DATADESIGN

BS

Schnittstellenbeschreibung

X S 2 A - A c c e c s s t o A c c o u n t

Übersicht

Titel	Thema	Datei	Autor
Schnittstellenbeschreibung	XS2A - Accecss to Account	Access to Account - Schnittstellenbeschreibung-BS.docx	Nimete Sallahaj, Stefan Arnecke

Seitenumfang	Version	Status	Datum
68 Seiten	1.0.1.5	Release	20.01.2020

Änderungsverzeichnis

Version	Datum	Autor	Änderungen/Kommentar
1.0.0.0	22.01.2019	Sallahaj, Nimete	Dokument neu erstellt
1.0.0.1	24.01.2019	Sallahaj, Nimete	Kapitel 1,2, 3
1.0.0.2	25.01.2019	Sallahaj, Nimete	Kapitel 4
1.0.0.3	29.01.2019	Sallahaj, Nimete	Kapitel 4
1.0.0.4	30.01.2019	Sallahaj, Nimete	Kapitel 5,6
1.0.0.5	01.02.2019	Sallahaj, Nimete	Kapitel 7
1.0.0.6	04.02.2019	Sallahaj, Nimete	Kapitel 11,12
1.0.0.7	06.02.2019	Sallahaj, Nimete	Korrektur
1.0.0.8	12.02.2019	Sallahaj, Nimete	Korrektur, Grafiken
1.0.0.9	13.02.2019	Sallahaj, Nimete	Korrektur
1.0.1.0	18.02.2019	Sallahaj, Nimete	Korrektur
1.0.1.1	19.02.2019	Sallahaj, Nimete Arnecke, Stefan	Korrektur
1.0.1.2	27.02.2019	Sallahaj, Nimete, Arnecke Stefan	Korrektur
1.0.1.3	28.02.2019	Gebhard Fabian, Nimete Sallahaj	Korrektur
1.0.1.4	11.03.2019	Fritz Markus, sNimete Sallahaj	Tabellenformatierung
1.0.1.5	17.01.2020	Fritz Markus	Aktualisierung Accountinformation in Bezug auf Kontoinhabername und Daueraufträgsbestand

Inhaltsverzeichnis

1 Einführung

Dieses Dokument beschreibt die XS2A–Implementierung und verwendet hierfür die PSD2-kompatible XS2A-Spezifikation der Berlin Group (NextGenPSD2, 2018) als Grundlage. Die XS2A- Schnittstelle ist als B2B-Schnittstelle zwischen einem Drittdienstleister und einem Finanzdienstleister (Bank) konzipiert. In diesem Dokument werden die die Implementierungen von Meldungen und detaillierten Datenstrukturen für die XS2A-Schnittstelle beschrieben.

2 Festlegungen

2.1 Konditionenen

Folgende Konditionen sind in allen Tabellen mit Endpoint/Methode/Attribut/Element möglich: M (Mandatory), K (Konditional) und O (Optional). In allen Tabellen ist die Spalte mit den Konditionen mit K beschriftet.

Kondition	Bedeutung	Erläuterung
M	Muss	Endpoint/Methode/Attribut/Element muss vorhanden sein und ist inhaltlich korrekt zu füllen.
K	Konditional	Endpoint/Methode/Attribut/Element ist konditional, d. h. es ist von einer Bedingung abhängig, die gesondert beschrieben ist.
O	Optional	Endpoint/Methode/Attribut/Element ist optional.

2.2 Character-Sets

Der Character-Set der Schnittstelle ist UTF-8 encoded und diese Schnittstellenbeschreibung verwendet nur die Basiselemente wie „String“, „Boolean“, „ISODateTime“, „ISODate“, „UUID“ und „Integer“ sowie ISO–basierte Codelisten. Für ISO-basierte Codes wird in Abschnitt 11.21 auf die entsprechende ISO-Norm verwiesen. Max35Text, Max70Text, Max140Text und Max512Text definieren Character-Sets mit einer maximalen Länge von 35, 70, 140 bzw. 512 Zeichen.

Die Schnittstelle akzeptiert für Strings mindestens folgenden Character-Set:

a b c d e f g h i j k l m n o p q r s t u v w x y z
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
0 1 2 3 4 5 6 7 8 9
/ - ? : ( ) . , ' +
Space

Darüber hinaus werden von der Schnittstelle weitere Character Sets im Kontext von Namen, Adressen und Text akzeptiert und bestimmte Sonderzeichen vor dem Weiterleiten konvertiert.

3 Transport Layer

Die Kommunikation zwischen TPP und der Schnittstelle findet immer durch eine TLS-gesicherte Verbindung mit TLS Version 1.2 oder höher statt. Die TLS-Verbindung wird vom TPP eingerichtet. Es ist nicht erforderlich für jede Transaktion eine neue TLS-Verbindung einzurichten. Der XS2A-Dienst kann die vorhandene TLS-Verbindung beenden, wenn dies aufgrund interner Sicherheitsrichtlinien erforderlich ist. Jede Verbindung ist immer mit einer Client-Authentifizierung des TPP aufzubauen. Beim Aufbau einer TLS-Verbindung ist immer eine Client-Authentifizierung enthalten. Für diese Authentifizierung muss der TPP ein geeignetes Zertifikat für die Website-Authentifizierung verwenden. Dieses Zertifikat muss von einem qualifiziertem Trust Service Provider gemäß der eIDAS (eIDAS, 2014) Regulation ausgestellt werden. Der Inhalt des Zertifikats muss den Anforderungen von EBA-RTS entsprechen (EBARTS, 2018). Das TPP – Zertifikat muss alle Rollen enthalten, zu deren Verwendung der TPP berechtigt ist.

4 Application Layer: Leitprinzipien

4.1 Signieren von Nachrichten im Application Layer

Der TPP muss alle von ihm gesendeten Nachrichten auf Applikationsebene signieren. Diese Signatur ist im HTTP Header einzustellen. Die Elektronische Signatur des TPP muss auf einem qualifizierten Zertifikat basieren. Dieses qualifizierte Zertifikat wird von einem qualifizierten Trust Service Provider gemäß der eIDAS-Verordnung ausgestellt. Der Inhalt des Zertifikats muss den Anforderungen des (EBARTS, 2018) entsprechen. Das Zertifikat des TPP muss die Rollen angeben, für die der TPP berechtigt ist.

Zur Unterstützung der Signaturfunktionalität sind in allen vom TPP gesendeten Nachrichten zusätzlich die folgenden HTTP-Header einzustellen.

Request Header

Attribut	Typ	K	Beschreibung
Digest	String	M	Hashwert des Nachrichtenkörpers
Signature	Vgl. Abschnitt 9.2	M	Eine Signatur der Nachricht durch den TPP auf Application Level.
TPP-Signature-Certificates	String	M	Das zum Signieren der Nachricht verwendete Zertifikat in base64-Kodierung.

Für eine bessere Lesbarkeit wird die Header Definition in den folgenden Abschnitten dieser Dokumentation nicht wiederholt.

4.2 Autorisierung Endpoints

Die Schnittstelle stellt dedizierte Autorisierungs-Endpoints für Payment Initiation Transaktionen und Consenteinrichtungs-Transaktionen zur Verfügung, um die Transaktionsautorisierung durch PSUs abzuwickeln. Für die Einreichung einer erfolgreichen Zahlung müssen zwei Ressourcen angelegt werden, wobei die zweite Sub-Ressource die SCA Autorisierung der Transaktion repräsentiert.

4.3 (API) Zugriffsmethoden

Die folgende Tabelle gibt eine Übersicht der HTTP Zugriffsmethoden, die von den (API) Endpoints und von den durch die API erstellten Ressourcen unterstützt werden.

Bitte beachten Sie, dass die „K“ des beschriebenen Endpoints relativ zum übergeordnetem Knoten seines Pfads gegeben ist, d.h. dass zum Beispiel eine Kondition *M* auf dem Endpoint */v1/consents/{consentId}* nur gilt, wenn der darüberliegende Endpoint */v1/consents* generell unterstützt wird.

Alle von einem TPP aufgerufenen Methoden, die dynamisch erstellte Ressourcen in der Schnittstelle ansprechen, gelten nur für Ressourcen, die zuvor von demselben TPP erstellt wurden.

Bitte beachten Sie auch die zusätzlichen Verweise in der Spalte „Beschreibung“. Die Abschnitte auf die an dieser Stelle verwiesen wird enthalten Beispiele zu den jeweiligen Endpoints.

4.3.1 Payment Endpoints

Endpoints/Ressourcen	Methode	K	Beschreibung
payments/{payment-product}	POST	M	Erstellt eine unter {paymentId} adressierbare Payment Initiation Ressource mit allen Daten, die für das entsprechende Zahlungsprodukt relevant sind. Dies ist der erste Schritt in der API, um die zugehörige Zahlung zu initiieren. Siehe Abschnitt 5.2.1
payments/{payment-product}/{paymentId}	GET	M	Liest die Details einer zuvor initiierten Zahlung. Siehe Abschnitt 5.4
payments/{payment-product}/{paymentId}/status	GET	M	Liest den Transaktionsstatus der Zahlung. Siehe Abschnitt 5.3
{payment-service}/{payment-product}/{paymentId}/authorisations	POST	M	Erstellt eine Autorisierungs-Sub-Ressource und startet den Autorisierungsprozess, indem zusätzlich Authentifizierungs- und autorisierungsbezogene Daten übertragen werden. Siehe Abschnitt 7.1.
{payment-service}/{payment-	GET	M	Liest eine Liste aller erstellten Autorisierungs-Sub-Ressourcen-

Endpoints/Ressourcen	Methode	K	Beschreibung
product}/{paymentId}/authorisations			IDs.
{payment-service}/{payment-product}/{paymentId}/authorisations/{authorisationId}	PUT	M	Aktualisiert bei Bedarf die Daten der Autorisierungsressource. Diese kann zum Beispiel eine Zahlung innerhalb des Embedded-SCA-Ansatzes genehmigen. Unabhängig vom SCA-Ansatz unterstützt die Methode z. B. die Auswahl der Authentifizierungsmethode die PSU Authentifizierung. Abschnitt 7.2 und Abschnitt 7.3.
{payment-service}/{payment-product}/{paymentId}/authorisations/{authorisationId}	GET	M	Liest den SCA-Status der Autorisierung. Abschnitt 7.4
payments/{payment-product}/{paymentId}	GET	M	Liest die Details einer zuvor initiierten Zahlung. Siehe Abschnitt 5.4
payments/{payment-product}/{paymentId}/status	GET	M	Liest den Transaktionsstatus der Zahlung. Siehe Abschnitt 5.3

4.3.2 Konto Endpoint

Endpoints/Ressourcen	Methode	K	Beschreibung
accounts	GET	M	Liest alle Konten-Identifikationen, auf die der PSU dem TPP über den /consents Endpoint Zugriff gewährt hat. Darüber hinaus werden relevante Informationen zu den Konten und Hyperlinks zu entsprechenden Account Informationsressourcen bereitgestellt, sofern bereits ein entsprechender Consent erteilt wurde. **Anmerkung**: Es ist zu beachten, dass der /consents Endpoint ggf. Zugriff auf alle verfügbaren Zahlungskonten eines PSU anbietet. In diesem Fall liefert dieser Endpoint die Informationen zu allen verfügbaren Zahlungskonten des PSU. Siehe Abschnitt 6.4.1
accounts/{account-id}	GET	M	Liefert detaillierte Salden für das adressierte Konto. Siehe Abschnitt 6.4.2
accounts/{account-id}/balances	GET	M	Gibt detaillierte Buchungsinformationen über das adressierte Konto. Abschnitt 6.4.3.
accounts/{account-id}/transactions	GET	M	Liest die Kontoumsätze eines bestimmten Kontos. Für ein bestimmtes Konto gibt es zusätzliche Parameter, z.B. die Attribute "dateFrom" und "dateTo". Siehe Abschnitt 6.4.4

4.3.3 Einwilligungsendpoint

Endpoints/Ressourcen	Methoden	K	Beschreibung
consents	POST	M	Erstellt eine Consent-Ressource, die Zugriffsrechte für dedizierte Konten einer bestimmten PSU-ID definiert. Wird das Zugriffsrecht ohne Konten eingestellt, so gilt dieses Recht für alle für PSD2 freigeschaltenen Konten der PSU. Alternativ kann statt der dedizierten Angabe von Konten und Zugriffsrechten das Attribut „availableAccounts“ verwendet werden, um einen Consent für den Kontenzugriff auf allen verfügbaren Konten zu erstellen. Siehe Abschnitt 6.2.1.
consents/{consentId}	GET	M	Liest die Definition der gegebenen Consentressource {consentId} inklusive Gültigkeitsstatus. Siehe Abschnitt 6.2.2.
	DELETE	M	Beendet den adressierten Consent. Siehe Abschnitt 6.3.
consents/{consentId}/status	GET	M	Liest den Consent Status der adressierten Ressource. Siehe Abschnitt 6.2.2.
consents/{consentId}/authorisations	POST	M	Erstellt eine Autorisierungs-Sub-Ressource und startet den Autorisierungsprozess, der zusätzlich authentifizierungs- und autorisierungsbezogene Daten übertragen kann. Die Schnittstelle könnte die Verwendung dieser Accessmethode überflüssig machen, da die entsprechende Autorisierungsressource nach der Übermittlung der Consent-Daten mit dem ersten POST von der Schnittstelle erstellt wird. Siehe Abschnitt 7.1.
consents/{consentId}/authorisations/{authorisationId}	PUT	M	Aktualisiert Daten für die Autorisierung. Mit der Embedded SCA Ansatz kann ein Consent autorisiert werden. Unabhängig vom SCA-Ansatz unterstützt die Methode z.B. die Auswahl der Authentifizierungsmethode und die PSU-Authentifizierung. Siehe Abschnitt 7.2 und Abschnitt 7.3.
consents/{consentId}/authorisations/{authorisationId}	GET	M	Liest den SCA-Status der Autorisierung des Consents. Siehe Abschnitt 7.4

4.3.4 Bonitäts- Endpoint (Funds-Confirmations Endpoint)

Endpoints/Resources	Methode	K	Beschreibung
funds-Confirmations	POST	M	Prüft, ob zum Zeitpunkt der Anfrage ein bestimmter Betrag auf einem Konto verfügbar ist. Siehe Abschnitt 8.1

4.4 Status Information

4.4.1 Status Information für PIS

Die Backend-Systeme der Schnittstelle unterstützen für Zahlungen einen Transaktionsstatus, welcher in der ISO20022 definiert ist und als Datenelement „transactionStatus“ angesprochen wird. Die Schnittstelle liefert diesen Status mit allen Response Messages zurück, nachdem eine Ressource zur Payment Initiation eingerichtet wurde und Fehler aufgetreten sind.

Der Transaktionsstatus einer Zahlung ändert sich während des Initiations-Prozesses, abhängig von den Ergebnissen der Teilschritte wie Formatprüfungen, SCA-Prüfungen, PSU-bezogene Profil-Prüfungen, Funds-Verfügbarkeitsprüfungen oder abhängig vom Start des Backend-Clearing-Prozesse. Am Ende eines Zahlungsprozesses lautet der Transaktionsstatus im Schnittstellen-Backend entweder "RJCT", was für "Rejected" steht, oder "ACTC", was für "AcceptedTechnicalCorrect steht, bei dem die Authentifizierung der PSU, die syntaktische und die semantische (Produkt-) Prüfung erfolgreich waren.

