asking for the transactions of all accessible account lists. This may be restricted in a PSU/ASPSP authorization dialogue.	O	Object
○ booked	List of booked transactions	O	List<Object>
▪ cardTransactionId	Unique end to end identity.	O	String
▪ terminalId	Identification of the Terminal, where the card has been used.	O	String
▪ transactionDate	Date of the actual card transaction. Format: YYYY-MM-DD	O	String
▪ bookingDate	Booking date of the related booking on the card account. Format: YYYY-MM-DD	O	String
▪ transactionAmount	The amount of the transaction as billed to the card account.	M	Object
• currency	The currency code. Codes following ISO 4217 Alpha 3	M	String
• amount	The amount given with fractional digits, where fractions must be compliant to the currency definition. Up to 14 significant figures. Negative amounts are signed by minus. The decimal separator is a dot. Example: Valid representations for EUR with up to two decimals are: • 1056 • 5768.2 • -1.50 • 5877.78	M	String
▪ exchangeRate	List of exchange rates	O	List<Object>
• currencyFrom	The currency code. Codes following ISO 4217 Alpha 3	M	String
• rateFrom	The exchange rate expressed in the source currency.	M	String
• currencyTo	The currency code. Codes following ISO 4217 Alpha 3	M	String
• rateTo	The exchange rate expressed in the destination currency.	M	String
• rateDate	Conversion rate validity date. Fromat: YYYY-MM-DD	M	String
• rateContract	Agreed exchange rate.	O	String
▪ originalAmount	Original amount of the transaction at the Point of Interaction in orginal currency	O	Object
• currency	The currency code. Codes following ISO 4217 Alpha 3	M	String

• amount	The amount given with fractional digits, where fractions must be compliant to the currency definition. Up to 14 significant figures. Negative amounts are signed by minus. The decimal separator is a dot. Example: Valid representations for EUR with up to two decimals are: • 1056 • 5768.2 • -1.50 • 5877.78	M	String
▪ markupFee	Any fee related to the transaction in billing currency.	O	Object
• currency	The currency code. Codes following ISO 4217 Alpha 3	M	String
• amount	The amount given with fractional digits, where fractions must be compliant to the currency definition. Up to 14 significant figures. Negative amounts are signed by minus. The decimal separator is a dot. Example: Valid representations for EUR with up to two decimals are: • 1056 • 5768.2 • -1.50 • 5877.78	M	String
▪ markupFeePercentage	Percentage of the involved transaction fee in relation to the billing amount.	O	String
▪ cardAcceptorId	Identification of the Card Acceptor (e.g. merchant) as given in the related card transaction.	O	String
▪ cardAcceptorAddress		O	String
• street	The street name	O	String
• buildingNumber	The building number	O	String
• city	The city	O	String
• postalCode	The postal code	O	String
• country	The country	M	String
▪ cardAcceptorCategoryCode	Card Acceptor Category Code of the Card Acceptor as given in the related card transaction.	O	String
▪ maskedPan	The masked PAN of the card used in the transaction.	O	String
▪ transactionDetails	Additional details given for the related card transactions.	O	String
▪ invoiced	Flag indicating whether the underlying card transaction is already invoiced.	O	Boolean
▪ proprietaryBankTransactionCode	Proprietary bank transaction code as used within a community or within an ASPSP e.g. for MT94x based transaction reports. Max length 35.	O	String
○ pending	List of pending transactions	O	List<Object>
▪ cardTransactionId	Unique end to end identity.	O	String
▪ terminalId	Identification of the Terminal, where the card has been used.	O	String

