Data Standards Body
Technical Working Group
Decision 052 – Scheduled Payments

Contact: James Bligh

Publish Date: 26th May 2019

Decision Approved By Chairman: 28th May 2019

Context

The interpretation of the designated CDR Data for Banking provided by the ACCC includes Scheduled Payments. These are payment instructions for recurring or future dated payments that have been stored by a Customer with a Bank. This decision describes the standards for the API end points and payloads for this data category.

Decision To Be Made

Define the API end point and payloads standards for Scheduled Payments.

Feedback Provided

The original proposal and the associated feedback can be found at:
https://github.com/ConsumerDataStandardsAustralia/open-banking/issues/51

The feedback was all highly constructive and mainly focussed on ensuring the payload would allow for payments to be sufficiently represented. This feedback was accommodated as much as possible.

Feedback also included concerns from Banks regarding the nature of the data included in the payload. In response to this feedback the data has been reviewed to ensure the data fields included are aligned with the requirements of the CDR Rules.

Decision For Approval

The sections below contain details of the end points to be included in the standards. Each end point identifies the HTTP Method, URI and payloads associated with each end point.

Scheduled Payments End Points Summary

A summary of the schedule payments end points:

- GET /banking/accounts/{accountId}/payments/scheduled
- GET /banking/payments/scheduled
- POST /banking/ payments/scheduled

Account Specific Scheduled Payments Data

High Level Information

Title	Obtain scheduled, outgoing payments for a specific account
HTTP Method	GET
URI	/banking/accounts/{accountId}/payments/scheduled
Security Scope	Bank Regular Payments
Pagination	Supported
Specific Errors	Not supported – No specific error payloads expected to be returned
Path Parameters	**accountId** ID of the account to get scheduled payments for. Must have previously been returned by one of the account list end points. The account specified is the source account for the payment
Query Parameters	**page** Page of results to request (standard pagination) **page-size** Page size to request. Default is 25 (standard pagination)

Request Payload

Not applicable

Response Payloads

HTTP Response Code: 200 OK

Field	Type	Mandatory	Description
data	Object	Mandatory	
{			
scheduledPayments	Array[Object]	Mandatory	The list of scheduled payments to return
[{			
scheduledPaymentId	ASCIIString	Mandatory	A unique ID of the scheduled payment adhering to the standards for ID permanence
nickname	String	Optional	The short display name of the payee as provided by the customer

Field	Type	Mandatory	Description
payerReference	String	Mandatory	The reference for the transaction that will be used by the originating institution for the purposes of constructing a statement narrative on the payer's account. Empty string if no data provided
payeeReference	String	Mandatory	The reference for the transaction that will be provided by the originating institution. Empty string if no data provided
status	Enum	Mandatory	Indicates whether the schedule is currently active. Valid values are: • ACTIVE • SKIP • INACTIVE The value SKIP is equivalent to ACTIVE except that the customer has requested the next normal occurrence to be skipped.
from	Object	Mandatory	Object containing details of the source of the payment. Currently only specifies an account ID but provided as an object to facilitate future extensibility and consistency with the to object
{			
accountId	ASCIIString	Mandatory	ID of the account that is the source of funds for the payment
}			
paymentSet	Array[Object]	Mandatory	The set of payment amounts and destination accounts for this payment accommodating multi-part payments. A single entry indicates a simple payment with one destination account. Must have at least one entry.
[{			
to	Object	Mandatory	Object containing details of the destination of the payment. Used to specify a variety of
{			

