Transaction Management : Capture / Cancellation / Refund

This page describes the possible operations on a transaction created via Axepta BNP Paribas Online:

- Capture (automatic, delayed, manual, partial)
- Cancellation (before capture)
- Refund (after capture)

Content

Use case: Authorize, Capture, Cancellation, Refund

Authorize

When a card transaction is created, an authorization is sent to the cardholder's bank to reserve the amount.

As long as the capture is not made, the funds are not debited.

Capture

The capture refers to the request for bank transfer.

This is the step that triggers the actual debit of the cardholder.

Cancellation

A cancellation is possible as long as the capture has not already been made.

She immediately releases the reserved funds.

Refund

A refund is made on a transaction already captured.

It can be total or partial.

Capture

When initializing the transaction, you can control the behavior via the captureMethod object.

```
"captureMethod": {  "type": "AUTOMATIC" | "DELAYED" | "MANUAL"}
```

If captureMethod is missing automatic capture by default.

Automatic capture (default mode)

Functional

The capture request is performed automatically right after the authorization.

No action is required, the transaction will be sent for settlement during the next collection cycle (end of day).

Unless you have specific use cases, this is the mode to prioritize.

Recommended use cases

- Classic e-commerce payment
- Process without complex inventory management
- Merchants not needing to check availability before capture

Automatic delayed capture (DELAYED)

Functional

When requesting a transaction, it is possible to specify a time limit (in hours) before the capture request is executed.

This option makes it possible to ensure that the transaction will not be captured during this period.

The payment may be delayed by one day if the deadline exceeds the daily settlement time.

This capture must be carried out within 7 days following authorization, in agreement with the funds guarantee deadline at the bank.

Features

- The deadline ensures that no capture will occur during X hours.
- Allows for easy cancellation of the transaction before capture.
- May shift the banking settlement by one day if the deadline exceeds the settlement time.

Use case

- Deferred stock verification
- Manual checks before shipping
- Anti-fraud requiring a delay before capture

Integration

The Capture endpoint allows capturing the funds using the `payId` of the original transaction.

(Optional amount partial capture if different from the authorized amount.)

captureMethod

```
...
        "captureMethod": {
                "type": "DELAYED",
                "delayed": {
                        "delayedHours": 2
                }
    },
...
```

Manual Capture (MANUAL)

Functional

Manual capture allows for capturing all or only part of the authorized amount.

In the case where part of the amount is captured, this is referred to as partial capture.

So, with this option, the capture is never executed automatically; it is essential to send a capture order for the funds to be collected.

The banking settlement date for these transactions will be the date of the capture request.

> This capture must be carried out within 7 days following authorization, in agreement with the funds guarantee deadline at the bank.

Integration

The Capture endpoint allows capturing the funds using the `payId` of the original transaction.

(Optional amount partial capture if different from the authorized amount.)

captureMethod

```
...
        "captureMethod": {
                "type": "MANUAL"
    },
...
```

Cancellation

Functional

- Cancellation is possible only if the transaction has not been captured (capturedValue = 0).
- It allows to release the funds reserved on the customer's card.

> The cancellation cannot be partial. The cancelled amount = authorized amount.

Integration

The Reverse endpoint allows to cancel a transaction (or a step of the transaction). It is necessary to provide the `payId` of the original transaction.

> The status affects the number of calls required.

Status	Meaning	Action of the Reversal Call	Number of Calls Required
OK	Capture requested but not executed	Cancels the capture request changes to AUTHORIZED	2 calls
AUTHORIZED	Authorization pending capture	Cancels the authorization	1 call

Refund

Functional

Only a transaction already captured (capturedValue > 0) or settled (status Sale) can be refunded.

The refund can be total or partial refund.

Several successive refunds possible up to the captured amount.

Integration

The Refund endpoint allows you to initiate a refund on an already settled transaction. It is necessary to provide the `payId` of the transaction to be refunded.

Should I cancel or refund my transaction?

In the context of transaction management, it is essential to clearly distinguish between situations requiring the cancellation of a payment and those requiring a refund.

This section describes the key steps to determine the best action to take when an order cannot be fulfilled.

Whether for logistical, technical, or other reasons, understanding the payment status and the amounts involved is essential.