▪ transactionDate	Date of the actual card transaction. Format: YYYY-MM-DD	O	String
▪ bookingDate	Booking date of the related booking on the card account. Format: YYYY-MM-DD	O	String
▪ transactionAmount	The amount of the transaction as billed to the card account.	M	Object
▪ currency	The currency code. Codes following ISO 4217 Alpha 3	M	String
• amount	The amount given with fractional digits, where fractions must be compliant to the currency definition. Up to 14 significant figures. Negative amounts are signed by minus. The decimal separator is a dot. Example: Valid representations for EUR with up to two decimals are: • 1056 • 5768.2 • -1.50 • 5877.78	M	String
▪ exchangeRate	List of exchange rates	O	List<Object>
• currencyFrom	The currency code. Codes following ISO 4217 Alpha 3	M	String
• rateFrom	The exchange rate expressed in the source currency.	M	String
• currencyTo	The currency code. Codes following ISO 4217 Alpha 3	M	String
• rateTo	The exchange rate expressed in the destination currency.	M	String
• rateDate	Conversion rate validity date. Fromat: YYYY-MM-DD	M	String
• rateContract	Agreed exchange rate.	O	String
▪ originalAmount	Original amount of the transaction at the Point of Interaction in orginal currency	O	Object
• currency	The currency code. Codes following ISO 4217 Alpha 3	M	String
• amount	The amount given with fractional digits, where fractions must be compliant to the currency definition. Up to 14 significant figures. Negative amounts are signed by minus. The decimal separator is a dot. Example: Valid representations for EUR with up to two decimals are: • 1056 • 5768.2 • -1.50 • 5877.78	M	String
▪ markupFee	Any fee related to the transaction in billing currency.	O	Object
• currency	The currency code. Codes following ISO 4217 Alpha 3	M	String
• amount	The amount given with fractional digits, where fractions must be compliant to the currency definition. Up to 14 significant figures. Negative amounts are signed by minus. The decimal separator is a dot. Example: Valid representations for EUR with up to two decimals are: • 1056 • 5768.2	M	String

	• -1.50 • 5877.78		
▪ markupFeePercentage	Percentage of the involved transaction fee in relation to the billing amount.	O	String
▪ cardAcceptorId	Identification of the Card Acceptor (e.g. merchant) as given in the related card transaction.	O	String
▪ cardAcceptorAddress	Address of the Card Acceptor as given in the related card transaction.	O	String
• street	The street name	O	String
• buildingNumber	The building number	O	String
• city	The city	O	String
• postalCode	The postal code	O	String
• country	The country	M	String
▪ cardAcceptorCategoryCode	Card Acceptor Category Code of the Card Acceptor as given in the related card transaction.	O	String
▪ maskedPan	The masked PAN of the card used in the transaction.	O	String
▪ transactionDetails	Additional details given for the related card transactions.	O	String
▪ invoiced	Flag indicating whether the underlying card transaction is already invoiced.	O	Boolean
▪ proprietaryBankTransactionCode	Proprietary bank transaction code as used within a community or within an ASPSP e.g. for MT94x based transaction reports. Max length 35.	O	String
○ _links	The following links could be used here: - account (mandatory);	O	Object
▪ account	Reference to the account	M	Object
• href	This field contains a link to a resource.	M	String
_links	The following links could be used here: **download**.	O	Object
○ download	A link to a resource, where the transaction report might be downloaded from in case where transaction reports have a huge size.	O	Object
▪ href	This field contains a link to a resource.	M	String
tppMessages	List of messages to the TPP on operational issues.	O	List<Message>
○ category	Only "ERROR" or "WARNING" permitted	M	String
○ code	The code of the error. Refers to the list of possible error code (Message code)	M	String
○ path	The path of the element of the request message which provoked this error message	O	String
○ text	Additional explaining text (max 512 characters)	O	String