Die Schnittstelle verwendet den Status "PDNG", welcher für "Pending" steht, um den TPP darüber zu informieren, dass der nächste Status der Zahlung noch nicht verfügbar ist. Dies kann der Fall sein, wenn die Autorisierung des PSU alleine nicht für die Ausführung der Zahlung ausreicht. Darüber hinaus informiert die Schnittstelle den TPP über den Status des technischen SCA-Prozesses für eine Payment Initiation innerhalb der GET SCA Status Response Message. Für dieses Statusreporting wird das Datenelement "scaStatus" verwendet.

4.4.2 Statusinformationen für die AIS innerhalb des Abstimmungseinrichtungsprozesses

Der Status einer Consent-Ressource ändert sich wie der Transaktionsstatus der Payment Initiation Ressource während des Initialisierungsprozesses. Im Gegensatz zu dem Payment Initiation Prozesses bezieht sich der Status nur auf die Consentressource der SCA-Prüfungen. Es gibt keine Feedbackschleife mit dem Bank-Backend. Das Datenelement für den Status des Consents ist als „consentStatus“ definiert.

Die einzigen für den Consentstatus unterstützten Codes innerhalb der **Initialisierungsphase** sind „received“, „rejected“ und „valid“. Der aktuelle Status der Consentressource wird mit allen Antwortnachrichten während des Autorisierungsprozesses des Consents zurückgeliefert.

Nach einer erfolgreichen Autorisierung eines Consents durch einen PSU kann die Consent-Ressource ihren Status während ihres Lebenszyklus ändern. Die folgenden Codes werden während der **Lifecycle Phase** des Consents unterstützt:

- „expired“: Der Consent ist abgelaufen (z. B. nach 90 Tagen).

- „revokedByPsu“: Der Consent wurde vom PSU zurückgezogen.
- „terminatedByTpp“. Der AIS hat den Consent beendet.

Der Status kann mit der GET Status Response Message abrufen werden.

Hinweis: Der Status "expired" gilt auch für einmalig nutzbare Consents, sobald sie abgelaufen oder genutzt wurden.

Hinweis: Der Status „terminatedByTpp“ wird auch gesetzt, wenn eine bestehende, wiederkehrende Einwilligung aufgrund der Einrichtung einer neuen, wiederkehrenden Einwilligung für den selben PSU und denselben TPP nicht mehr gültig ist.

Darüber hinaus informiert die Schnittstelle den TPP über den Status des technischen SCA-Prozesses zur Herstellung eines Consents in dem GET SCA-Response Message. Für diese Statusmeldung wird das Datenelement „scaStatus“ verwendet.

5 Payment Initiation Service

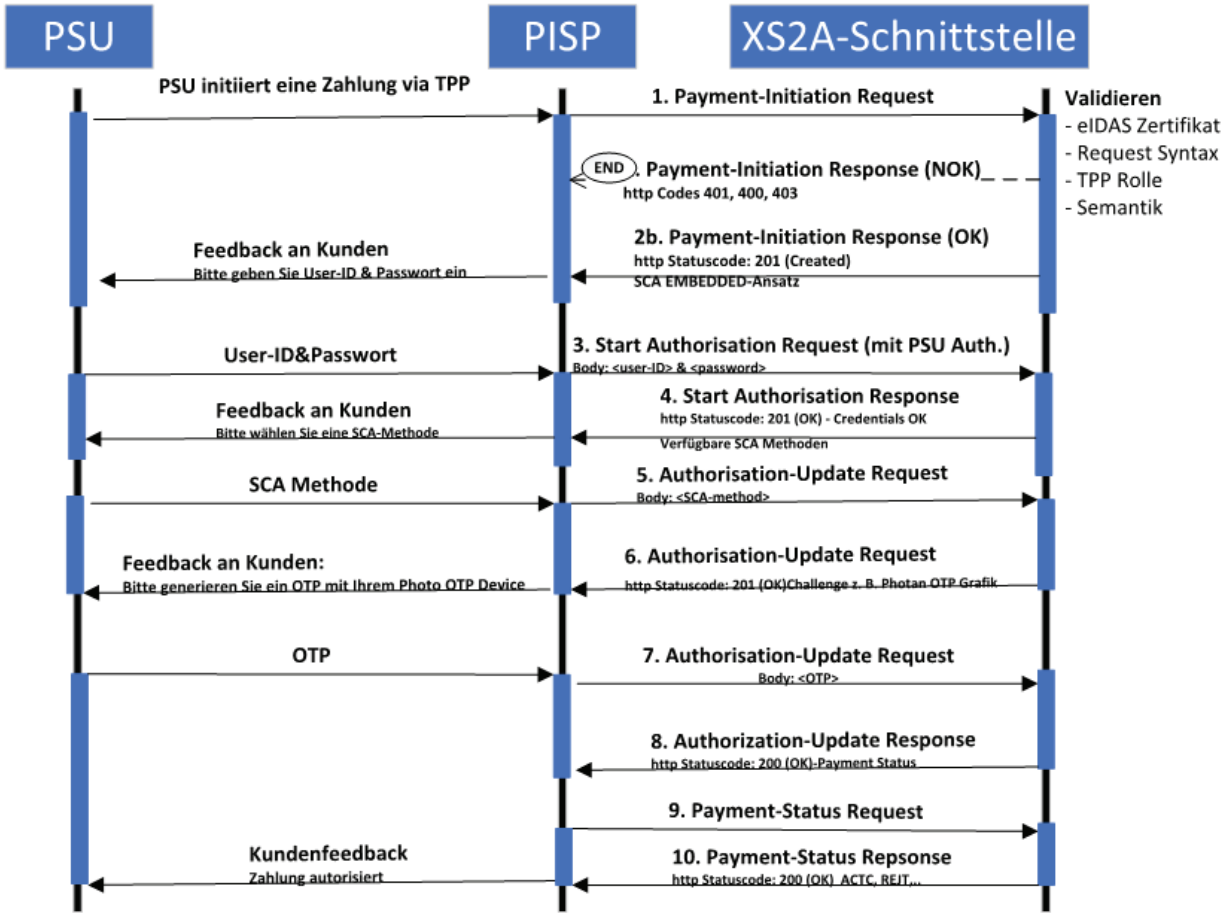
5.1 Payment Initiation Abläufe

Der Payment Initiation Ablauf hängt stark von dem von der Schnittstelle implementierten SCA-Ansatz ab. Der Ablauf für den Embedded-SCA-Ansatz ist ein komplexer Ablauf, welcher sich durch die Verfügbarkeit verschiedener Authentifizierungsmethoden für den PSU weiter differenzieren kann. Im Folgenden werden die verschiedenen API-Abläufe als Übersicht für diese verschiedenen Szenarien bereitgestellt.

Anmerkung: Die Abläufe decken nicht immer alle Abweichungen oder Komplexitäten der Implementierung ab und sind exemplarische Abläufe.

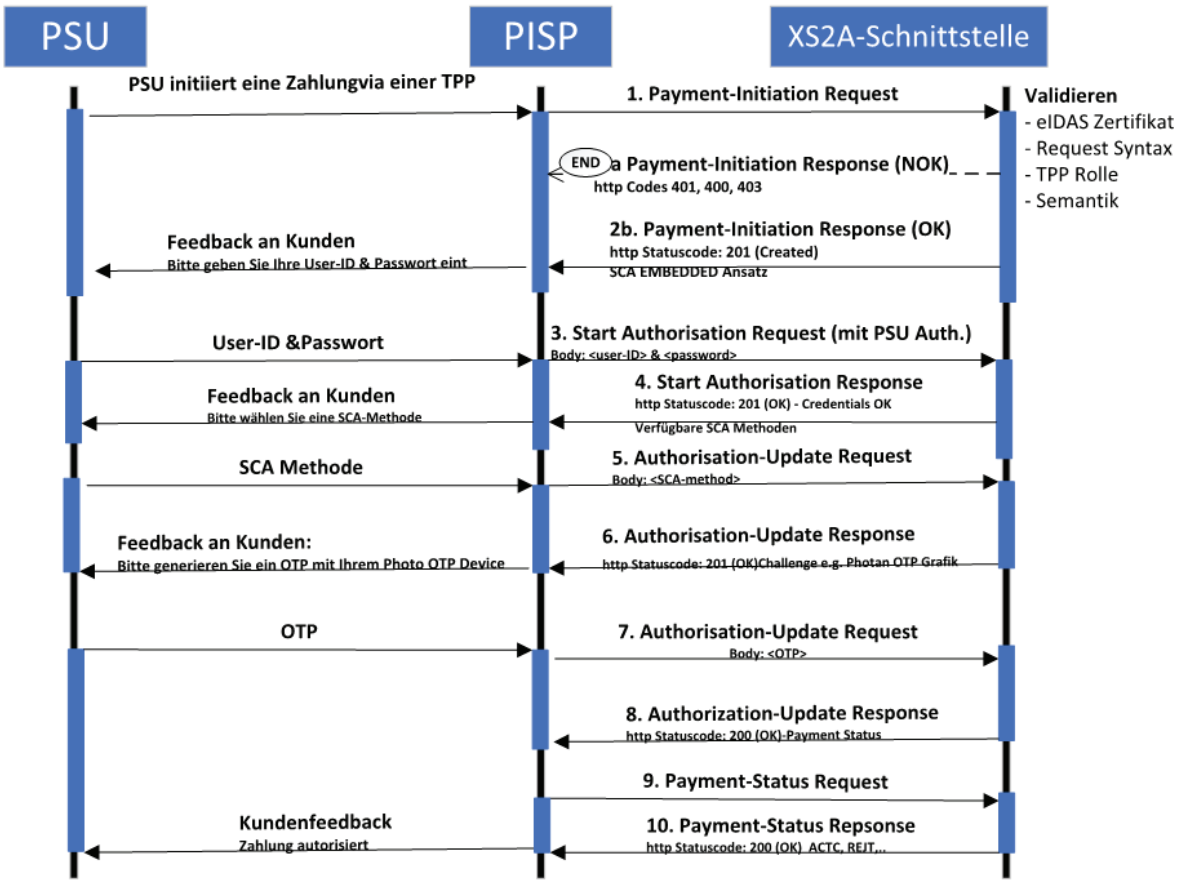
5.1.1 Embedded SCA Ansatz mit nur einer verfügbaren SCA Methode

Falls nur eine SCA-Methode verfügbar ist, wird der Ablauf um den "Authorise Transaction Request“ erweitert, in dem der TPP die Authentifizierungsdaten des Kunden überträgt, z.B. ein OTP mit dynamischen Bezug zu den Transaktionsdetails.



5.1.2 Embedded SCA-Ansatz mit Auswahl einer SCA-Methode

Im folgenden Ablauf wird die Auswahl einer SCA-Methode hinzugefügt. Die Schnittstelle überträgt zunächst die verfügbaren Methoden an den PISP. Wenn es technisch nicht möglich ist, alle Authentifizierungsmethoden zu unterstützten kann der PISP diese filtern. Die verfügbaren Methoden werden dann dem PSU zur Auswahl angeboten.



5.2 Payment Initiation Request

5.2.1 Payment Initiation mit JSON-Kodierung der Zahlungsanweisung

Aufruf

POST /v1/payments/{payment-product}

Erstellt einen Payment Initiation Request an der XS2A-Schnittstelle.

Pfad Parameter

Attribute	Typ	Kondition
payment-product	String	Der adressierte Zahlungsprodukt Endpoint, z. B. für SEPA Credit Transfer (SCT). Die unterstützten Produkte sind: - SEPA-credit-transfers

Abfrage Parameter

Kein Abfrage Parameter.

Request Header

Attribute	Typ	K	Beschreibung
Content-Type	String	M	application/json
X-Request-ID	UUID	M	Eindeutige ID des Requests für den Aufruf festgelegt vom initiierenden Teilnehmer. Dies ist die eindeutige ID des TPP für die Payment Initiation gemäß PSD2 Artikel 46b, 47 und EBARTS Artikel 29.
PSU-ID	String	K	Benutzerkennung des PSU aus der FinTS-Schnittstelle der Bank.
PSU-IP-Address	String	M	Das Headerfeld für die weitergeleitete IP-Adresse entfällt die entsprechende HTTP-Request-IP-Adresse des PSU. Wenn nicht verfügbar, ist die IP-Adresse zu verwenden unter der dieser Request eingereicht wird.

Request Body

Die Zahlungsdaten, die im Request Body transportiert werden sind in Abschnitt 11 dieses Dokuments definiert.

Response Code

Der HTTP-Response Erfolgs-Code ist 201.

Response Header

Attribut	Typ	K	Beschreibung
Location	String	M	Ort der erstellten Ressource (falls angelegt).
X-Request-ID	UUID	M	Eindeutige ID des Requests für den Aufruf vom initiierenden Teilnehmer festgelegt.
ASPSP-SCA-Approach	String	K	Dieses Datenelement muss enthalten sein, wenn der SCA-Ansatz bereits festgelegt ist. Möglicher Wert ist EMBEDDED.

Response Body

Attribut	Typ	K	Beschreibung
transactionStatus	Transaction Status	M	Es können die in Abschnitt 11.20.1 definierten Werte verwendet werden.
paymentId	String	M	Ressourcen-Identifikation der generierten Payment Initiation Resource.
_links	Links	M	Die in dem Response verwendeten Hyperlinks hängen von den dynamischen Entscheidungen des Schnittstelle bei der Bearbeitung der Requests ab. **Anmerkung**: Alle Links sind relative Links. Art der in diesem Response zugelassenen Links: "startAuthorisationWithPsuAuthentication": Der Link zum Autorisierungsendpoint unter dem die Autorisierungs-Sub-Ressource beim Hochladen der PSU-Authentifizierungsdaten generiert wird. "self": Der Link zu der durch diesem Request erstellten Payment Initiation Ressource. Über diesen Link können die Ressourcendaten abgerufen werden. "Status": Der Link, um den Transaktionsstatus der Payment Initiation abzurufen.
psuMessage	Max512Text	O	Anzuzeigender Text für den PSU.
tppMessages	Array der TPP Message Information	O	Nachrichten an dem TPP zur Verarbeitung.

Beispiel

Request

POST https://api.testbank.com/v1/payments/sepa-credit-transfers
Content-Type:application/json
X-Request-ID:99391c7e-ad88-49ec-a2ad-99ddcb1f7721
PSU-IP-Address:192.168.8.78
PSU-GEO-Location:GEO:52.506931,
13.144558PSU-User-Agent:Mozilla/5.0 (Windows NT 10.0; WOW64; rv:54.0) Gecko/20100101 Firefox/54.0
Date:Sun,
06Aug 2017 15:02:37GMT{
  "instructedAmount":{
    "currency":"EUR",
    "amount":"123.50"

```
  },
  "debtorAccount":{
    "iban":"DE40100100103307118608"
  },
  "creditorName":"Merchant123",
  "creditorAccount":{
    "iban":"DE02100100109307118603"
  },
  "remittanceInformationUnstructured":"Ref Number Merchant"
}
```


Response im Falle eines Embedded Ansatzes mit explizitem Autorisierungsstart


```
HTTP/1.x 201 Created
X-Request-ID:99391c7e-ad88-49ec-a2ad-99ddcb1f7721
ASPSP-SCA-Approach:EMBEDDED
Date:Sun,
06Aug 2017 15:03:47GMT
Location:https://www.testbank.com/psd2/v1/payments/sepa-credit-transfers/1234-wertiq-983
Content-Type:application/json{
  "transactionStatus":"RCVD",
  "paymentId":"1234-wertiq-983",
  "_links":{
    "startAuthenticationWithPsuAuthentication":{
      "href":"/v1/payments/sepa-credit-transfers/1234-wertiq-983/authorisations"
    },
    "self":{
      "href":"/v1/payments/sepa-credit-transfers/1234-wertiq-983"
    }
  }
}
```


5.3 Get Transaction Status Request

Aufruf

GET /v1/{payment-service}/{payment-product/}{paymentId}/status

Prüft den Status einer Payment Initiation.