Field	Type	Mandatory	Description
toUType	Enum	Mandatory	The type of object provided that specifies the destination of the funds for the payment. Valid values are: • accountId • payeeId • domestic • biller • international
accountId	String	Conditional	Present if toUType is set to accountId. Indicates that the payment is to another account that is accessible under the current consent.
payeeId	String	Conditional	Present if toUType is set to payeeId. Indicates that the payment is to registered payee that can be accessed using the payee end point. If the Bank Payees scope has not been consented to then a payeeId should not be provided and the full payee details should be provided instead
domestic	Domestic Payee Object as defined in Decision 058	Conditional	Present if toUType is set to domestic. Indicates a domestic account
biller	Biller Payee Object as defined in Decision 058	Conditional	Present if toUType is set to biller. Indicates a BPAY biller
international	International Payee Object as defined in Decision 058	Conditional	Present if toUType is set to international. Indicates an international beneficiary
}			
isAmountCalculated	Boolean	Optional	Flag indicating whether the amount of the payment is calculated based on the context of the event. For instance a payment to reduce the balance of a credit card to zero. If absent then false is assumed.
amount	AmountString	Conditional	The amount of the next payment if known. Mandatory unless the isAmountCalculated field is set to true. Must be zero or positive if present.
currency	CurrencyString	Optional	The currency for the payment. AUD assumed if not present
}]			

Field	Type	Mandatory	Description
recurrence	Object	Mandatory	Object containing the detail of the schedule for the payment
{			
nextPaymentDate	DateString	Optional	The date of the next payment under the recurrence schedule if known. For event based payments the specific date of the next payment may be undetermined
recurrenceUType	Enum	Mandatory	The type of recurrence used to define the schedule. Valid values are: • onceOff • intervalSchedule • lastWeekDay • eventBased
onceOff	Object	Conditional	Indicates that the payment is a once off payment on a specific future date. Mandatory if recurrenceUType is set to onceOff.
{			
paymentDate	DateString	Mandatory	The scheduled date for the once off payment
}			
intervalSchedule	Object	Conditional	Indicates that the schedule of payments is defined by a series of intervals. Mandatory if recurrenceUType is set to intervalSchedule
{			
finalPaymentDate	DateString	Optional	The limit date after which no more payments should be made using this schedule. If both finalPaymentDate and paymentsRemaining are present then payments will stop according to the most constraining value. If neither field is present the payments will continue indefinitely
paymentsRemaining	PositiveInteger	Optional	Indicates the number of payments remaining in the schedule. If both finalPaymentDate and paymentsRemaining are present then payments will stop according to the most constraining value. If neither field is present the payments will continue indefinitely

Field	Type	Mandatory	Description
nonBusinessDayTreatment	Enum	Optional	Enumerated field giving the treatment where a scheduled payment date is not a business day. If absent assumed to be “ON”. Valid values are: • AFTER If a scheduled payment date is a non-business day the payment will be made on the first business day after the scheduled payment date • BEFORE If a scheduled payment date is a non-business day the payment will be made on the first business day before the scheduled payment date • ON If a scheduled payment date is a non-business day the payment will be made on that day regardless • ONLY Payments only occur on business days. If a scheduled payment date is a non-business day the payment will be ignored
intervals	Array[Object]	Mandatory	An array of interval objects defining the payment schedule. Each entry in the array is additive, in that it adds payments to the overall payment schedule. If multiple intervals result in a payment on the same day then only one payment will be made. Must have at least one entry.
[{			
interval	ExternalRef	Mandatory	An interval for the payment. Formatted according to ISO 8601 Durations with components less than a day in length ignored. This duration defines the period between payments starting with nextPaymentDate.
dayInInterval	ExternalRef	Optional	Uses an interval to define the ordinal day within the interval defined by the interval field on which the payment occurs. If the resulting duration is 0 days in length or larger than the number of days in the interval then the payment will occur on the last day of the interval. A duration of 1 day indicates the first day of the interval. If absent the assumed value is P1D. Formatted according to ISO 8601 Durations with components less than a day in length ignored. The first day of a week is considered to be Sunday.