HTTP Code	Result Description
200	Service executed successfully

Error management		
HTTP Code	Error code	Error Description
400	PSD2_01.000.A0002	Missing header parameter: {field name}
400	PSD2_01.000.A0003	Invalid header parameter: {field name}
400	PSD2_01.190.A0030	Entity not found
400	PSD2_01.190.A0018	Inconsistent consent resource status
400	PSD2_01.000.A0017	Repetition of query param not admitted: {param}
400	PSD2_00.189.A0005*	ASPSP provided error. Reference to the TPP-messages.
401	PSD2_01.000.A0009	Invalid signature
401	PSD2_00.189.A0005*	ASPSP provided error. Reference to the TPP-messages.
401	PSD2_01.190.A0032	Invalid consent resource
401	PSD2_01.190.A0034	The consent was created by this TPP but has expired and needs to be renewed.
401	PSD2_01.190.A0035	The consent has been invalidated by the ASPSP
403	PSD2_01.001.A0004	Unknown ASPSP
403	PSD2_01.001.A0020	Unknown TPP
403	PSD2_00.189.A0005*	ASPSP provided error. Reference to the TPP-messages.
404	PSD2_00.189.A0005*	ASPSP provided error. Reference to the TPP-messages.
406	PSD2_00.189.A0005*	ASPSP provided error. Reference to the TPP-messages.
409	PSD2_01.000.A0001	Operation not allowed
429	PSD2_01.190.A0033	The access on the account has been exceeding the consented multiplicity per day.
500	PSD2_00.000.A0000	Generic Error

* In this case the error is provided by the ASPSP. The http code and the TPP-Messages are defined by using the BG specification. Refers to Message Code section for details.

Example of readCardTransactionList

GET

https://<IAM_DNS>/platform/enabler/psd2orchestrator/ais/1.0.0/card-accounts/qwer3456tzui7890/transactions?date_from=2017-07-01&date_to=2017-07-30

Request:

```
HEADERS:
aspsp-code=12345
consent-id=xxxxx
Content-Type: application/json
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
```

```
BODY:
N/A
```

Response:

```
HTTP Status code: 200

HEADERS:
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Content-type: application/json
```

```
BODY:

{
      "transactions": {
            "booked": [{
                        "cardTransactionId": "14402567800002019179662",
                        "terminalId": "221366000186177",
                        "transactionDate": "2019-01-17",
                        "transactionAmount": {
                              "currency": "EUR",
                              "amount": "127.85"
                        },
                        "cardAcceptorId": "AMAZON IT",
                        "cardAcceptorAddress": null,
                        "cardAcceptorCategoryCode": "5999",
                        "maskedPan": "493592******8444",
                        "invoiced": false
                  },
                  {
                        "cardTransactionId": "14402567800002020938475",
                        "terminalId": "221366000186177",
                        "transactionDate": "2019-01-17",
                        "transactionAmount": {
                              "currency": "EUR",
                              "amount": "15.00"
                        },
                        "cardAcceptorId": "Netflix",
                        "cardAcceptorAddress": null,
                        "cardAcceptorCategoryCode": "8473",
                        "maskedPan": "493592******8444",
                        "invoiced": false
                  }
            ],
            "_links": {
                  "cardAccount": {
                        "href": "/v1/card-accounts/201903148444"
                  }
            }
      },
      "cardAccount": [{
            "maskedPan": "493592******8444"
      }]
}
```

6 tppRetrievalServices

In this section are described the APIs to manage the TPP retrieval services.

API	Description	Visibility	Access Token
retrieveAspsps	This api allows retrieval of a list of ASPSPs subscribed to the PSD2-Gateway according to te search criteria	Public	Application

There’s an example of a JSON Request/Response below every API .

6.1 retrieveAspsps

This api allows a TPP tp retrieve a list of ASPSPs subscribed to the PSD2-Gateway according to search criteria

Description:
The API allows a TPP to retrieve a paginated list of the ASPSPs subscribed to the PSD2 Gateway. This search can be carried out by filtering the ASPSP company name (business_name) or the ASPSP code (aspsp_code). The PSD2 Gateway provides its main attributes for each ASPSP in the response, and a list of the managed products (aspsp_product_code). In case an ASPSP would manage several products, the TPP shall present that list to the PSU to allow him to choose the right one to start a payment order request or to start an account information consent establishment. The chosen aspsp_product_code will drive the PSU authentication approach (simple and strong) that will be used to complete the requested transaction.