Pfad Parameter

Attribute	Typ	Beschreibung
payment-service	String	Der einzig mögliche Wert ist „payments“
paymentId	String	Ressourcen Identifikation der entsprechenden Zahlung.
payment-product	String	Der einzig mögliche Wert ist „sepa-credit-transfer“. Das Zahlungsprodukt, unter dem die Zahlung unter paymentId ausgelöst wurde. Es wird von der Schnittstelle überprüft, ob das Zahlungsprodukt mit der von paymentId adressierten Payment Initiation übereinstimmt.

Request Header

Attribute	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Aufruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.

Abfrage Parameter

Keine spezifische Abfrage Parameter.

Request Body

Kein Request Body.

Response Code

Der HTTP-Response Code entspricht 200.

Response Header

Attribut	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Aufruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.

Response Body

Attribut	Typ	K	Beschreibung
transactionStatus	Transaction Status	M	Der Status wird in JSON - basiertem Encoding zurückgegeben.

Beispiel

Request

GET *https://api.testbank.com/v1/payments/1234-wertiq-983/status*
Accept: application/json
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date: Sun, 06 Aug 2017 15:04:07 GMT

Response

HTTP/1.x 200 Ok
X-Request-ID:99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date:Sun,
06Aug 2017 15:04:08GMT
Content-Type:application/json{
"transactionStatus":"ACTC",
}

5.4 Get Payment Request

GET /v1/ *{payment-service}/{payment-product}/{paymentId}*

Liefert den Inhalt einer Zahlung.

Pfad Parameter

Attribut	Typ	Beschreibung
payment-service	String	Der einzig mögliche Wert ist "payments".
payment-product	String	Der einzig mögliche Wert ist „sepa-credit-transfer“. Das Zahlungsprodukt, unter dem die Zahlung unter paymentId ausgelöst wurde. Es wird von der Schnittstelle überprüft, ob das Zahlungsprodukt mit der von paymentId adressierten Payment Initiation übereinstimmt.
paymentId	String	Ressourcen Identifikation der entsprechenden Zahlung.

Abfrage-Parameter

Keine spezifische Abfrage Parameter.

Request Headers

Attribut	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Aufruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.

Reponse Code

Der HTTP-Response Code ist 200.

Response Body

Die Response Body enthält den zuvor eingereichten Zahlungsauftrag. Die Datenelement-Einträge können der ursprünglichen Einreichungen abweichen, wenn Inhalte neu formatiert wurden.

6 Account Information Service

Unterstützte Sub-Services

Diese Spezifikation definiert verschiedene Arten von Account Information Services:

- Umsatzberichte für ein bestimmtes Konto.
- Salden eines bestimmten Kontos,
- Eine Liste der verfügbaren Konten,

- Konto Details eines bestimmten Kontos oder der Liste aller verfügbaren Konten im Zusammenhang mit einer erteilten Einwilligung (Consent).

Im Folgenden wird die Liste der verfügbaren und zugänglichen Konten wie folgt definiert:

Definition: Die Liste der **verfügbaren** Konten eines PSU ist die Liste der Konten eines PSU, die über die XS2A-Schnittstelle im Sinne von PSD2 Zahlungskonten zugänglich sind.

Definition: Die Liste der über ein Abonnement eines PSU **zugänglichen** Konten einer Schnittstelle ist die Liste der Konten, bei denen der Consent des PSU für mindestens einen der definierten Konto Informationstypen erteilt wurde.

Hinweis: Der Read Data Request für die Liste des verfügbaren Kontos und für die Kontodetails eines bestimmten Accounts ist syntaktisch identisch. Der Unterschied liegt nur in der zugrundeliegenden Consent-Ressource, die durch den HTTP-Header-Parameter "Consent-ID" bezeichnet wird.

Beispiel: Eine Schnittstelle stellt IBAN1 und IBAN2 einem PSU zur Verfügung. Der PSU hat dem TPP einen Consent zum Zugriff auf Transaktionen und Salden der IBAN1 erteilt. In diesem Fall sind die verfügbaren Konten IBAN1 und IBAN2, die Liste der zugänglichen Accounts besteht nur aus IBAN1.

Einrichten eines Consent und Lesen der Kontoinformationenen

Innerhalb dieser Spezifikation ist der Account Information Service in zwei Phasen unterteilt:

- Einrichten eines Kontoinformations-Abonnements (Consents)

  Innerhalb dieser Phase des Account Information Service gibt der PSU dem AISP die Einwilligung/ das Einverständnis zu folgenden Punkten:

  - die Art des Account Information Services, zu dem ein Zugang gewährt werden soll (siehe die Liste am Anfang dieses Abschnitts),
  - die Multiplizität des Account Information Service, d.h. einen einmaligen oder wiederkehrenden Zugriff und
  - im letzteren Fall über die Dauer des Consent in und gegebenenfalls die Häufigkeit eines wiederkehrenden Antrags.

  Dieser Consent wird dann vom PSU gemäß (EBARTS, 2018) gegenüber der Bank genehmigt.

  Das Ergebnis des Prozesses ist eine Abonnement-Ressource / Consent-Ressource. Ein Link zu dieser Ressource wird innerhalb dieses Prozesses an den AISP zurückgegeben. Das Consentobjekt kann abgerufen werden, indem eine GET-Methode auf dieser

Ressource aufgerufen wird. Dieses Objekt enthält unter anderem die detaillierten Zugriffsrechte, die aktuelle Gültigkeit und ein Consent-ID Token.

- Read Account Data

In dieser Phase erhält der AISP Zugang zu den Kontodaten, wie sie durch den Consent des PSU definiert sind (siehe oben). Der Request zum Lesen von Kontodaten bezieht sich auf die entsprechende Consent-Ressource, indem er den oben genannten Link zu dieser Ressource verwenden.

Der Read Account Data Request zeigt folgendes an:

- die Art der Kontodaten, auf die zugegriffen werden soll,
- gegebenenfalls die Identifizierung des adressierten Kontos,
- ob ein PSU den Request direkt in Echtzeit ausgelöst hat,
- bei Transaktionsberichten als Account-Informationstyp zusätzlich
    - die adressierte Kontokennung und
    - der Zeitraum des Transaktionsberichts

Für den Konten-Zugriff werden die üblichen Bankkonten nach Endpoints getrennt, da die Daten in der Regel auch im Bank-Backend getrennt sind.

Im Falle einer einmaligen Consent wird der Zugriff verweigert, wenn der AISP die Daten mehr als einmal anfordert. Der Lesezugriff auf die Daten wird weiterhin verweigert, wenn die Art des Account Information Services nicht mit dem vereinbarten Service übereinstimmt oder wenn der tatsächliche Zugriff nicht mit der vereinbarten Dauer oder Frequenz übereinstimmt.

Wird der Consent des PSU zum Zugriff auf eine Accountliste erteilt, wird die Häufigkeit des Zugriffs von der Schnittstelle pro abgerufenem Accounts und pro PSU, der den Consent zum Zugriff erteilt hat, überprüft.

Hinweis: Die einzelnen Read Account Data Transaktionen sind eigene Transaktionen nach (XS2A-OR, 2018), so dass eine Transaktionsidentifikation beim Lesen von Transaktionslisten/Kontoauszügen nur im Falle eines Umbruchs mehrfach verwendet wird.

Consent Modelle

Diese XS2A-Schnittstelle unterstützt lediglich eines der von der Berlin Group beschriebenen Consent-Modelle, vgl. (XS2A-OR, 2018):

Detaillierter Consent

Das Consent Management wird zwischen TPP und PSU abgewickelt. Der TPP übermittelt dann die detaillierten Consentinformationen-PSU-Identifikation, Services und betroffene Account-Nummern - an die Schnittstelle zur Autorisierung durch den PSU. Die Schnittstelle zeigt die Consentdetails für den PSU an, wenn sie die SCA durchführt.

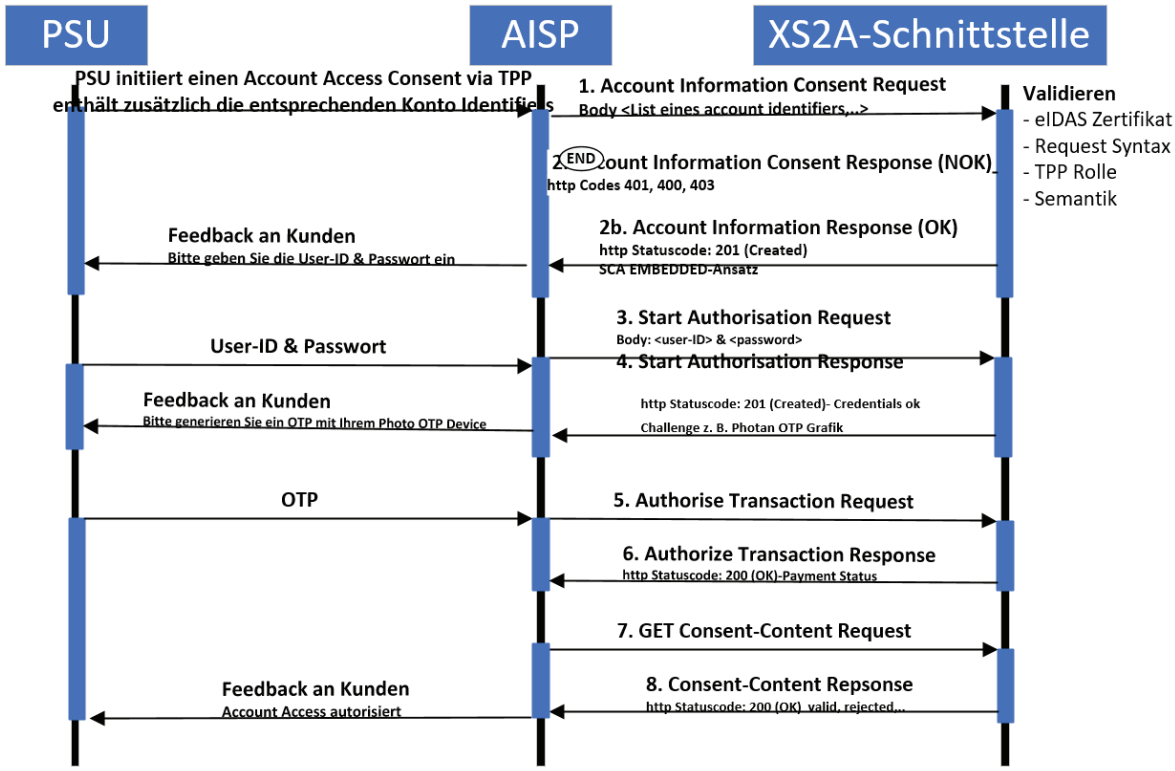
6.1 Account Information Service Ablauf

Wie bei Payment Initiation ist zu beachten, dass die folgenden Abläufe nicht alle möglichen Variationen abdecken.

6.1.1.1 Embedded SCA Approach mit nur einer verfügbaren SCA Methode

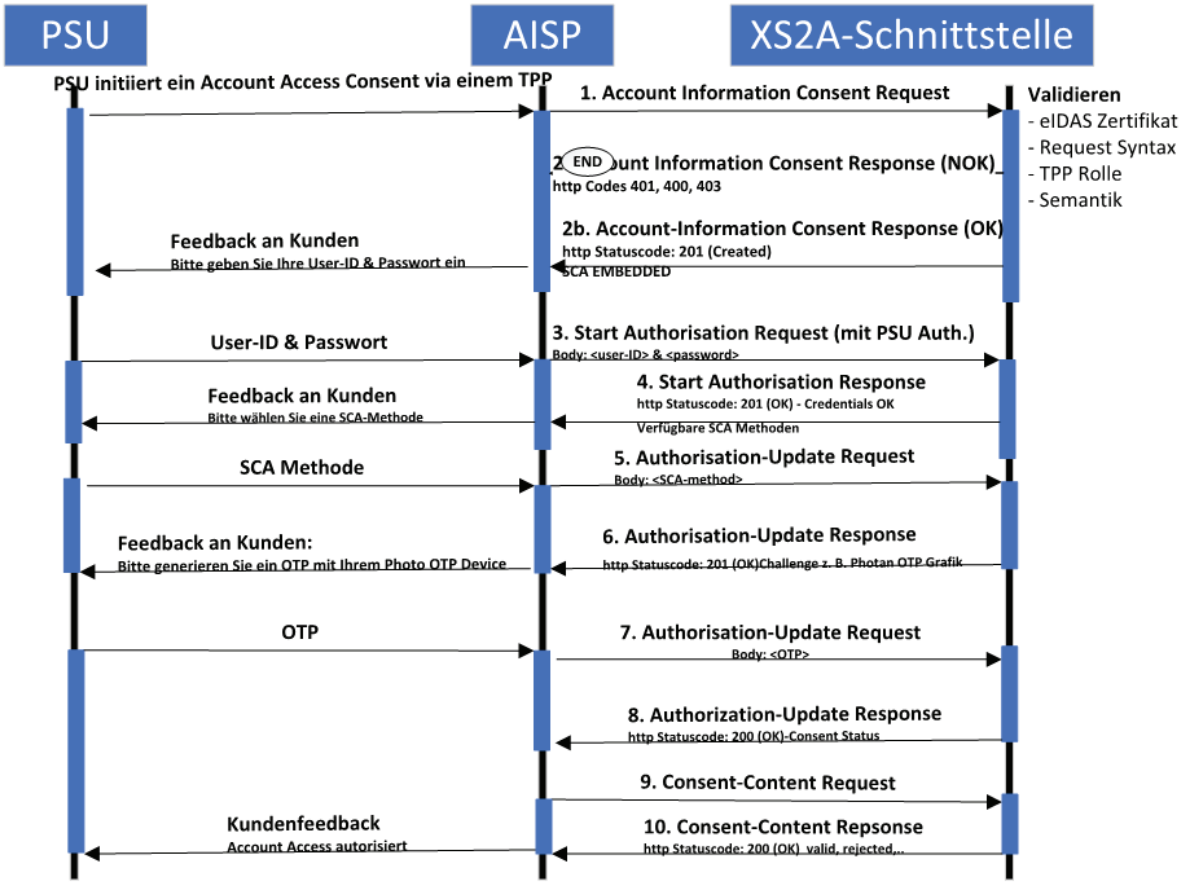
Im Folgenden werden einige exemplarische Abläufe dargestellt. Der PSU muss sich normalerweise mit einem (ersten Faktor) authentifizieren, bevor dem PISP Details zu Konten oder SCA-Methoden zur Verfügung stehen.

Falls nur eine SCA-Methode verfügbar ist, wird der Ablauf um den "Authorise Transaction Request" erweitert, wobei (der TPP) die Authentifizierungsdaten des Kunden übermittelt werden, z.B. ein OTP mit dynamischem Bezug zu den Transaktionsdetails.