Field	Type	Mandatory	Description
}]			
}			
lastWeekDay	Object	Conditional	Indicates that the schedule of payments is defined according to the last occurrence of a specific weekday in an interval. Mandatory if recurrenceUType is set to lastWeekDay
{			
finalPaymentDate	DateString	Optional	The limit date after which no more payments should be made using this schedule. If both finalPaymentDate and paymentsRemaining are present then payments will stop according to the most constraining value. If neither field is present the payments will continue indefinitely
paymentsRemaining	PositiveInteger	Optional	Indicates the number of payments remaining in the schedule. If both finalPaymentDate and paymentsRemaining are present then payments will stop according to the most constraining value. If neither field is present the payments will continue indefinitely
interval	ExternalRef	Mandatory	The interval for the payment. Formatted according to ISO 8601 Durations with components less than a day in length ignored. This duration defines the period between payments starting with nextPaymentDate.
lastWeekDay	PositiveInteger	Mandatory	The weekDay specified. The payment will occur on the last occurrence of this weekday in the interval. Value is constrained to 1 to 7 with 1 indicating Sunday.
}			
eventBased	Object	Conditional	Indicates that the schedule of payments is defined according to an external event that cannot be predetermined. Mandatory if recurrenceUType is set to eventBased
{			
description	String	Mandatory	Description of the event and conditions that will result in the payment. Expected to be formatted for display to a customer
}			
}			

Field	Type	Mandatory	Description
}]			
}			
links	Object	Mandatory	
{			
self	URI	Mandatory	Fully qualified link to this API call
first	URI	Conditional	URI to the first page of this set. Mandatory if this response is not the first page
prev	URI	Conditional	URI to the previous page of this set. Mandatory if this response is not the first page
next	URI	Conditional	URI to the next page of this set. Mandatory if this response is not the last page
last	URI	Conditional	URI to the last page of this set. Mandatory if this response is not the last page
}			
meta			
{			
totalRecords	NaturalNumber	Mandatory	The total number of records in the full set
totalPages	NaturalNumber	Mandatory	The total number of pages in the full set
}			

Bulk Scheduled Payments Data

High Level Information

Title	Obtain scheduled, outgoing payments for multiple, filtered accounts that are the source of funds for the payments
HTTP Method	GET
URI	/banking/payments/scheduled
Security Scope	Bank Regular Payments
Pagination	Supported
Specific Errors	Not supported – No specific error payloads expected to be returned
Path Parameters	None
Query Parameters	**product-category** Used to filter results on the productCategory field defined in the account end points. Any one of the valid values for this field can be supplied. If absent then transactions for all available accounts returned. **open-status** Used to filter results according to open/closed status. Values can be OPEN, CLOSED or ALL. If absent then ALL is assumed **is-owned** Filters accounts based on whether they are owned by the authorised customer **page** Page of results to request (standard pagination) **page-size** Page size to request. Default is 25 (standard pagination)

Request Payload

Not applicable

Response Payloads

HTTP Response Code: 200 OK

Field	Type	Mandatory	Description
data	Object	Mandatory	
{			
scheduledPayments	Object	Mandatory	Duplicate of the definition of scheduledPayments defined above in the *Account Specific Scheduled Payments* end point definition.

Field	Type	Mandatory	Description
}			
links	Object	Mandatory	
{			
self	URI	Mandatory	Fully qualified link to this API call
first	URI	Conditional	URI to the first page of this set. Mandatory if this response is not the first page
prev	URI	Conditional	URI to the previous page of this set. Mandatory if this response is not the first page
next	URI	Conditional	URI to the next page of this set. Mandatory if this response is not the last page
last	URI	Conditional	URI to the last page of this set. Mandatory if this response is not the last page
}			
meta			
{			
totalRecords	NaturalNumber	Mandatory	The total number of records in the full set
totalPages	NaturalNumber	Mandatory	The total number of pages in the full set
}			

Specific Scheduled Payments Data

High Level Information

Title	Obtain scheduled, outgoing payments for a specified list of accounts
HTTP Method	POST
URI	/banking/payments/scheduled
Security Scope	Bank Regular Payments
Pagination	Supported
Specific Errors	Supported – Specific account specified cannot be identified
Path Parameters	None
Query Parameters	**page** Page of results to request (standard pagination) **page-size** Page size to request. Default is 25 (standard pagination)

Request Payload

Field	Type	Mandatory	Description
data	Object	Mandatory	
{			
accountIds	Array[String]	Mandatory	Array of specific accountIds to obtain scheduled payments for. The accounts specified are the source of funds for the payments returned
}			
meta	Object	Optional	
{			
}			