The TPP must provide the certificate, issued by the competent National Authority and qualifying the TPP as a PISP, AISP or PIISP, in order to access to this API.

Tags: aspsp, retrieve

PROTOCOL	HTTP
PATH (Public Exposure)	https://<ASPSP_FQDN>/platform/enabler/psd2orchestrator/tpp/aspsps/1.0.0
METHOD	GET

Parameter description

INPUT				
HEADER PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***
authorization:Bearer	The value of the access token	M-Public	-	String
tpp-registration-number	Caller TPPidentifier.	M	255	String
cpaas-transaction-id	The request identifier	M	255	String
QUERY PARAM				
Parameter	***Description***	***Mandatory / Optional***	***Max Length***	***Type***
business_name	ASPSP Company name The field is searched using like criteria This field can be used alternatively to the aspsp_code field.	O	255	String
aspsp_code	ASPSP identifier Multi-Values in OR (CVS String) This filter can be used alternatively to business_name. Max elements list size is 5 (for CSV Strings each element max 20 chars)	O	104	String

	This field can be used alternatively to the business_name field.			
offset	Requested page number. Only positive integers allowed. **Mandatory if limit param is provided	O	Unbounded	String
limit	Max elements per page to be returned. Only positive integers allowed. For this API the maximum value is set to 5 elements per page (which is also the default value)	O	-	String
sort	Field to be used for sort capability. Only one sort parameter can be specified for this API. The '-' preceding the parameter means descending order. If no sort field is indicated, the default sorting is by business_name ascending. Admitted values: - aspsp_code - business_name - creation_date - updated_date	O	-	String

OUTPUT

HEADER

Parameter	***Description***	***Mandatory / Optional***	***Type***
cpaas-transaction-id	The transaction identifier provided in input.	M	String
cpaas-platform-transaction-id	Transaction identifier generated by Platform - Automatically set through API Manager	O	String
cpaas-total-elements	If query param offset=1, this field contains the total elements of the query executed on backend, before the pagination	O	String
cpaas-total-pages	If query param offset=1, this field contains the number of pages provided by the query executed on backend, before the pagination	O	String

BODY

Parameter	***Description***	***Mandatory / Optional***	***Type***
errorManagement	Object identifying the error Provided only if there is an error	O	Object
• errorCode	Code that identifies error occurred	M	String
• errorDescription	Error description	M	String
aspsps	List of ASPSPs	O	List<Object>
• id	UUID of business user	M	String
• businessName	TPP Company Name	M	String
• aspspCode	The code of the ASPSP	M	String
• status	Status identifier	M	String
• attribute	Attributes for the business user In the attributes list will be delivered ASPSP attrbiutes like: address, city, countryRegion, zipCode, vatCode	O	List<Object>
○ attributeName	Name of the business user's attribute	M	String
○ attributeValue	Value of the business user's attribute	M	String
• creationDate	Creation date. Date time with time zone. Format: YYYY-MM-DDTHH:mm:ss.sssZ	M	String
• updatedDate	Updated date. Date time with time zone. Format YYYY-MM-DDTHH:mm:ss.sssZ	O	String
• aspspProductsList	The ASPSP products list object	M	List<Object>
○ aspspProductUuid	The uuid of the product	M	String

○ aspspProductCode	The aspsp product code	M	String
○ aspspProductDescription	The aspsp product description	O	String

HTTP Code	Result Description
200	Service executed successfully

Error management		
HTTP Code	Error code	Error Description
400	PSD2_01.000.A0002	Missing header parameter: {field name}
400	PSD2_01.000.A0003	Invalid header parameter: {field name}
400	PSD2_01.000.A0026	Invalid query parameter: {field name}
403	PSD2_01.001.A0020	Unknown TPP
500	PSD2_00.000.A0000	Generic Error