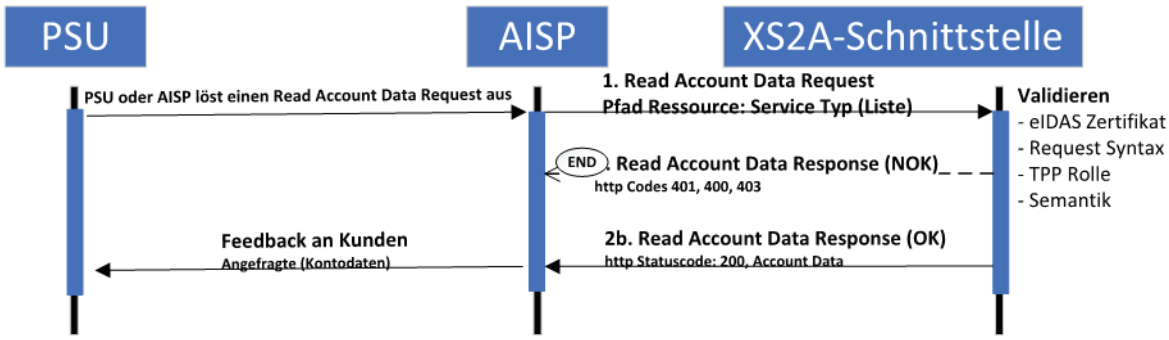
6.1.1.2 Embedded SCA Approach mit Auswahl einer SCA-Methode

Im folgenden Ablauf wird eine Auswahl einer SCA-Methode hinzugefügt, falls die Schnittstelle mehrere SCA-Methoden für den entsprechenden PSU unterstützt. Die Schnittstelle überträgt zunächst die verfügbaren Methoden an den AISP. Der AISP kann sie filtern, wenn nicht alle Authentifizierungsmethoden technisch unterstützt werden können. Die verfügbaren Methoden werden dann dem PSU zur Auswahl angeboten.



6.1.2 Read Account Data Ablauf

Der lesende Account Datenablauf ist unabhängig vom entsprechenden Consent Managementablauf. Es handelt sich um einen einfachen Request/Response-Prozess wie folgt:



6.2 Establish Account Information Consent

In diesem Abschnitt wird der Establish Account Information Consent (Einwilligung) Prozess für die XS2A-Schnittstelle beschrieben.

6.2.1 Account Information Consent Request für dedizierte Accounts

Aufruf

POST /v1/consents

Erstellt eine Consent Ressource für Account Information bezüglich des Zugriffs auf die in diesen Request angegebenen Konten.

Side Effects

Handelt es sich bei diesem Consent Request um einen Request, bei den der "recurringIndicator" auf "TRUE" steht, und wenn bereits ein früherer Consent für den wiederkehrenden Zugriff auf Kontoinformationen für den betroffenen PSU, der von diesen TPP übermittelt wurde, läuft der frühere Consent automatisch aus, sobald der neue Consent Request vom PSU genehmigt wird.

Für Consent Requests sind keine Side Effects vorgesehen, bei denen der " recurringIndicator " auf "false" steht.

Ablauf Parameter

Kein spezifischer Ablauf Parameter.

Request Header

Attribut	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Aufruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.
PSU-ID	String	K	Kann vorgeschrieben sein.

Request Body

Attribute	Typ	K	Beschreibung
access	Account Access	M	Angeforderte Kontenzugriffsdienste.
recurringIndicator	Boolean		True, falls Einwilligung für wiederkehrenden Kontozugriff. False, falls Einwilligung für einmaligen Kontenzugriff.
validUntil	ISODate	M	Dieser Parameter fordert ein *„gültig bis“* Datum für den angeforderten Consent an. Der Inhalt ist das lokale Datum der Bank im ISODate Format, z.B. 30.10.2018. Ein zukünftiges Datum wird möglicherweise von der Schnittstelle angepasst. Das Consentobjekt, das vom GET-Consent-Request abgerufen werden soll, welches das angepasste Datum enthält.
frequencyPerDay	Integer	M	Dieses Feld gibt die gewünschte max. Frequenz pro Tag für einen Zugriff ohne PSU Beteiligung an. Für einen einmaligen Zugriff wird dieses Attribut auf „1“ gesetzt. Diese Frequenz muss größer als 1 sein. Sofern bilateral zwischen TPP und der Schnittstelle nichts anderes vereinbart ist, ist die Frequenz kleiner als 4.

Hinweis: Alle in dieser Nachricht verwendeten zulässigen „Account“-Attribute („accounts“, „balances“ und „transactions“) müssen eine nicht leere Liste von Kontoreferenzen enthalten, die die Konten für für die Art des Zugriffs beantragt wird. Bitte beachten Sie, dass ein Zugriffsrecht auf „transactions“ oder „balances“ auch für den Zugriff die generischen Endpoints /Accounts ermöglicht, d. h. implizit auch den Zugriff auf „accounts“ unterstützt.

Zusätzlich unterstützt die Schnittstelle auch Consent Requests, bei denen das Unterattribut „availableAccounts“ mit dem Wert „allAccounts“ verwendet wird.

Response Code

Der HTTP Response Code ist 201.

Response Header

Attribute	Typ	K	Beschreibung
Location	String	M	Ort der erstellten Ressource.
X-Request-ID	UUID	M	ID des Requests, die für den Anruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.
ASPSP-SCA-Approach	String	K	Möglicher Wert ist EMBEDDED.

Response Body

Attribute	Typ	K	Beschreibung
consentStatus	Consent Status	M	Status des Consent.
consentId	String	M	Identifizierung der Consent Ressource, wie sie in der API-Struktur verwendet wird.
_links	Links	M	Art der in diesem Response zugelassenen Links „startAuthorisationWithPSUAuthentication": Der Link zur Endpoints-Autorisierung, an dem die Sub-Ressourcen-Autorisierung beim Hochladen der PSU-Identifikationsdaten generiert werden muss. "startAutorisierungMitPsuAuthentificatoin": Der Link zur Endpoints-Autorisierung, unter dem die Autorisierungs-Sub-Ressource beim Hochladen der PSU-Authentifizierungsdaten generiert wird. "Self": Der Link zur Ressource "Establish Account Information Consent", die durch diesen Request erstellt wurde. Über diesen Link können die Ressourcendaten abgerufen werden. "Status": Der Link, um den Transaktionsstatus der Payment Initiation abzurufen.
psuMessage	Max512Text	O	Text der dem PSU angezeigt wird.

Beispiel

Request

POST https://api.testbank.com/v1/consents
Content-Type:application/json
X-Request-ID:99391c7e-ad88-49ec-a2ad-99ddcb1f7756
PSU-IP-Address:192.168.8.78
PSU-User-Agent:Mozilla/5.0 (Windows NT 10.0; WOW64; rv:54.0) Gecko/20100101 Firefox/54.0
Date:Sun,
06Aug 2017 15:05:37GMT

```
{
  "access":{
    "balances":[
      {
        "iban":"DE40100100103307118608"
      },
      {
        "iban":"DE02100100109307118603"
      },
      {
        "iban":"DE67100100101306118605"
      }
    ]
  },
  "transactions":[
    {
      "iban":"DE40100100103307118608"
```

```
    }
  ],
  "recurringIndicator":true,
  "validUntil":"2017-11-01",
  "frequencyPerDay":"4"
}
```


Response


```
HTTP/1.x 201 Created
X-Request-ID:99391c7e-ad88-49ec-a2ad-99ddcb1f7721
ASPSP-SCA-Approach:EMBEDDED
Date:Sun,
06Aug 2017 15:05:47GMT
Location:"v1/consents/1234-wertiq-983"Content-Type:application/json{
  "consentStatus":"received",
  "consentId":"1234-wertiq-983",
  "_links":{
    "startAuthorisationWithPsuAuthentication":{
      "href":"/v1/consents/1234-wertiq-983/authorisations"
    }
  }
}
```


6.2.1.1 Consent Request auf der Kontenliste

Consent Request auf der Accountliste der verfügbaren Accounts

Diese Funktion wird durch den gleichen Aufruf wie der Consent Request für dedizierte Konten unterstützt. Der einzige Unterschied besteht darin, dass der Aufruf nur das Unterattribut "availableAccounts" innerhalb des Attributs "access" mit dem Wert "allAccounts" enthält.

In diesem Fall erstellt der Aufruf eine Consent Ressource, um eine Liste aller verfügbaren Konten bzw. aller verfügbaren Konten mit ihren Salden zurückzugeben. Die Einrichtung dieser Einwilligung (dieses Consents) erfordert kein SCA durch den PSU.

Beispiel Consent zur Account Liste der verfügbaren Accounts


```
POST https://api.testbank.com/v1/consents
Content-Type:application/json
X-Request-ID:99391c7e-ad88-49ec-a2ad-99ddcb1f7756
PSU-IP-Address:192.168.8.78
PSU-User-Agent:Mozilla/5.0 (Windows NT 10.0; WOW64; rv:54.0) Gecko/20100101 Firefox/54.0
Date:Sun,
06Aug 2017 15:05:37GMT{
  "access":{
    "availableAccounts":"allAccounts"
  },
  "recurringIndicator":false,
  "validUntil":"2017-08-06",
  "frequencyPerDay":"1"
}
```

6.2.2 Get Status Request

Aufruf

GET /v1/consents/{consentId}/status

Kann den Status einer Account Information Consent Ressource überprüfen.

Pfad Parameter

Attribute	Typ	Beschreibung
consentId	String	Die Identifikation des Consents, die der angelegten Ressource zugeordnet ist.

Request Header

Attribute	Typ	K	Beschreibung
X-Request-ID	UUID	M	Request-ID, die für den Aufruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.

Request Body

Kein Request Body.

Response Code

HTTP Response Code ist 200.

Response Header

Attribute	Typ	K	Beschreibung
X-Request-ID	UUID	M	Request-ID, die für den Aufruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.

Response Body

Attribute	Typ	K	Beschreibung
consentStatus	Consent Status	M	Dies ist der übergreifende Lebenszyklus-Status des Consents.

Beispiel

Request

GET *https://api.testbank.com/v1/consents/qwer3456tzui7890/status*
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
PSU-IP-Address: 192.168.8.78

PSU-User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:54.0) Gecko/20100101 Firefox/54.0
Date: Sun, 06 Aug 2017 15:05:46 GMT

Response
HTTP/1.x 200 Ok
X-Request-ID:99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date:Sun,
06Aug 2017 15:05:47GMT
Content-Type:application/json{
  "consentStatus":"valid"
}

6.2.3 Get Consent Request

Aufruf

GET /v1/consents/{consentId}

Gibt den Inhalt eines Account Information Consent Objects zurück.

Pfad Parameter

Attribute	Typ	Beschreibung
consentId	String	ID des entsprechenden Consentobjekts, wie sie von einem Account Information Consent Request zurückgegeben wird.

Abfrage Parameters

Keine spezifische Abfrage Parameter.

Request Header

Siehe Abschnitt 6.2.3.

Request Body

Kein Request Body.

Response Code

HTTP Response Code ist 200.

Response Header

Siehe Abschnitt 6.4.2.

Response Body

Attribut	Typ	K	Beschreibung
Access	Account Access	M	Angeforderte Kontenzugriffsdienste.
recurringIndicator	Boolean	M	True, falls Einwilligung für wiederkehrenden Kontozugriff. False, falls Einwilligung für einmaligen Kontenzugriff.
validUntil	ISODate	M	Dieser Parameter fordert ein *„gültig bis“* Datum für den angeforderten Consent an. Der Inhalt ist das lokale Datum der Bank im ISODate Format, z.B. 30.10.2018. Ein zukünftiges Datum wird möglicherweise von der Schnittstelle angepasst. Das Consentobjekt, das vom GET-Consent-Request abgerufen werden soll, welches das angepasste Datum enthält.
frequencyPerDay	Integer	M	Dieses Feld gibt die gewünschte max. Frequenz pro Tag für einen Zugriff ohne PSU Beteiligung an. Für einen einmaligen Zugriff wird dieses Attribut auf „1“ gesetzt. Diese Frequenz muss größer als 1 sein. Sofern bilateral zwischen TPP und der Schnittstelle nichts anderes vereinbart ist, ist die Frequenz kleiner als 4.
lastActionDate	ISODate	M	Dieses Datum enthält das Datum der letzten Aktion auf das Consent Object entweder über die XS2A-Schnittstelle oder die PSU/Schnittstelle mit Auswirkungen auf den Status.
consentStatus	Consent Status	M	Status der Consent Ressource.

Beispiel

Request
GET https://api.testbank.com/v1/consents/qwadfer3adfasd
Response

HTTP/1.x 200 Ok
X-Request-ID:99391c7e-ad88-4d34-355-23d
Date:Sun,
06Aug 2017 15:05:47GMT
Content-Type:application/json{


```
  "access":{
    "balances":[
      {
        "iban":"DE2310010010123456789"
      }
    ],
    "transactions":[
      {
        "iban":"DE2310010010123456789"
      }
    ],
    "recurringIndicator":true,
    "validUntil":"2017-11-01",
    "frequencyPerDay":"4",
    "consentStatus":"valid",
    "_links":{
      "account":{
        "href":"/v1/accounts"
      }
```

```
    }
  }
}
```

6.3 Löschen des Account Information Consent Objekts

Der TPP kann ein Account Information Consent Objekt mit folgenden Aufruf löschen.

Aufruf

DELETE /v1/consents/{consentId}

Löscht einen gegebenen Consent.

Pfad Parameter

Attribut	Typ	Beschreibung
consentId	String	Enthält die zu löschende Ressourcen-ID des Consents.

Abfrage Parameter

Keine spezifische Abfrage Parameter.

Request Header

Attribut	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, eindeutig zum Aufruf, wie vom initiierenden Teilnehmer festgelegt.

Request Body

Kein Request Body.

Response Code

HTTP Response Code ist 204.

Response Header

Attribut	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requets, eindeutig zum Aufruf, wie vom initiierenden Teilnehmer festgelegt.

Response Body

Kein Response Body.

Beispiel

Request
DELETE *https://api.testbank.com/v1/consents/qwer3456tzui7890*
X-Request-ID 99391c7e-ad88-49ec-a2ad-99ddcb1f7757
Date Sun, 13 Aug 2017 17:05:37 GMT

Response
HTTP/1.x 204 No Content
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7757
Date: Sun, 06 Aug 2017 15:05:47 GMT

6.4 Read Account Data Requests

6.4.1 Read Account List

Aufruf

GET / v1 / accounts

Liest eine Liste von Bankkonten, ggf. mit Salden. Es wird davon ausgegangen, dass ein Consent des PSUs zu diesem Zugriff bereits erteilt und auf im Banksystem gespeichert ist. Die adressierte Kontenliste hängt dann von der PSU-ID und dem gespeicherten Consent ab, der von der consentId adressiert wird.

Hinweis: Wenn der Consent nur erteilt wird, um die Liste der verfügbaren Konten anzuzeigen, ("availableAccounts" Zugriffsrechte, vgl. Abschnitt 6.2.1) werden keine Hyperlinks zu Salden oder Transaktion Endpoints geliefert.

Abfrage Parameters

Keine spezifischen Abfrage Parameter.

Request Header

Attribut	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Request eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.
Consent-ID	String	M	Enthält die zuvor über den "Establish Consent Transaction" Prozess erhaltene ID.

Request Body

Kein Request Body.

Response Code

HTTP Response Code ist 200.

Response Header

Attribute	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Aufruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.

Response Body

Attribute	Typ	K	Beschreibung
Accounts	Array der Account Details	M	

Beispiel

Response Body

Response im Falle eines Beispiels, bei dem der Consent zu zwei verschiedenen IBANs erteilt wurde.


```
{
  "accounts":[
    {
      "resourceId":"DE23100010010123456789",
      "iban":"DE23100010010123456789",
      "currency":"EUR",
      "ownerName":"Hans Mustermann"
      "product":"Girokonto",
      "name":"US Dollar Account",
      "_links":{
        "balances":{
          "href":"/v1/accounts/DE23100010010123456789/balances"
        },
        "transactions":{
          "href":"/v1/accounts/DE23100010010123456789/transactions"
        }
      }
    },
    {
      "resourceId":" DE23100010010123456789",
      "iban":"DE23100010010123456788",
      "currency":"USD",
      "ownerName":"Hans Mustermann"
      "product":"Fremdwährungskonto",
      "name":"US Dollar Account",
      "_links":{
        "balances":{
          "href":"/v1/accounts/ DE23100010010123456789/balances"
        }
      }
    }
```

```
  ]
}
```

6.4.2 Read Account Details

Aufruf

GET /v1/accounts/{account-id}

Liest Details zu einem Konto, ggf. mit Salden. Es wird davon ausgegangen, dass einen Consent der PSU zu diesem Zugriff bereits gegeben und im Schnittstellensystem gespeichert ist. Die adressierten Details dieses Kontos hängen dann von dem gespeicherten Consent ab, die von consentId bzw. dem OAuth2-Zugriffstoken adressiert wird.