Response Payloads

HTTP Response Code: 200 OK

Field	Type	Mandatory	Description
data	Object	Mandatory	
{			

Field	Type	Mandatory	Description
scheduledPayments	Object	Mandatory	Duplicate of the definition of scheduledPayments defined above in the *Account Specific Scheduled Payments* end point definition.
}			
links	Object	Mandatory	
{			
self	URI	Mandatory	Fully qualified link to this API call
first	URI	Conditional	URI to the first page of this set. Mandatory if this response is not the first page
prev	URI	Conditional	URI to the previous page of this set. Mandatory if this response is not the first page
next	URI	Conditional	URI to the next page of this set. Mandatory if this response is not the last page
last	URI	Conditional	URI to the last page of this set. Mandatory if this response is not the last page
}			
meta			
{			
totalRecords	NaturalNumber	Mandatory	The total number of records in the full set
totalPages	NaturalNumber	Mandatory	The total number of pages in the full set
}			

HTTP Response Code: 422 Unprocessable Entity

Field	Type	Mandatory	Description
errors	Array[Object]	Mandatory	
[			
{			
code	String	Mandatory	Must be one of the following: • 0001 – Account not able to be found

Field	Type	Mandatory	Description
title	String	Mandatory	Must be one of the following: • “Invalid account”
detail	String	Mandatory	ID of the account not found
}			
]			

Example Schedules

This section describes how the recurrence objects would be used to describe different types of payment schedules. This section is not considered part of the proposal and does not include any additions to the proposed standard.

Payment every interval

Payments every four days from the next payment date:

```
"intervalSchedule": {
  "intervals": [
    {
      "interval": "P4D"
    }
  ]
}
```

Payment every interval of business days

Payments every 27 business days:

```
"intervalSchedule": {
  "nonBusinessDayTreatment": "ONLY",
  "intervals": [
    {
      "interval": "P27D"
    }
  ]
}
```

Payment each week on specific weekday

Payment every week on Tuesday

```
"intervalSchedule": {
  "intervals": [
    {
      "interval": "P1W",
      "dayInInterval": "P3D"
    }
  ]
}
```

Payment every month on a specific date

Payment each month on the 21st

```
"intervalSchedule": {
  "intervals": [
    {
      "interval": "P1M",
      "dayInInterval": "P21D"
    }
  ]
}
```

Payment on the first day of the month

```
"intervalSchedule": {
  "intervals": [
```

```
      {
        "interval": "P1M"
      }
    ]
  }
```

Payment on the 15th and last day of the month

```
  "intervalSchedule": {
    "intervals": [
      {
        "interval": "P1M",
        "dayInInterval": "P15D"
      },
      {
        "interval": "P1M",
        "dayInInterval": "P0D"
      }
    ]
  }
```

Payment on the last week day of an interval

Payment on the last Wednesday of the month

```
  "lastWeekDay": {
      "interval": "P1M",
      "lastWeekDay": 4
    }
  }
```

Payment on the last business day of an interval

Payment on the last business day of the month

```
  "intervalSchedule": {
    "nonBusinessDayTreatment": "BEFORE",
    "intervals": [
      {
        "interval": "P1M",
        "dayInInterval": "P0D"
      }
    ]
  }
```

Payment every quarter

Payment on the first business day of the second month each quarter

```
  "intervalSchedule": {
    "nonBusinessDayTreatment": "AFTER",
    "intervals": [
      {
        "interval": "P3M",
        "dayInInterval": "P1M"
      }
    ]
  }
```

Payment every half year

Payment on the last Monday each half year

```
"lastWeekDay": {
    "interval": "P6M",
    "lastWeekDay": 1
  }
}
```

Payment every year

Payment every year on the 30th August

```
"intervalSchedule": {
  "intervals": [
    {
      "interval": "P1Y",
      "dayInInterval": "P7M30D"
    }
  ]
}
```

Automatic credit card payment

Payment of full balance if the balance goes above $1,000

```
"eventBased": {
  "description": "Payment of balance when $1,000 is exceeded"
}
```