Example of retrieveAspsps

https://<ASPSP_FQDN>/platform/enabler/psd2orchestrator/tpp/aspsps/1.0.0?aspsp_code=aspspTestProv1005

Request:

```
HEADERS:
Content-Type:application/json
cpaas-transaction-id:123456-123456-123456-123456
tpp-registration-number:123456
```

```
BODY:
N/A
```

Response:

```
HTTP Status code: 200

HEADERS:
cpaas-total-elements: 1
cpaas-total-pages: 1
cpaas-transaction-id: 123456-123456-123456-123456
```

```
BODY:
{
  "aspsps" : [
    {
      "id" : "201889e5-51c5-4bfb-906e-f5ddb7f27379",
      "aspspCode" : "aspspTestProv1005",
      "businessName" : "AUTO_BU_06748",
      "status" : "1",
      "attribute" : [
        {
          "attributeName" : "zipCode",
          "attributeValue" : "80020"
        },
        {
          "attributeName" : "name",
          "attributeValue" : "CDEnterprise"
        },
        {
          "attributeName" : "businessName",
          "attributeValue" : "BU_CD_Enterprise"
        },
        {
          "attributeName" : "city",
          "attributeValue" : "Naples"
        }
      ],
      "creationDate" : "2018-09-13T00:48:50Z",
      "updatedDate" : "2018-12-11T11:18:14.000Z",
      "aspspProductsList" : [
        {
          "aspspProductCode" : "prod00x",
          "aspspProductDescription" : "descr00x",
          "aspspProductUuid" : "84716785-65de-4593-8a1f-661a1b84b1c0"
        }
      ]
    }
  ]
}
```

7 Appendix

7.1 ISO Related Basic Types

Type	*Description*	*Max Lenght*	*ISO Standard*
PurposeCode	This code is used for defining the purpose of the payment.	4	ExternalPurpose1Code from ISO 20022
CashAccountType	Specifies the nature, or use, of the cash account in the format of character string with a maximum length of 4 characters.	4	ExternalCashAccountType1Code from ISO 20022
BankTransactionCode	Specifies the bank transaction code domain	4	ISO 20022
BICFI	Define a standard format Bank Identifier Code according to ISO 9362. pattern = "[A-Z]{6,6}[A-Z2-9][A-NP-Z0-9]([A-Z0-9]{3,3}){0,1}"	11	BICIdentifier ISO 20022
IBAN	An identifier used internationally by financial institutions to uniquely identify the account of a customer at a financial institution. According to ISO 13616: Pattern: [A-Z]{2}[0-9]{2}[A-Z0-9]{1,30}	34	IBANIdentifier ISO 20022
BBAN	Specifies the Basic Bank Account Number, an Identifier used nationally by financial institutions, ie, in individual countries, generally as part of a National Account Numbering Scheme(s), which uniquely identifies the account of a customer. pattern = "[a-zA-Z0-9]{1,30}"	30	BBANIdentifier ISO 20022
CurrencyCode	Codes following ISO 4217 Alpha 3 pattern = "[A-Z]{3,3}"	3	Codes following ISO 4217 Alpha 3
CountryCode	Define a codified Country name. According to ISO 3166: pattern = "[A-Z]{2,2}"	2	CountryCode ISO 20022

7.2 Complex Data Types and Code Lists

7.2.1 AuthenticationType

More authentication types might be added during implementation projects and documented in the ASPSP documentation.

Parameter	*Description*
SMS_OTP	An SCA method, where an OTP linked to the transaction to be authorised is sent to the PSU through a SMS channel.
CHIP_OTP	An SCA method, where an OTP is generated by a chip card, e.g. an TOP derived from an EMV cryptogram. To contact the card, the PSU normally needs a (handheld) device. With this device, the PSU either reads the challenging data through a visual interface like flickering or the PSU types in the challenge through the device key pad. The device then derives an OTP from the challenge data and displays the OTP to the PSU.
PHOTO_OTP	An SCA method, where the challenge is a QR code or similar encoded visual data which can be read in by a consumer device or specific mobile app. The device resp. the specific app than derives an OTP from the visual challenge data and displays the OTP to the PSU.
PUSH_OTP	An OTP is pushed to a dedicated authentication APP and displayed to the PSU.