Abfrage Parameter

Keine spezifischen Abfrage Parameter.

Request Header

Attribute	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Request eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.
Consent-ID	String	M	Enthält die zuvor über den "Establish Consent Transaction" Prozess erhaltene ID.

Request Body

Kein Request Body.

Response Code

HTTP Response Code ist 200.

Response Header

Attribute	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Request eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.

Response Body

Attribute	Typ	K	Beschreibung
account	Account Details	M	

Beispiel

Response Body für einen regulären Account


```
{
  "account":{
    "resourceId":" FR7612345678976501234567890l4",
    "iban":"FR7612345987650123456789014",
    "currency":"EUR",
    "ownerName":"Hans Mustermann"
    "product":"Girokonto",
    "name":"Main Account",
    "_links":{
      "balances":{
        "href":"/v1/accounts/ FR7612345678976501234567890l4/balances"
      },
      "transactions":{
        "href":"/v1/accounts/ FR7612345678976501234567890l4/transactions"
      }
    }
  }
}
```


6.4.3 Read Balance

Aufruf

GET /v1/accounts/{account-id}/balances

Liest Kontodaten von einem bestimmten Konto, das mit "account-id" adressiert ist.

Pfad Parameter

Attribut	Typ	Beschreibung
account-id	String	Diese Identifikation bezeichnet das adressierte Konto und wird über den Aufruf "Read Account List" abgerufen. Die account-id ist das Attribut "resourceId" der Kontenstruktur.

Abfrage Parameters

Keine spezifischen Abfrage Parameter.

Response Code

HTTP Response Code ist 200.

Request Header

Attribute	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Request eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.

Attribute	Typ	K	Beschreibung
Consent	String	M	

Request Body

Kein Request Body.

Response Header

Attribute	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Request eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.

Response Body

Attribute	Typ	K	Beschreibung
account	Account Reference	O	Kennung des adressierten Kontos.
balances	Array of Balance	M	Eine Liste der Salden zu diesem Konto, z.B. der aktuelle Saldo oder der zuletzt gebuchte Saldo.

Beispiel

Response Body

Response bei einem regulären Konto.


```
{
  "account":{
    "iban":"FR7612345987650123456789014"
  },
  "balances":[
    {
      "balanceType":"closingBooked",
      "balanceAmount":{
        "currency":"EUR",
        "amount":"500.00"
      },
      "referenceDate":"2017-10-26"
    },
    {
      "balanceType":"expected",
      "balanceAmount":{
        "currency":"EUR",
        "amount":"900.00"
      },
```

```
      "referenceDate":"2017-10-27"
    }
  ]
}
```


6.4.4 Read Transaction List

Aufruf

GET /v1/accounts/{account-id}/transactions {query-parameters}

Liest Kontodaten von einem bestimmten Konto, das mit der "account-id" angesprochen wird.

Pfad Parameter

Attribute	Typ	K	Beschreibung
Account-ID	String	M	Diese Identifikation bezeichnet das adressierte Konto. Die Account-ID wird über den Aufruf "Read Account List" abgerufen. Die Account-ID ist das Attribut "resourcenId" der Account-Struktur.

Abfrage Parameter

Attribute	Typ	K	Beschreibung
dateFrom	ISODate	M	Anfangsdatum (einschließlich des Datums dateFrom) der Transaktionsliste. Bei gebuchten Transaktionen ist das Buchungsdatum relevant. Für anstehende Transaktionen ist das maßgebliche Datum das Eintrittsdatum, das weder in dieser Schnittstelle noch in anderen Kanälen der Bank transparent sein darf.
dateTo	ISODate	O	Enddatum (inklusive Datum dateTo) der Transaktionsliste, Standard ist "jetzt", wenn nicht angegeben. Für gebuchte Transaktionen ist das relevante Datum das Buchungsdatum. Für ausstehende Transaktionen ist das relevante Datum das Eintrittsdatum, das weder in dieser API noch in anderen Kanälen der Schnittstelle transparent sein kann.
bookingStatus	String	M	Erlaubte Codes sind "booked“, „pending“ und „both“ für die Abfrage der Kontoumsätze und „information“ für die Abfrage des Dauerauftragsbestand.

Request Header

Attribute	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID der Requests, die für den Request eindeutig ist, wie vom

Attribute	Typ	K	Beschreibung
			initiierenden Teilnehmer festgelegt.
PSU-IP-Address	String	C	Das weitergeleitete IP-Adresse Headerfeld besteht aus dem entsprechenden HTTP-Anforderungs-IP-Adressfeld zwischen PSU und TPP. Sie darf nur dann enthalten sein, wenn diese Anforderung vom PSU aktiv ausgelöst wurde.
Consent-ID	String	M	Enthält die zuvor über den "Establish Consent Transaction" Prozess erhaltene ID.

Request Body

Kein Request Body.

Response Code

HTTP Response Code ist 200.

Response Header

Attribut	Typ	K	Beschreibung
Content-Type	String	M	Mögliche Werte sind: • application/json
X-Request-ID	UUID	M	ID des Requests, die für den Aufruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.

Hinweis: Der ASPSP kann Standardkompressionsverfahren auf dem Application Level für die Antwortnachricht verwenden, wie im Header der Inhaltskodierung angegeben.

Response Body

Der JSON Response ist folgendermaßen definiert:

Attribut	Typ	K	Beschreibung
account	Account Reference	O	Kennung des adressierten Kontos.
transactions	Account Report	O	JSON-basierter Kontoauszug. Dieser Account Report enthält Transaktionen von den Abfrage Parametern. Dieser Kontenbericht enthält Transaktionen, die sich aus den Abfrageparametern ergeben." Des weiteren können hier, bei Abfrage der Daueraufträge, die Dauerauftragsdetails aufgeführt sein.

Request

GET *https://api.testbank.com/v1/accounts/qwer3456tzui7890/transactions?dateFrom=2017-07-01&dateTo=2017-07-30*

Accept:application/json

Response

Response im JSON-Format für einen Zugriff auf ein reguläres Konto:

HTTP/1.x 200 Ok
X-Request-ID:99391c7e-ad88-49ec-a2ad-99ddcb1f7757
Date:Sun,
06Aug 2017 15:05:47GMT
Content-Type:application/json{


```
  “account”:{
    "iban":"DE2310010010123456788"
  },
  "transactions":{
    "booked":[
      {
        "transactionId":"1234567",
        "creditorName":"John Miles",
        "creditorAccount":{
          "iban":"DE67100100101306118605"
        },
        "transactionAmount":{
          "currency":"EUR",
          "amount":"256.67"
        },
        "bookingDate":"2017-10-25",
        "valueDate":"2017-10-26",
        "remittanceInformationUnstructured":"Example 1"
      },
      {
        "transactionId":"1234568",
        "debtorName":"Paul Simpson",
        "debtorAccount":{
          "iban":"NL76RABO0359400371"
        },
        "transactionAmount":{
          "currency":"EUR",
          "amount":"343.01"
        },
        "bookingDate":"2017-10-25",
        "valueDate":"2017-10-26",
        "remittanceInformationUnstructured":"Example 2"
      }
    ],
    "pending":[
      {
        "transactionId":"1234569",
        "creditorName":"Claude Renault",
        "creditorAccount":{
          "iban":"FR7612345987650123456789014"
        },
        "transactionAmount":{
          "currency":"EUR",
          "amount":"-100.03"
        },
        "valueDate":"2017-10-26",
        "remittanceInformationUnstructured":"Example 3"
      }
```

```
    ],
    "_links":{
      "account":{
        "href":"/v1/accounts/ FR7612345987650123456789014"
      }
    }
  }
}
```


Response im Fall von Daueraufträgen:


```
{
  "account": {"iban": "DE11201201001111111117"},
  "transactions":   {
    "information":      [
              {
          "transactionAmount":          {
            "currency": "EUR",
            "amount": "123.00"
          },
          "creditorName": "Hans Mustermann",
          "creditorAccount": {"iban": "AT116000000072627442"},
          "bankTransactionCode": "PMNT-ICDT-STDO",
          "additionalInformationStructured": {"standingOrderDetails":          {
            "startDate": "2016-09-28",
            "frequency": "monthly",
            "dayOfExecution": "28"
          }}
        }
    ],
    "_links": {"account": {"href": "/v1/accounts/DE11201201001111111117"}}
  }
}
```


7 Gemeinsam verwendete Prozesse in AIS und PIS Services

Die Prozesse zum Starten von Autorisierungen, zum Aktualisieren der PSU-Identifikation oder der PSU-Authentifizierungsdaten und zur expliziten Autorisierung von Transaktionen mit SCA sind bei PIS- und AIS-Services sehr ähnlich. Die API-Aufrufe, die diese Prozesse unterstützen, werden im Folgenden unabhängig vom betroffenen Service/Endpoint beschrieben. Aus Gründen der Übersichtlichkeit werden die Endpoints seperat für den Payment Initiation Service und den Account Information Service separat definiert. Diese Prozesse werden in der Regel über einem Hyperlink der Schnittstelle verwendet. Die Verwendung wird zu Beginn der folgenden Abschnitte definiert.

7.1 Start Authorisation Process

Verwendung

Der „Start Authorisation Process“ ist ein Prozess, der für die Erstellung einer neuen Autorisierungs-Sub-Ressource erforderlich ist.

Die Schnittstelle hat in dem vorangegangenen Response mit einem Hyperlink "startAuthorisation" darauf hingewiesen, dass ein expliziter Start des Autorisierungsprozesses erforderlich ist. Der Hyperlink wird in der erweiterten Form "startAuthorisationWithAuthentciation" verwendet.

Aufruf im Rahmen eines Payment Initiation Request

POST /v1/{payment-service}/{payment-product}/{paymentId}/authorisations

Startet den Autorisierungsprozess für eine Payment Initiation.

Aufruf im Rahmen einer Account Information Consent Request

POST /v1/consents/{consentId}/authorisations

Startet den Autorisierungsprozess um den Account Information Consent auf dem Server einzurichten.

Pfad Parameter

Attribut	Typ	Beschreibung
payment-service	String	Die möglichen Werte sind "payments".
payment-product	String	Das Zahlungsprodukt, unter dem die Zahlung ausgelöst wurde.
paymentId oder consentId		Ressourcenidentifikation der damit verbundenen Payment Initiation oder Consentsressource.

Abfrage Parameter

Keine spezifischen Abfrage Parameter.

Request Header

Attribut	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Aufruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.
PSU-ID	String	O	Benutzerkennung des PSU in der FinTS Banking Schnittstelle. Dieses Attribut muss gesendet werden, wenn dieses Attribut noch nicht vorher übertragen wurde.

Request Body

Kein Request Body.

Hinweis: Wenn die Hyperlinks in den folgenden erweiterten Formen in der Responsenachricht verwendet wurden, gelten weitere Bedingungen für Body Parameter des Requests, wie im Folgenden beschrieben:

- "startAuthorisationWithPsuAuthentication": vgl. Abschnitt 7.1.

Die Unterschiede in den Aufrufen bestehen dann nur darin, ob man einen POST-Befehl verwendet, um die Sub-Ressourcen-Autorisierung zu erstellen und gleichzeitig die angegebenen Daten zu aktualisieren, oder ob man mit einem PUT-Befehl nur die angegebenen Daten in einer bereits erstellten Sub-Ressource aktualisiert.

Response Code

Der HTTP-Responsecode ist 201.

Response Header

Attribut	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Aufruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.
ASPSP-SCA-Approach	String	K	Einzig Möglicher wert ist EMBEDDED.

Response Body

Attribut	Typ	K	Beschreibung
scaStatus	SCA Status	M	
authorisationId	String	M	Eindeutige Ressourcenidentifikation der erstellten Autorisierungs-Sub-Ressource.
scaMethods	Array der Authenticatio n Objekte	K	Dieses Datenelement kann enthalten sein, wenn der PSU die Wahl zwischen verschiedenen Authentifizierungsmethoden hat. Ist dieses Datenelement enthalten, so ist auch ein Hyperlink vom Typ "selectAuthenticationMethod" im Response Body enthalten. Diese Methoden sind dem PSU zur Auswahl vorzulegen.
chosenSca Method	Authenticatio n Objekt	K	Dieses Datenelement ist nur dann in der Response enthalten, die Authentifizierungsmethode implizit ausgewählt ist.
challengeData	Challenge	K	Die Challenge ist zusätzlich zum Datenelement "chosenScaMethod" enthalten, wenn für SCA Challenge-Daten benötigt werden.
_links			Liste von Hyperlinks. **Anmerkung**: Alle Links sind Links, die von der Schnittstelle festgelegt werden. Art der in dieser Response zugelassenen Links:

Attribut	Typ	K	Beschreibung
			"selectAuthenticationMethod": Der Link zur Sub-Ressourcen-Autorisierung, wobei die gewählte Authentifizierungsmethode hochgeladen werden muss. Dieser Link ist unter den genau gleichen Bedingungen wie das Datenelement "scaMethods“ enthalten. "authoriseTransaction": Der Link zur Sub-Ressourcen-Autorisierung, in der die Autorisierungsdaten hochgeladen werden müssen, z.B. das per SMS empfangene OTP. "scaStatus": Der Link zum Abrufen des scaStatus der entsprechenden Autorisierungs-Sub-Ressource.
psuMessage	Max512Text	O	

7.1.1 Update PSU Data (Authentification) in dem Embedded Ansatz

Dieser Aufruf wird verwendet, wenn im vorhergehenden Aufruf der Hyperlink vom Typ "updatePsuAuthentication" in der Antwort enthalten war und von dem TPP übernommen wird.

Aufruf im Rahmen einer Payment Initiierung

PUT /v1/{payment-service}/{payment-product}/{paymentId}/authorisations/{authorisationId}

Aktualisiert die Subressourcendaten der Payment Initiation auf dem Server durch PSU-Anmeldeinformationen, wenn diese von der ASPSP angefordert werden.

Aufruf im Kontext eines Account Information Consent Requests

PUT /v1/consents/{consentId}/authorisations/{authorisationId}

Aktualisiert die Account Information Consent Authorisation Sub-Ressourcendaten auf dem Server durch PSU-Anmeldeinformationen, falls von der ASPSP angefordert.

Path Parameters

Attribut	Typ	Beschreibung
payment-service	String	Einzig möglicher Wert ist „Payments“.
payment-product	String	Das Zahlungsprodukt, unter dem die Zahlung unter paymentId ausgelöst wurde. Es wird von der ASPSP überprüft, ob das Zahlungsprodukt mit der von paymentId adressierten Payment Initiierung übereinstimmt. Ressourcenidentifikation der damit verbundenen Payment Initiation, oder Consentsressource.
paymentId oder consentId	String	Ressourcenidentifikation der zugehörigen Payment Initiierung oder Consent.
authorisationId	String	Ressourcenidentifikation der zugehörigen Payment Initiierung oder Consent.

Abfrage Parameters

Keine spezifischen Query Parameter.

Request Header

Attribut	Typ	Kondition	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Aufruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.
PSU-ID	String	K	Enthalten, wenn sie noch nicht in einem vorangegangenen Request enthalten sind, und von der ASPSP in der zugehörigen Response beauftragt.

Request Body

Attribut	Typ	Kondition	Beschreibung
psuData	PSU Data	M	

Response Code

HTTP Response Code ist 200.