By checking these elements, it will be possible to determine whether the payment should be canceled (when the funds have not been claimed by the holder) or refunded (when the funds have indeed been claimed). This approach ensures transparent and compliant management with client expectations, while optimizing operational processes.

> If the transaction has been paid, then the customer has been debited. In this case, a refund must be initiated.
>
> If the transaction has not been paid, then the customer has not been debited. In this case, a cancellation must be initiated.

> For all operations, it is necessary to use the payment reference 'payId' returned by the payment platform during the transaction.

Step 1: Check the transaction status

The GetByPayId endpoint allows you to recover the transaction information by providing the corresponding `payId`.

Example of getByPayId response


```
{
  "amount": {
    "currency": "EUR",
    "value": 1000,
        "capturedValue": 1000
  },
  "externalIntegrationId": "12345-abcde-67890-fghij",
  "language": "en",
  "payId": "ed7773c09f314c45b2a0de47a1994e1d",
  "xId": "86a8198cc9da4b4d9fbcdb15a5ddd6ca",
  "transId": "20240103-121913-554",
  "status": "OK",
  "responseCode": "00000000",
  "responseDescription": "success",
  "refNr": "refNr01",
  "metadata": {
    "userData": "my user data",
    "plain": "some plain text",
    "key1": "value1",
    "key2": "value2"
  },
  "paymentMethods": {
    "type": "CARD",
    "card": { … }
  }
}
```


To determine whether it is necessary to cancel or refund a transaction, it is necessary to consult the properties **'status'** and **'responseCode'** of the transaction:

- the property **'status'** must be equal to "OK" or "AUTHORIZED" (to keep if cancellation is needed)
- the property **'responseCode'** must be equal to "00000000".
    - Refer to Codes réponse / Error codes (1) for details on the response codes


```
   "status": "OK",
   "responseCode": "00000000",
```


Step 2: Check the transaction amounts

The amounts of the transaction are detailed in the JSON object ***'amount'***.

Objet 'amount'


```
   "amount": {
    "currency": "EUR",
    "value": 1000,
        "capturedValue": 1000
  },
```

It is necessary to consult the amounts ***'value'*** and ***'capturedValue'***

- **'value'**: Authorized amount. It must be greater than zero.

  If it is equal to zero, it means that the transaction authorization has failed, and that the transaction did not go through.

- **'capturedValue':** Amount of the captured transaction.

'capturedValue' will determine whether a refund or a cancellation should be initiated.

If capturedValue = 0: the transaction is not captured => Cancellation
If capturedValue > 0: the transaction is captured => Refund

Step 3a: Refund

The Refund endpoint allows you to initiate a refund on an already settled transaction. It is necessary to provide the `payId` of the transaction to be refunded.

The amount to be refunded may be all or part of the initial transaction amount.
This refers to a partial refund.
It is possible to make a series of partial refunds up to the authorized amount.

Step 3b: Cancellation

The Reverse endpoint allows you to cancel a transaction (or a step of the transaction). It is necessary to provide the `payId` of the original transaction.

The cancellation amount must be exactly equal to the authorized amount. Partial cancellations are not possible.

Cancellation can be done in 1 or 2 steps, depending on whether the capture request has already been made or not.

The Reverse endpoint allows for:

- Cancel the capture request and return to the 'Authorized' step, if the transaction is pending capture.
- Cancel the authorization request, and definitively cancel the entire transaction, if the transaction is authorized and the capture request has not been made.

To know at what step the transaction is, you need to check its 'status':

- **"OK"**
  The capture has been requested but has not been done yet.

The transaction is pending capture.
The API call to reverse will cancel the capture request and change the transaction status to "AUTHORIZED".

- **"AUTHORIZED"**
  The request for capture was not made (or was canceled).
  The API call to reverse will cancel the authorization.
  The funds reserved during the authorization will be released.
  The transaction is definitely cancelled.

In conclusion:

- To cancel a transaction with the status equal to **"OK"**, 2 calls to the Reverse endpoint are necessary
  - a first call to cancel the capture request, and a second to cancel the authorization
- To cancel a transaction with the status equal to **"AUTHORIZED"**, 1 call to the Reverse endpoint
  - a single call to cancel the authorization