7.2.2 BalanceType

Parameter	*Description*
closingBooked	Balance of the account at the end of the pre-agreed account reporting period. It is the sum of the opening booked balance at the beginning of the period and all entries booked to the account during the pre-agreed account reporting period.
expected	Balance composed of booked entries and pending items known at the time of calculation, which projects the end of day balance if everything is booked on the account and no other entry is posted.
authorised	The expected balance together with the value of a pre-approved credit line the ASPSP makes permanently available to the user.
openingBooked	Book balance of the account at the beginning of the account reporting period. It always equals the closing book balance from the previous report.
interimAvailable	Available balance calculated in the course of the account 'servicer's business day, at the time specified, and subject to further changes during the business day. The interim balance is calculated on the basis of booked credit and debit items during the calculation time/period specified.
forwardAvailable	Forward available balance of money that is at the disposal of the account owner on the date specified.

7.2.3 TransactionStatus

The transaction status is filled with value of the ISO20022 data table.

Code	*Name*	*ISO 20022 Definition*
ACCP	AcceptedCustomerProfile	Preceding check of technical validation was successful. Customer profile check was also successful.
ACSC	AcceptedSettlementCompleted	Settlement on the debtor's account has been completed. 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
ACSP	AcceptedSettlementInProcess	All preceding checks such as technical validation and customer profile were successful and therefore the payment initiation has been accepted for execution.
ACTC	AcceptedTechnicalValidation	Authentication and syntactical and semantical validation are successful
ACWC	AcceptedWithChange	Instruction is accepted but a change will be made, such as date or remittance not sent.
ACWP	AcceptedWithoutPosting	Payment instruction included in the credit transfer is accepted without being posted to the creditor customer's account.
RCVD	Received	Payment initiation has been received by the receiving agent.
PDNG	Pending	Payment initiation or individual transaction included in the payment initiation is pending. Further checks and status update will be performed.
RJCT	Rejected	Payment initiation or individual transaction included in the payment initiation has been rejected.

7.2.4 Consent Status

Code	*Description*
received	The consent data have been received and are technically correct. The data is not authorised yet.
rejected	The consent data have been rejected e.g. since no successful authorisation has taken place.
valid	The consent is accepted and valid for GET account data calls and others as specified in the consent object.
revokedByPsu	The consent has been revoked by the PSU towards the ASPSP.
expired	The consent expired.
terminatedByTpp	The corresponding TPP has terminated the consent by applying the DELETE method to the consent resource.
replaced	The consent data have been replaced when a new recurring consent is finalized.
invalidated	The consent data have been invalidated when a consent is invalidated by the ASPSP.
pendingExpired	The consent data have been set in this status when you try to use a consent valid but expired.

7.2.1 Message Code

The permitted message error codes and related HTTP response codes are listed below:

Service Unspecific HTTP Error Codes		
HTTP response code	***Message code***	***Message description***
401	SIGNATURE_INVALID	Application layer eIDAS Signature for TPP authentication is not correct.
401	SIGNATURE_MISSING	Application layer eIDAS Signature for TPP
400	FORMAT_ERROR	Format of certain request fields are not matching the XS2A requirements. An explicit path to the corresponding field might be added in the return message.
400	PARAMETER_NOT_SUPPORTED	The parameter is not supported by the API provider.
401	PSU_CREDENTIALS_INVALID	The PSU-ID cannot be matched by the addressed ASPSP or is blocked, or a password resp. OTP was not correct.
400	SERVICE_INVALID	The addressed service is not valid for the addressed resources or the submitted data.
403	SERVICE_BLOCKED	This service is not reachable for the addressed PSU due to a channel independent blocking by the ASPSP. Additional information might be given by the ASPSP.
401	CORPORATE_ID_INVALID	The PSU-Corporate-ID cannot be matched by the addressed ASPSP
403	CONSENT_UNKNOWN	The Consent-ID cannot be matched by the ASPSP relative to the TPP.
401	CONSENT_INVALID	The consent was created by this TPP but is not valid for the addressed service/resource.
401	CONSENT_EXPIRED	The consent was created by this TPP but has expired and needs to be renewed.
401	TOKEN_UNKNOWN	The OAuth2 token cannot be matched by the ASPSP relative to the TPP.
401	TOKEN_INVALID	The OAuth2 token is associated to the TPP but is not valid for the addressed service/resource.
401	TOKEN_EXPIRED	The OAuth2 token is associated to the TPP but has expired and needs to be renewed.
403	RESOURCE_UNKNOWN	The addressed resource is unknown relative to the TPP.
403	RESOURCE_EXPIRED	The addressed resource is associated with the TPP but has expired, not addressable anymore.
400	TIMESTAMP_INVALID	Timestamp not in accepted time period.
400	PERIOD_INVALID	Requested time period out of bound.
400	SCA_METHOD_UNKNOWN	Addressed SCA method in the Authentication Method Select Request is unknown or cannot be matched by the ASPSP with the PSU.
500	GENERIC_ERROR	Generic error

PIS specific HTTP Error Codes		
HTTP response code	***Message code***	***Message description***
403	PRODUCT_INVALID	The addressed payment product is not available for the PSU .
404	PRODUCT_UNKNOWN	The addressed payment product is not supported by the ASPSP.
400	PAYMENT_FAILED	The payment request failed.
400	EXECUTION_DATE_INVALID	The requested execution date is not a valid execution date for the ASPSP.
401	REQUIRED_KID_MISSING	The payment initiation has failed due to a missing KID. This is a specific message code for the Norwegian market, where ASPSP can require the payer to transmit the KID.

AIS specific HTTP Error Codes		
HTTP response code	***Message code***	***Message description***
401	CONSENT_INVALID	The consent definition is not complete or invalid. In case of being not complete, the bank is not supporting a completion of the consent towards the PSU. Additional information will be provided.
400	SESSIONS_NOT_SUPPORTED	The combined service flag may not be used with this ASPSP.
429	ACCESS_EXCEEDED	The access on the account has been exceeding the consented multiplicity per day.
406	REQUESTED_FORMATS_INVALID	The requested formats in the Accept header entry are not matching the formats offered by the ASPSP.

PIIS specific HTTP Error Codes		
HTTP response code	***Message code***	***Message description***
400	CARD_INVALID	Addressed card number is unknown to the ASPSP or not associated to the PSU.
400	NO_PIIS_ACTIVATION	The PSU has not activated the addressed account for the usage of the PIIS associated with the TPP.

7.3 Json Data Structure

7.3.1 aspspCredentials

Field name	Type	Description
credentialsId	String	The field id specified by the ASPSP. It identifies, uniquely, the credentials on ASPSP system.
isSecret	String/Boolean	If true, it indicates that the field is a password so it should be secreted.
labelList	Array	The list of the labels to show to the end user. They are internationalized.
- label	String	The labe associated to the credentials to show to the end user.
- language	String	Label internationalization. It specifies the language of the label.

JSON data sample:

```
{
        "aspspCredentials": [{
                "credentialId": "my1ASPSPId",
                "isSecret": "true",
                "labelList": [{
                        "label": "myFirstLabel",
                        "language": "EN"
                },
                {
                        "label": "laMiaPrimaEtichetta",
                        "language": "IT"
                }]
        },
        {
                "credentialId": "my2ASPSPId",
                "isSecret": "true",
                "labelList": [{
                        "label": "mySecondLabel",
                        "language": "EN"
                },
                {
                        "label": "laMiaSecondaEtichetta",
                        "language": "IT"
                }]
        }]
}
```