Response Header

Attribut	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Aufruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.
ASPSP-SCA-Approach	String	K	Einzig möglicher Wert ist EMBEDDED.

Response Body

Attribut	Typ	Kondition	Beschreibung
chosenSca Method	Authentication Objekt	K	Eine Definition der bereitgestellten SCA-Methode ist enthalten, wenn nur eine Authentifizierungsmethode verfügbar ist.
challengeData	Challenge	K	Challenge-Daten können enthalten sein, wenn nur eine Authentifizierungsmethode verfügbar ist.
scaMethods	Array der Authentifizierung	K	Kann enthalten sein, wenn mehrere Authentifizierungsmethoden verfügbar sind (Name, Typ).
_links	Links	K	Eine Liste von Hyperlinks, die vom TPP zu erkennen sind. Kann enthalten sein, wenn mehrere Authentifizierungsmethoden für den PSU verfügbar sind.

Attribut	Typ	Kondition	Beschreibung
			Art der in dieser Response zugelassenen Links: "selectAuthenticationMethod": Dies ist ein Link zu einer Ressource, bei dem das TPP die anwendbaren Authentifizierungsmethoden des zweiten Faktors den das PSU auswählen kann, wenn es mehrere verfügbare Authentifizierungsmethoden gibt. Dieser Link ist nur dann enthalten wenn der PSU die Wahl zwischen verschiedenen Authentifizierungsmethoden hat. Wenn dieser Link enthalten ist, dann ist auch das Datenelement "scaMethods" im Response Body enthalten. "authoriseTransaction": Der Link zu der Ressource, an die der "Transaction Authorisation Request" gesendet wird. Dies ist der Link zu der Ressource, die die Transaktion autorisiert, indem sie die SCA-Authentifizierungsdaten überprüft. "scaStatus": Der Link zum Abrufen des scaStatus der entsprechenden Autorisierungs-Sub-Ressource.
scaStatus	SCA Status	M	
psuMessage	MaxText512	O	

Hinweis: Im Falle eines falschen Passworts muss der TPP die PSU auffordern, das Passwort erneut einzugeben. Das neu eingegebene Passwort muss auf den gleichen Pfad aktualisiert werden.

Beispiel

Request im Falle eines Embedded Ansatzes

PUT *https://api.testbank.com/v1/payments/sepa-credit-transfers/qwer3456tzui7890/authorisations/123auth456*

```
X-Request-ID:99391c7e-ad88-49ec-a2ad-99ddcb1f7721
PSU-ID:PSU-1234{
  "psuData":{
    "password":"start12"
  }
}
```

Response im Falle eines Embedded Ansatzes

```
HTTP/1.x 200 OK
X-Request-ID:99391c7e-ad88-49ec-a2ad-99ddcb1f7721
ASPSP-SCA-Ansatz:EMBEDDED
Date:Sun,
06Aug 2017 15:05:47GMT
```

```
Content-Type:application/json{
  "scaStatus":"psuAuthenticated",
  "_links":{
    "authoriseTransaction":{
      "href":"/v1/payments/sepa-credit-transfers/1234-wertiq-983/authorisations/123auth456"
    }
  }
```

7.2 PSU Daten (Authentifizierungsmethode auswählen)

Dieser Aufruf wird verwendet, wenn im vorhergehenden Aufruf der Hyperlink vom Typ "selectAuthenticationMethod" in dem Response enthalten war.

Aufruf im Kontext des Payment Initiation Request

PUT /v1/{payment-service}/{payment-product}/{paymentId}/authorisations/{authorisationId}

Aktualisiert die Sub-Ressourcendaten der Payment Initiation auf dem Server mit der Authentifizierungsmethode.

Aufruf im Kontext des Account Information Consent Request

PUT /v1/consents/{consentId}/authorisations/{authorisationId}

Aktualisiert die Autorisierungsdaten des Accounts Information Consents auf dem Server mit der Authentifizierungsmethode.

Path Parameters

Attribut	Typ	Beschreibung
payment-service	String	Der einzig mögliche Wert ist "payments" .
payment-product	String	Das Zahlungsprodukt, unter dem die Zahlung unter paymentId ausgelöst wurde.
paymentId oder consentId	String	Ressourcenidentifikation der zugehörigen Payment Initiation oder Consent Ressource.
authorisationId	String	Ressourcenidentifikation der zugehörigen Payment Initiation oder Consent Autorisierungs Sub-Ressource.

Query Parameters

Keine spezifischen Abfrage Parameters.

Response Code

The HTTP Response Code ist 200.

Request Header

Attribute	Typ	Kondition	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Aufruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.

Request Body

Attribute	Typ	K	Beschreibung
Authentification Method	String	M	Die Authentifizierungsmethoden-ID wird von der Schnittstelle angeboten.

Response Code

HTTP Response Code ist 200.

Response Header

Attribut	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Aufruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.
ASPSP-SCA-Approach	String		Einzig möglicher Wert ist EMBEDDED

Response Body

Attribut	Typ	K	Beschreibung
chosenScaMethod	Authentication object	K	Eine Definition der bereitgestellten SCA-Methode.
challengeData	Challenge	K	
_links	Links	K	Eine Liste von Hyperlinks. **Anmerkung**: Alle Links sind relative oder vollständige Links sein, die von der ASPSP festgelegt werden. Art der in dieser Response zugelassenen Links: "authoriseTransaction": Der Link zur Sub-Ressource Autorisierung, in der die Autorisierungsdaten hochgeladen werden müssen, z.B. das per SMS empfangene TOP. "scaStatus": Der Link zum Abrufen des scaStatus der entsprechenden Autorisierungs-Sub-Ressource.
scaStatus	SCA Status	M	
psuMessage	MaxText512	O	

Beispiel

Request im Falle einer Embedded Approach

PUT *https://api.testbank.com/v1/payments/sepa-credit-transfers/qwer3456tzui7890/authorisations/123auth456*
X-Request-ID:asdfoeljkasdfoelkjasdf-123479093{
  authenticationMethodId:"myAuthenticationID"
}

Response im Falle einer Embedded Approach

HTTP/1.x 200 OK
X-Request-ID:99391c7e-ad88-49ec-a2ad-99ddcb1f7721
ASPSP-SCA-Ansatz:EMBEDDED
Date:Sun,
06Aug 2017 15:05:47GMT
Content-Type:application/json{
  "scaStatus":"scaMethodSelected",
  "chosenScaMethod":{
    "authenticationType":"SMS_OTP",
    "authenticationMethodId":"myAuthenticationID"
  },
  "challengeData":{
    "otpMaxLength":"6",
    "otpFormat":"integer"
  },
  "_links":{

```
    "authoriseTransaction":{
      "href":"/v1/payments/sepa-credit-transfers/1234-wertiq-983/authorisations/123auth456"
    }
  }
}
```

7.3 Autorisierung der Transaktion

Aufruf im Kontext eines Payment Initiation Requests

PUT /v1/payments/{payment-product}/{paymentId}/authorisations/{authorisationId}

Überträgt Antworten zur Challenge für die SCA-Prüfungen durch die Schnittstelle.

Aufruf im Kontext eines Account Information Consent Requests

PUT /v1/consents/{consentId}/authorisation/{authorisationId}

Überträgt Antworten zur Challenge für die SCA-Prüfungen durch die Schnittstelle.

Pfad Parameter

Attribute	Typ	Beschreibung
payment - product	String	Das zugehörige Zahlungsprodukt der zu autorisierenden Payment Initiation.
paymentId oder consentId	String	Ressourcenidentifikation der damit verbundenen Payment Initiation oder des Consents.
authorisationId	String	Ressourcenidentifikation der Autorisierungs-Sub-Ressource für Payment Initiation oder Consent.

Abfrage Parameter

Keine spezifische Abfrage Parameter.

Request Header

Attribute	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Aufruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.

Request Body

Attribute	Typ	K	Beschreibung
scaAuthentificationData	String	M	SCA-Authentifizierungsdaten, abhängig von der gewählten Authentifizierungsmethode. Wenn die Daten binär sind, dann sind sie base64zu kodieren.

Response Code

HTTP Response Code 200.

Response Header

Attribute	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Aufruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.

Response Body

Attribute	Typ	K	Beschreibung
scaStatus	SCA Status	M	

Hinweis: Bei fehlerhaften scaAuthenticationData muss der TPP den PSU auffordern, die Authentifizierungsdaten erneut einzugeben. Hierzu muss der TPP einen Restart des vollständigen Autorisierungsprozesses durch Generierung einer neuen Autorisierungs-Sub-Ressource durchführen. Die Schnittstelle informiert den TPP darüber, indem sie in den Fehlerinformationen einen zusätzlichen Abschnitt _links hinzufügt und einen entsprechenden Hyperlink präsentiert.

Beispiel

Request
PUT *https://api.testbank.com/v1/payments/sepa-credit-transfers/qwer3456tzui7890/authorisations/123auth456*
X-Request-ID:99391c7e-ad88-49ec-a2ad-99ddcb1f7721{
  "scaAuthenticationData":"123456"
}Response im Falle eines Embedded Ansatzes

Response Code 200
Response Body{
  "scaStatus":"finalised",
  "_links":{
    "scaStatus":{
      "href":"/v1/payments/sepa-credit-transfers/qwer3456tzui7890/authorisations/123auth456"
    }
  }
}

7.4 Get SCA Status Request

Aufruf im Kontext eines Payment Initiation Requests

GET /v1/{payment-service}/{payment-product}/{paymentId}/authorisations/{authorisationId}

Überprüft den SCA-Status einer Autorisierungs-Sub-Ressource.

Aufruf im Kontext eines Account Information Consent Requests

GET /v1/consents/{consentId}/authorisations/{authorisationId}

Überprüft den SCA-Status einer Autorisierungs-Sub-Ressource.

Pfad Parameter

Attribut	Typ	Beschreibung
payment-service	String	Der einzig mögliche Wert ist "payments“.
payment-product	String	Das Zahlungsprodukt, unter dem die Zahlung identifiziert durch die paymentId ausgelöst wurde.
paymentId oder consentId	String	Ressourcenidentifikation der damit verbundenen Payment Initiation oder Consent-Ressource.
authorisationId	String	Ressourcenidentifikation der zugehörigen Payment Initiation oder Consent-Ressource.

Request Header

Attribut	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Aufruf eindeutig ist, wie von der initiierenden Partei festgelegt.

Abfrage Parameters

Keine spezifische Abfrage Parameter definiert.

Request Body

Kein Request Body.

Response Code

The HTTP Response Code ist 200.

Response Header

Attribut	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Aufruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.
scaStatus	SCA Status	M	Dieses Datenelement enthält Informationen über den Status der angewandten SCA-Methode.

Beispiel

Request

GET *https://api.testbank.com/v1/payments/sepa-credit-transfers/1234-wertiq-983/authorisations/123auth456*
Accept: application/json
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date: Sun, 06 Aug 2017 15:04:07 GMT

Response
HTTP/1.x 200 Ok
X-Request-ID:99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date:Sun,
06Aug 2017 15:04:08GMT
Content-Type:application/json{
"scaStatus":"finalised"
}

8 Confirmation of Funds Service

8.1 Confirmation of Funds Request

Aufruf

POST /v1/Funds-Confirmations

Erstellt eine Bonitätsanfrage (Confirmation of Funds Request) bei der Schnittstelle an.

Abfrage Parameter

Keine spezifischen Abfrage Parameter.

Request Header

Attribut	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Aufruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.
Digest	vgl. Abschnitt 11	K	Nur enthalten, Element "Signature" im Header der Request enthalten ist.
Signature	vgl. Abschnitt 11	K	Eine Signatur des Requests durch den TPP auf Application Level.

Attribut	Typ	K	Beschreibung
TPP-Signature-Certificate	String	K	Das Zertifikat, das zum Signieren der Request verwendet wird, in base64-Codierung.
Consent-ID	String	K	Angegeben, wenn die Zustimmung der PSU im Rahmen des Consent-Verfahrens erteilt wurde.

Request Body

Attribut	Typ	K	Beschreibung
cardNumber	String	O	Kartennummer der vom PIISP ausgegebenen Karte. Sollte eingestellt werden, wenn verfügbar.
account	Account Reference	M	Die Kontonummer des PSU.
payee	Max70Text	O	Der Händler, bei dem die Karte akzeptiert wird als Information PSU.
instructedAmount	Amount	M	Transaktionsbetrag, der innerhalb des Bonitäts-Prüfmechanismus zu überprüfen ist.

Response Code

The HTTP Response Code ist 200.

Response Header

Attribut	Typ	K	Beschreibung
X-Request-ID	UUID	M	ID des Requests, die für den Aufruf eindeutig ist, wie vom initiierenden Teilnehmer festgelegt.
fundsAvailable	Boolean	M	Ist true, wenn zum Zeitpunkt des Requests genügend Geld vorhanden ist, ansonsten false.

Example

POST *https://api.testbank.com/v1/Funds-Confirmations*
Content-Type:application/json
X-Request-ID:99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date:Sun,
06Aug 2017 15:02:37GMT{

```
  "cardNumber":"12345678901234",
  "account":{
    "iban":"DE23100120020123456789"
  },
  "instructedAmount":{
    "currency":"EUR",
    "amount":"123"
  }
}
```

Response Body

```
      {"fundsAvailable": true}
```

9 Zahlungsverkehr Datenstrukturen

9.1 Einzelaufträge

Die folgende Tabelle gibt einen Überblick über die von dieser Schnittstelle unterstützten Aufträgeformate.

Datenelement	Typ	SCT SEPA-Überweisung
endToEnd Identification	Max35Text	O
debtorAccount (inkl. Typ)	Account Reference	M
debtorId	Max35Text	n.a.
ultimateDebtor	Max70Text	n.a.
instructedAmount (Inkl. Währung.)	Amount	M
transactionCurrency	Currency Code	n.a.
creditorAccount	Account Reference	M
creditorAgent	BICFI	n.a.
creditorAgentName	Max70Text	n.a.
creditorName	Max70Text	M
creditorId	Max35Text	n.a.
creditorAddress	Address	n.a.
ultimateCreditor	Max70Text	n.a.
purposeCode	Purpose Code	n.a.
chargeBearer	Charge Bearer	n.a.
remittance Information Unstructured	Max140Text	O
remittance Information Unstructured Array	Array of Max140Text	n.a.
remittance Information Structured	Remittance	n.a.
requestedExecution Date	ISODate	n.a.
requestedExecution Time	ISODateTime	n.a.

Die mit "n.a." gekennzeichneten Datenelemente werden nicht in den adressierten Coreservices verwendet, die von dieser Schnittstelle angeboten werden.

Anmerkung: Der Deptor Account ist ein Pflichtfeld für ein Einzelauftrag.

Anmerkung: Die Schnittstelle kann ein Payment Request ablehnen, wenn zusätzliche Datenelemente verwendet werden, die nicht angegeben sind.

9.2 Signaturen

Diese Schnittstelle verlangt, dass der TPP, wie in (signHTTP, 2019), in Kapitel 4 definiert eine digitale Signatur in seinem HTTP Request sendet. Die Signatur muss die folgenden Anforderungen zusätzlich zu (signHTTP, 2019), Kapitel 4 erfüllen.

Anmerkung: Im Falle einer mehrteiligen Nachricht wird das gleiche Verfahren zur Berechnung des Digests verwendet. D.h. ein Hash des (gesamten) Message Body wird berechnet, der alle Teile der mehrteiligen Nachricht sowie die Separatoren beinhaltet.

9.3 „Digest“ Header Mandatory

Wenn eine TPP eine Signatur gemäß (signHTTP, 2019), in Kapitel 4 definiert enthält, muss sie auch einen "Digest"-Header enthalten. Der Header "Digest" enthält einen Hash des Nachrichten-Bodys. Wenn die Nachricht keinen Body enthält, muss der Header "Digest" den Hash einer leeren Byteliste enthalten. Die einzigen Hash-Algorithmen, die zur Berechnung des Digest im Rahmen dieser Spezifikation verwendet werden können, sind SHA-256 und SHA-512.

10 Anforderungen an den „Signature“ Header

Wie in (signHTTP, 2019), Kapitel 4 definiert, muss ein "Signature"-Header vorhanden sein. Die Struktur eines "Signature"-Headers ist in (signHTTP, 2019), Kapitel 4.1 definiert, die folgende Tabelle listet die Anforderungen an den "Signature"-Header gemäß (signHTTP, 2019) und weitere spezifische Anforderungen auf.

Anmerkung: Im Falle einer mehrteiligen Nachricht wird das gleiche Verfahren zur Berechnung des Digests verwendet. D.h. ein Hash des (gesamten) Nachrichtenkörpers wird berechnet, der alle Teile der mehrteiligen Nachricht sowie die Separatoren beinhaltet.

Element	Typ	K	Anforderung	Ergänzende Anforderungen
keyId	String	M	Das Feld keyId ist eine Zeichenkette, die der Server verwenden kann, um die Komponente auszuwählen, die er zur Überprüfung der Signatur benötigt.	Seriennummer des Zertifikats des TPP, das im "TPP-Signaturzertifikat" Header dieses Requests enthalten ist. Es ist wie folgt zu formatieren: keyId="SN=XXX,CA=YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY". wobei "XXX" die Seriennummer des Zertifikats in hexadezimaler Codierung ist, die im TPP-Signature-Certificate-Header angegeben ist, und

Element	Typ	K	Anforderung	Ergänzende Anforderungen
				"YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY" der vollständige Name der Zertifizierungsstelle ist, die dieses Zertifikat erstellt hat.
algorithm	String	M	Der Parameter "Algorithmus" wird verwendet, um den digitalen Signaturalgorithmus anzugeben, der bei der Erzeugung der Signatur verwendet werden soll. Gültige Werte für diesen Parameter sind in der Registrierung für Signature Algorithms unter http://www.iana.org/assignments/signature-algorithms zu finden und DÜRFEN NICHT als "veraltet" markiert sein. Es wird bevorzugt, dass der von einer Implementierung verwendete Algorithmus aus den Schlüsselmetadaten abgeleitet wird, die durch die "keyId" identifiziert werden, anstatt aus diesem Feld. Der Parameter 'Algorithmus' [...] wird höchstwahrscheinlich in Zukunft veraltet sein.	Obligatorisch Der Algorithmus muss den gleichen Algorithmus für die Signatur identifizieren, wie er im Zertifikat (Element "TPP-Signatur-Zertifikat") dieses Requests angegeben ist. Der Algorithmus muss SHA-256 oder SHA-512 als Hash-Algorithmus identifizieren.
headers	String	M	Mit dem Parameter "headers" wird die Liste der HTTP-Header angegeben, die beim Generieren der Signatur für die Nachricht enthalten sind. Wenn angegeben, muss es sich um eine kleinegeschriebene, quotierte Liste von HTTP-Header Feldern in Anführungszeichen handeln, die jeweils durch ein Leerzeichen getrennt sind. Wenn nicht angegeben, MÜSSEN Implementierungen so erfolgen, als ob das Feld mit einem einzelnen	Obligatorisch Muss enthalten: • "digest", • "x-request-id", • "psu-id" (genau dann wenn "psu-id" als Header des HTTP-Request enthalten ist). Es dürfen keine anderen Einträge enthalten sein.

Element	Typ	K	Anforderung	Ergänzende Anforderungen
			Wert, dem `Date`-Header, in der Liste der HTTP-Header angegeben ist. Es ist zu beachten, dass die Listenreihenfolge wichtig ist und in der Reihenfolge angegeben werden MUSS, in der die Feldwertpaare des HTTP-Headers beim Signieren miteinander verkettet werden.	
signature	String	M	Der Parameter "signature" ist eine Base 64 kodierte digitale Signatur, wie in RFC 4648 (RFC4648, 2006), Abschnitt 4 beschrieben. Der Client verwendet die Signaturparameter `algorithm` und `Header`, um eine kanonisierte `Signaturkette` zu bilden. Diese Zeichenkette wird dann mit dem Schlüssel, der mit `keyId` verknüpft ist, und dem Algorithmus, der mit `algorithm` übereinstimmt, signiert. Der Parameter `signature` wird dann auf die Base64 Codierung der Signatur gesetzt.	[Keine weiteren Anforderungen]

Beispiel

Angenommen, ein TPP muss eine Signatur in den folgenden Request einfügen:

POST
https://api.testbank.com/v1/payments/sepa-credit-transfers/Content-Type:application/json
X-Request-ID:99391c7e-ad88-49ec-a2ad-99ddcb1f7721
PSU-IP-Address:192.168.8.78
PSU-ID:PSU-1234
PSU-User-Agent:Mozilla/5.0 (Windows NT 10.0; WOW64; rv:54.0) Gecko/20100101 Firefox/54.0{

```
  "instructedAmount":{
    "currency":"EUR",
    "amount":"123"
  },
  "debtorAccount":{
    "iban":"DE2310010010123456789"
  },
  "creditor":{
    "name":"Merchant123"
  },
  "creditorAccount":{
    "iban":"DE23100120020123456789"
```

```
  },
  "remittanceInformationUnstructured":"Ref Number Merchant"
}
```


So würde der Body als folgender String in Base64 kodiert:

eyAgICANCiAgICJpbnN0cnVjdGVkQW1vdW50IjogeyJjdXJyZW5jeSI6ICJFVVIiLCAiYW1vdW50Ij
ogIjEyMyJ9LA0KICAgImRlYnRvckFjY291bnQiOiB7ImliYW4iOiAiREUyMzEwMDEwMDEwMTIzN
DU2Nzg5In0sDQogICAiY3JlZGl0b3IiOiB7Im5hbWUiOiAiTWVyY2hhbnQxMjMifSwNCiAgICJjcmV
kaXRvckFjY291bnQiOiB7ImliYW4iOiAiREUyMzEwMDEyMDAyMDEyMzQ1Njc4OSJ9LA0KICAgI
nJlbWl0dGFuY2VJbmZvcm1hdGlvblVuc3RydWN0dXJlZCI6ICJSZWYgTnVtYmVyIE1lcmNoYW5
0Ig0KfQ==

und SHA-256 vom Request Body wäre

F9li3V7yu8S/QKVOhWiiqJBhGMVId8UGZ4sBRVPkok=in Base64
('17D962DD5EF2BBC4BF40A54E8568A28AA24184631521DF14199E2C05154F9289'
in hexadecimaler Darstellung).

Unter Verwendung des Signaturalgorithmus rsa-sha256 sieht der signierte Request wie folgt aus:


```
POST https://api.testbank.com/v1/payments/sepa-credit-transfers
Content-Type:application/json
X-Request-ID:99391c7e-ad88-49ec-a2ad-99ddcb1f7721
PSU-IP-Address:192.168.8.78
PSU-ID:PSU-1234
PSU-User-Agent:Mozilla/5.0 (Windows NT 10.0; WOW64; rv:54.0) Gecko/20100101 Firefox/54.0
code_Cchallenge_Mmethod="S256"
Digest:SHA-256=ZuYiOtZkVxhjWmwTO5lOpsPevUNMezvk6dfb6fVhebM=
Signature:keyId="SN=9FA1,
CA=D-TRUST%20CA%202-1%202015,
O=D-Trust%20GmbH,
C=DE",
algorithm="rsa-sha256",
headers="Digest X-Request-ID PSU-ID TPP-Redirect-URI Date",
signature="Base64(RSA-SHA256(signing string))"
TPP-Signature-Certificate:TPP's_eIDAS_Certificate{
  "instructedAmount":{
    "currency":"EUR",
    "amount":"123"
  },
  "debtorAccount":{
    "iban":"DE2310010010123456789"
  },
  "creditor":{
    "name":"Merchant123"
  },
  "creditorAccount":{
    "iban":"DE23100120020123456789"
  },
  "remittanceInformationUnstructured":"Ref Number Merchant"
}
```


Wobei die Signatur-„Zeichenkette“ wie folgt lautet:


```
digest: SHA-256=ZuYiOtZkVxhjWmwTO5lOpsPevUNMezvk6dfb6fVhebM=
x-request-id: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
```

psu-id: PSU-1234

Hinweis: Die zu signierenden Header-Felder werden in kleinen Buchstaben angegeben, um zu verdeutlichen, dass der Digest kleine Buchstaben für die Normalisierung verwendet.

11 Komplexe Datentypen und Codelisten

Im Folgenden werden struktuierte Datentypen definiert, die in den Parameterabschnitten dieses Dokuments verwendet werden.

11.1 PSU Data

Attribut	Typ	Datentyp	K	Beschreibung
password		String	K	Enthält ein Passwort im Klartext.

11.2 TPP Message Information

Attribute	Typ	K	Beschreibung
category	String	M	Nur "ERROR" or "WARNING" erlaubt.
code	Message Code	M	
path	String	K	
text	Max512Text	O	Zusätzlicher Text

11.3 Amount

Attribut	Typ	K	Beschreibung
currency	Currency Code	M	ISO 4217 Alpha 3 Währungscode.
amount	String	M	Der Betrag, mit Nachkommastellen angegeben, wobei die Stellenanzahl der Währungsdefinition entsprechen muss. Bis zu 14 signifikante Stellen. Negative Beträge sind mit Minuszeichen versehen. Das Dezimaltrennzeichen ist ein Punkt. Beispiel: Gültige Darstellungen für EUR mit bis zu zwei Dezimalstellen sind: 1056 5768.2 -1.50 5877.78

11.4 Adresse

Attribut	Typ	K
street	Max70Text	O
buildingNumber	String	O
city	String	O
postalCode	String	O
country	Country Code	M

11.5 Remittance

Attribut	Typ	K	Beschreibung
Reference	Max35Text	M	Die aktuelle Referenz.
Reference Type	Max35Text	O	
ReferenceIssuer	Max35Text	O	

11.6 Links

Die Struktur der Links entspricht (HAL, 2013).

Attribut	Typ	K	Beschreibung
startAuthorisation	href Type	O	Ein Link zu einem Endpoint, an dem der Autorisierung einer Transaktion gestartet werden soll. Für diesen Prozessstart werden keine spezifischen Daten benötigt.
startAuthorisationWithPsuAuthentication	href Type	O	Der Link zu einem Endpoint, an dem die Autorisierung einer Transaktion gestartet werden soll, an dem PSU-Authentifizierungsdaten mit dem entsprechenden Request hochgeladen werden sollen.
selectAuthenticationMethod	href Type	O	Dies ist ein Link zu einer Ressource, bei dem der TPP die anwendbaren Authentifizierungsmethoden des zweiten Faktors für das PSU auswählen kann, wenn es mehrere verfügbare Authentifizierungsmethoden gibt.
authoriseTransaction	href Type	O	Der Link zur Payment Initiation- oder Consentressource, an die die "Transaction Authorisation"-Request gesendet wird. Dies ist der Link zu der Ressource, die die Zahlung oder den Consent autorisiert, indem sie die SCA-Authentifizierungsdaten innerhalb des Embedded SCA-Ansatzes überprüft.
self	href Type	O	Der Link zu der Payment Initiation Resource, die von der Request selbst erstellt wurde. Dieser Link kann später verwendet werden, um den Transaktionsstatus der Payment Initiation abzurufen.
status	href Type	O	Ein Link, um den Status der Transaktionsressource abzurufen.
scaStatus	href Type	O	Ein Link, um den Status der Sub-Ressourcen-Autorisierung oder abzurufen.
account	href Type	O	Ein Link zur Ressource, die die Details eines Kontos bereitstellt.
balances	href Type	O	Ein Link zur Ressource, die den Saldo eines dedizierten Kontos bereitstellt.
transactions	href Type	O	Ein Link zu der Ressource, die die Transaktionshistorie eines bestimmten Kontos bereitstellt.

11.7 href Typen

Attribut	Typ	K	Beschreibung
href	String	M	

11.8 Authentification Objekt

Attribut	Typ	K	Beschreibung
authenticationType	Authentication Type	M	Typ der Authentifizierungsmethode.
authenticationMethodId	Max35Text	M	Eine von der Schnittstelle bereitgestellte Identifikation zur späteren Identifizierung der Auswahl der Authentifizierungsmethode.

Attribut	Typ	K	Beschreibung
name	String	M	Beschreibung der Authentifizierungsmethode wie "SMS OTP am Telefon +49160 xxxxx 28".

11.9 Authentification Typ

Name	Beschreibung
SMS_OTP	Ein SCA-Verfahren, bei dem ein OTP, das mit der zu autorisierenden Transaktion verknüpft ist, über einen SMS-Kanal an den PSU gesendet wird.
TOKEN_OTP	Der Token wird anhand einer Challenge an die TAN ermittelt. Die Challenge wird anhand der Auftragsdaten berechnet.

11.10 Challenge

Attribut	Typ	K	Beschreibung
data	Array of Strings	O	Eine Sammlung von Challenge-Daten.
otpMaxLength	Integer	O	Die maximale Länge für das OTP, die vom PSU eingegeben werden kann.
otpFormat	String	O	Der Formattyp des einzutragenden OTPs. Die zulässigen Werte sind "character" oder "integer".
additionalInformation	String	O	Zusätzliche Erklärung für das PSU, um z.B. den Fallback-Mechanismus für die gewählte SCA-Methode zu erklären. Der TPP ist verpflichtet, dies dem PSU nachzuweisen.

11.11 Message Code

Die zulässigen Fehlercodes der Meldungen und die zugehörigen HTTP-Responsecodes sind nachfolgend aufgeführt.

11.11.1 Service unspezifische HTTP Error Codes

Message Code	HTTP Response Code	Beschreibung
CERTIFICATE_INVALID	401	Der Inhalt des Signatur-/Siegel-Zertifikats entspricht nicht den allgemeinen PSD2- oder Attributanforderungen.
CERTIFICATE_EXPIRED	401	Das Signatur /Siegel-Zertifikat ist abgelaufen.
CERTIFICATE_BLOCKED	401	Das Signatur / Siegel-Zertifikat wurde von der Bank gesperrt.
CERTIFICATE_REVOKED	401	Das Signatur / Siegel-Zerfitikat wurde von der QSTP widerrufen.
CERTIFICATE_MISSING	401	Das obligatorische Signatur / Siegel-Zertifikat war im Request nicht vorhanden.
SIGNATURE_INVALID	401	Die Application Layer eIDAS Signatur für TPP-Authentifizierung ist nicht korrekt.
SIGNATURE_MISSING	401	Die obligatorische Application Layer eIDAS Signatur für TPP-Authentifizierung fehlt.

Message Code	HTTP Response Code	Beschreibung
FORMAT_ERROR	400	Das Format bestimmter Requestfelder entspricht nicht den XS2A-Anforderungen. In der Rückmeldung kann ein expliziter Pfad zum entsprechenden Feld angegeben sein. Dies gilt für Header und Body-Einträge. Sie gilt auch in Fällen, in denen sich diese Einträge auf fehlerhafte oder nicht vorhandene Dateninstanzen beziehen, z. B. eine fehlerhafte IBAN.
PARAMETER_NOT_CONSISTENT	400	Die von TPP übermittelten Parameter sind nicht konsistent. Das gilt nur für Abfrageparameter.
PARAMETER_NOT_SUPPORTED	400	Der Parameter wird vom API-Provider nicht unterstützt.
PSU_CREDENTIALS_INVALID	401	Die PSU-ID kann von der adressierten Schnittstelle nicht abgeglichen werden oder ist gesperrt, oder ein Passwort bzw. OTP war nicht korrekt. Zusätzliche Informationen können hinzugefügt werden.
SERVICE_INVALID	400 (falls Payload) 405 (falls HTTP Methode)	Der angesprochene Service ist für die angesprochenen Ressourcen oder die übermittelten Daten nicht gültig.
SERVICE_BLOCKED	403	Dieser Service ist für das adressierte PSU aufgrund einer kanalübergreifende Sperrung durch die Bank nicht erreichbar.
CONSENT_UNKNOWN	403 (falls Pfad) 400 (falls Payload)	
CONSENT_INVALID	401	Der Consent wurde von diesem TPP erstellt, gilt aber nicht für den angesprochenen Service/die angesprochene Ressource.
CONSENT_EXPIRED	401	Der Consent wurde von diesem TPP erstellt, ist aber abgelaufen und muss erneuert werden.
RESOURCE_UNKNOWN	404 (falls account-id im Pfad) 403 (falls andere Ressource im Pfad) 400 (falls Payload)	Die adressierte Ressource ist für TPP unbekannt.
RESOURCE_EXPIRED	403 (falls Pfad) 400 (falls Payload)	Die adressierte Ressource ist dem TPP zugeordnet, aber abgelaufen, d. h. nicht mehr adressierbar.
RESOURCE_BLOCKED	400	Die adressierte Ressource ist durch diesen Request nicht adressierbar, da sie blockiert wird.
TIMESTAMP_INVALID	400	Timestamp nicht im akzeptierten Zeitraum.
PERIOD_INVALID	400	Gewünschte Zeitspanne außerhalb des gültigen Bereich.
SCA_METHOD_UNKNOWN	400	Die adressierte SCA-Methode in der Authentification Method Select Request ist unbekannt oder kann nicht mit dem PSU assoziiert werden.
STATUS_INVALID	409	Die angesprochene Ressource erlaubt keine zusätzliche Autorisierung.

11.11.2 PIS spezifische Error Codes

Message Code	HTTP Response Code	Beschreibung
PRODUCT_INVALID	403	Das adressierte Zahlungsprodukt ist für dem PSU nicht verfügbar.
PRODUCT_UNKNOWN	404	Das adressierte Zahlungsprodukt wird von der Schnittstelle nicht unterstützt.
PAYMENT_FAILED	400	Die Payment Initiation des POST-Requests ist während des initialen Prozesses fehlgeschlagen.
EXECUTION_DATE_INVALID	400	Das gewünschte Ausführungsdatum ist kein gültiges Ausführungsdatum für die Schnittstelle.

11.11.3 AIS spezifische HTTP Error Codes

Message Code	HTTP Response Code	Beschreibung
CONSENT_INVALID	401	Die Consentsdefinition ist nicht vollständig oder ungültig.
SESSIONS_NOT_SUPPORTED	400	Der „combined service Flag“ kann mit dieser Schnittstelle nicht verwendet werden.
ACCESS_EXCEEDED	429	Der Zugriff auf das Konto übersteigt die genehmigte Multiplizität ohne PSU-Beteiligung pro Tag. Der Zugriff auf das Konto hat die vereinbarte Menge pro Tag überschritten. Dieser Code wird nur bei Zugriff auf Kontoinformationen ohne die Beteiligung der PSU verwendet.

11.11.4 PIIS spezifische Error Codes

Message Code	HTTP Response Code	Beschreibung
CARD_INVALID	400	Die adressierte Kartennummer ist der Schnittstelle nicht bekannt oder kann nicht dem PSU zugeordnet werden.
NO_PIIS_ACTIVATION	400	Der PSU hat das adressierte Konto für die Nutzung des dem TPP zugeordneten PIIS aktiviert.

11.12 Transaction Status

Dies ist ein Datenelement zur Unterstützung der Deklaration zusätzlicher Fehler.

Code	Name	ISO 20022 Definition
ACTC	AcceptedTechnicalValidation	Authentifizierung, syntaktische und semantische Validierung sind erfolgreich.
RCVD	Received	Die Payment Initiation ist bei der Empfangsstelle eingegangen.
PDNG	Pending	Die Payment Initiation oder einzelne Transaktionen, die in der Payment Initiation enthalten sind, sind noch nicht abgeschlossen. Weitere Prüfungen und Statusaktualisierungen werden durchgeführt.

Code	Name	ISO 20022 Definition
RJCT	Rejected	Die Payment Initiation oder eine einzelne Transaktion, die in der Payment Initiation enthalten ist, wurde abgelehnt.

11.13 Consent Status

Code	Beschreibung
received	Die Consentdaten sind eingegangen und technisch korrekt. Die Daten sind noch nicht autorisiert.
rejected	Die Consentdaten wurden abgelehnt, z.B. weil keine erfolgreiche Autorisierung stattgefunden hat.
valid	Der Consent wird akzeptiert und gilt für GET-Konto-Datenabrufe und andere, wie im Consentobjekt angegeben.
revokedByPsu	Der Consent wurde vom PSU gegenüber der Schnittstelle widerrufen.
expired	Der Consent ist abgelaufen.
terminatedByTpp	Die entsprechende TPP hat den Consent durch Anwendung der DELETE-Methode auf der Consentressource beendet.

11.14 SCA Status

Für diesen Datentyp sind die folgenden Codes definiert.

Code	Beschreibung
received	Eine Autorisierungsressource wurde erfolgreich angelegt.
psuIdentified	Der PSU, auf den sich die Autorisierung bezieht, wurde identifiziert.
psuAuthenticated	Der PSU, auf den sich die Autorisierung bezieht, wurde identifiziert und authentifiziert.
scaMethodSelected	Der PSU/TPP hat die zugehörige SCA-Routine ausgewählt. Wird die SCA-Methode implizit gewählt, da nur eine SCA-Methode verfügbar ist, so ist dies der erste Status, der anstelle von "received" gemeldet wird.
started	Die angesprochene SCA-Routine wurde gestartet.
finalised	Die angesprochene SCA-Routine wurde beendet.
failed	Die SCA-Routine ist gescheitert.
exempted	SCA wurde für die betreffende Transaktion freigestellt, die entsprechende Autorisierung ist erfolgreich.

11.15 Account Access

Attribut	Type	K	Beschreibung
accounts	Array der Kontoreferenz	O	Fordert eine detaillierte Account Information an. Wenn angegeben, muss eine nicht leere Liste von Kontoreferenzen eingestellt werden.
balances	Array der Kontoreferenz	O	Fordert die Salden der angegebenen Konten an. Wenn angegeben muss eine nicht leere Liste von Kontoreferenzen eingestellt werden.
transactions	Array der Kontoreferenz	O	Fordert Umsatzinformaitionen der adressierten Konten an.
availableAccounts	String	O	Es wird nur der Wert "allAccounts" zugelassen.

11.16 Account Reference

Dieser Typ enthält eine Kontokennung, die auf Payload-Level verwendet werden kann, um Konten anzusprechen.

Attribut	Typ	K	Beschreibung
iban	IBAN	K	

11.17 Account Details

Attribute	Type	K	Beschreibung
resourceId	String	K	Dies ist das Datenelement, das im Pfad beim Abrufen von Daten von einem dedizierten Konto verwendet werden soll, siehe Abschnitt 6.4.1 oder Abschnitt 6.4.2 unten. Dies ist auszufüllen, wenn adressierbare Ressourcen von der Schnittstelle auf dem /accounts Endpoint erstellt werden.
iban	IBAN	O	Dieses Datenelement kann im Body der Consent Request-Nachricht verwendet werden, um den Consent zum Zugriff auf das Konto von diesem Zahlungskonto abzurufen.
currency	Currency Code	M	Kontowährung.
ownerName	Max140Text	M	Name des Kontoinhabers
name	Max35Text	O	Name des von der Bank oder dem PSU angegebenen Kontos im Online Banking.
bic	BICFI	O	Der BIC, der dem Konto zugeordnet ist.
_links	Links	O	Links zum Konto, die direkt zum Abrufen von Accounts Information von diesem speziellen Konto verwendet werden können. Links zu "balances" und/oder "transactions". Diese Links werden nur dann unterstützt, wenn der entsprechende Consent bereits erteilt wurde.

11.18 Balance Type

Die folgenden Saldenarten schließen Kreditlimits aus.

Anmerkung: Diese Definition folgt der ISO20022-Logik für die Definition von Saldenarten.

Type	Beschreibung
closingBooked	Saldo des Kontos am Ende der vorab vereinbarten Berichtsperiode. Es ist die Summe aus dem Eröffnungssaldo zu Beginn der Periode und allen Buchungen, die während der vorher vereinbarten Berichtsperiode auf dem Konto gebucht wurden.
expected	Saldo, der sich aus gebuchten Buchungen und zum Zeitpunkt der Berechnung bekannten offenen Posten zusammensetzt, der den Tagesendbestand projiziert, wenn alles auf dem Konto gebucht ist und kein anderer Eintrag gebucht wird.
interimBooked	Der Saldo wird im Laufe des Werktages des Kontoführers, zu dem angegebenen Zeitpunkt und weiterer Änderungen während des Werktages berechnet. Das Zwischensaldo wird auf Basis dergebuchten Kreditor-und Debitorposten während des Berechnungszeitraums berechnet.

11.19 Balance

Attribut	Typ	K	Beschreibung
balanceAmount	Amount	M	
balanceType	Balance Type	M	
referenceDate	ISODate	O	Referenzdatumdes Saldos

11.20 Account Report

Attribut	Typ	K	Beschreibung
Booked	Array der Transaktionen	K	Muss enthalten sein, wenn der Parameter „bookingStatus“ „booked“ oder „both“ gesetzt ist.
pending	Array der Transaktionen	K	Nicht enthalten, wenn der Parameter „bookingStatus“ auf "booked" gesetzt ist."
_links	Links	M	Der Link „account“ (obligatorisch) muss in diesem Kontext verwendet werden.

11.20.1 Transaktionen

Attribut	Typ	K	Beschreibung
endToEndId	Max35Text	O	Eindeutige End-to-End-Identität.
mandateId	Max35Text	O	Identifizierung von Mandaten, z.B. eine SEPA-Mandats-ID.
creditorId	Max35Text	O	Identifizierung von Kreditoren, z.B. eine SEPA-Creditor-ID.
bookingDate	ISODate	O	Das Datum, an dem eine Buchung auf ein Konto der Schnittstelle gebucht wird.
valueDate	ISODate	O	Das Datum, an dem die Vermögenswerte dem Kontoinhaber im Falle einer Gutschrift zur Verfügung stehen.
transactionAmount	Amount	M	Der Betrag der fakturierten Transaktion auf dem Konto.
creditorName	Max70Text	O	Name des Kreditors bei einer "belasteten" Transaktion.
creditorAccount	Account Reference	K	Kontonummer / IBAN des Empfängers einer Zahlung
debtorName	Account Reference	O	Name des Schuldners bei einem "gutgeschriebenen" Transaktion.
debtorAccount	Account Reference	K	Kontonummer / IBAN des Zahlungspflichtigen einer Zahlung
remittance Information Unstructured	Max140Text	O	Verwendungszweck
proprietaryBank TransactionCode	Max35Text	O	Proprietärer Banktransaktionscode, wie er innerhalb einer Community oder innerhalb eines Schnittstelle verwendet wird, z.B. für MT94x-basierte Transaktionsberichte.
additionalInformation	Max512Text	O	Kann von der ASPSP verwendet werden, um zusätzliche transaktionsbezogene Informationen an das PSU zu übertragen.
standingOrderDetails	Standing Order Details	M	Detailangaben des vorhandenen Dauerauftrages

Attribut	Typ	K	Beschreibung
startDate	ISODate	M	Tag der ersten Ausführung des Dauerauftrages
frequency	Frequency Code	M	Turnus der Ausführungen z.B. monatlich, wöchentlich, jährlich … Folgende Daten sind möglich: weekly everyTwoWeeks monthly everyTwoMonths quarterly semiAnnual annual
Multiplicator	Numerical	O	Multiplikator des Turnus. Beispiel: Turnus = monatlich und Multiplikator = 4. Dies bedeutet, dass der Dauerauftrag alle 4 Monate ausgeführt wird.
dayOfExecution	Max2Text	O	Ausführungstag des Dauerauftrages. Der Ausführungstag 31 steht für eine Ausführugn am Ultimo.

11.21 Andere ISO-basierte Basistypen

Die folgenden Codes und Definitionen werden ab ISO 20022 verwendet:

- **BICFI**: BICFIIdentifier
- **IBAN**: IBAN2007Identifier
  Muster: [A-Z]{2}[0-9]{2}[A-Z0-9]{1-30}

Die folgenden Codes werden von anderen ISO-Normen verwendet:

- **Currency Code**: Codes nach ISO 4217 Alpha 3
- **Country Code**: Zwei Zeichen gemäß ISO 3166

Weitere grundlegende ISO-Datentypen:

- **ISODateTime**: Ein bestimmter Punkt im Verlauf der Zeit, der durch ein obligatorisches Datum und eine obligatorische Zeitkomponente definiert ist, ausgedrückt entweder im UTC-Zeitformat (YYYYY-MM-DDThh:mm:ss.sssZ), der lokalen Zeit mit UTC-Offsetformat (YYYYY-MM-DDThh:mm:ss.sss+/-hh:mm) oder dem lokalen Zeitformat (YYYY-MMDThh:mm:ss.sss). Diese Darstellungen sind im "XML-Schema Teil 2: Datatypes Second Edition - W3C Recommendation 28 October 2004" definiert, das auf ISO 8601 abgestimmt ist.
- **ISODate**: Ein bestimmter Zeitpunkt im Zeitverlauf eines Kalenderjahres, ausgedrückt im Format JJJJJ-MM-TT.

12 Literaturverzeichnis

EBARTS. (13. März 2018). Commission Delegated Regulation (EU) 2018/389 of 27 November 2017 supplementing Directive 2015/2366 of the European Parliament and of the Council with regard to Regulatory Technical Standards for Strong Customer Authentication and Common and Secure Open S. *C(2017) 7782 final.*

eIDAS. (28. 08 2014). Regulation (EU) No 910/2014 of the European Parliament and of the Council on Electronic Identification and Trust Services for Electronic Transactions in the Internal Market,. 23 July 2014.

HAL. (18. 09 2013). *HAL - Hypertext Application Language*. Von http://stateless.co/hal_specification.html abgerufen

NextGenPSD2. (März. 2019 2018). *Access to Account Interoperability Framework - Implementation Guidelines.* 1.3. Von https://www.berlin-group.org/nextgenpsd2-downloads abgerufen

PSD2. (23. Dezember 2015). Directive (EU) 2015/2366 of the European Parliament and of the Council on payment services in the internal market.

RFC4648. (10 2006). *The Base16, Base32, and Base64 Data Encodings*. Von Josefsson, S.: https://tools.ietf.org/html/rfc4648 abgerufen

RFC7807. (03 2016). *Problem Details for HTTP APIs*. Von M. Nottingham, Akamai, E. Wilde: https://tools.ietf.org/html/rfc7807 abgerufen

signHTTP. (10. 02 2019). *Signing HTTP messages, Network Working Group, Internet Draft version 10*. Von https://datatracker.ietf.org/doc/draft-cavage-http-signatures/ abgerufen

XS2A-DP. (Aktuelle Version). *NextGenPSD2 XS2A Framework, Domestic Payment Definitions, The Berlin Group Joint Initiative on a PSD2 Compliant XS2A Interface.*

XS2A-OR. (08. 02 2018). NextGenPSD2 XS2A Framework, Operational Rules, The Berlin Group Joint Initiative on a PSD2 Compliant XS2A Interface. Version 1.0.