|  |  |  |
| --- | --- | --- |
| **Nature of the document** | MIP interface contract | |
| **project code** | MIP | |
| **Project identification**  **MIP interface contract** | | Creation date: 05/04/2015  Version: V1.3 |

**Identification document**

|  |  |
| --- | --- |
| descriptive |  |
| Keywords | To be determined later |
| reference Documents |  |

**Version History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Goal** | **Author** | **Dated** |
| V1.0 | Created from an extract of SFG Interfaces ETS | D. Dentant | 05/04/2015 |
| V1.1 | Complete review by Steve Negrine | S. Negrine | 09/06/2015 |
| V1.2 | Validation of revisions before distribution ICDC | D. Dentant | 06/10/2015 |
| V1.3 | Adding experience feedback ICDC / ATOS | FJ Saint-Lager | 06/22/2015 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Approval of the document**

|  |  |  |
| --- | --- | --- |
| **Name** | **Function** | **Date / Visa** |
|  |  |  |

**Dissemination of the document**

|  |  |  |
| --- | --- | --- |
| **Direction** | **Service** | **No one** |
|  |  |  |

This document is property of the INPI, it can not be distributed without express authorization of the ISD INPI.

[I. Process payment requests: MIP 4](#_Toc422750729)

[I.1 Header requests MIP 4](#_Toc422750730)

[I.2 The bodies of the messages addressed to MIP 5](#_Toc422750731)

[I.2.1 Question: Credit card payment - PAYER\_CB (paymentCb) 7](#_Toc422750732)

[I.2.2 Answer: Credit card payment - PAYER\_CB 8](#_Toc422750733)

[I.2.3 Question: Check credit card payment - VERIFIER\_PAIEMENT - checkPayment 8](#_Toc422750734)

[I.2.4 Answer: Check credit card payment - VERIFIER\_PAIEMENT 8](#_Toc422750735)

[I.2.5 Question: Consult the customer account balance - CONSULTER\_COMPTE - consult 9](#_Toc422750736)

[I.2.6 Answer: Consult the customer account balance - CONSULTER\_COMPTE 9](#_Toc422750737)

[I.2.7 Question: Payment on Account and order creation in the collection Application - PAYER\_CREER\_COMMANDE - CreateOrder 9](#_Toc422750738)

[I.2.8 Answer: Payment on Account and order creation in the collection Application - PAYER\_CREER\_COMMANDE - CreateOrder 12](#_Toc422750739)

[I.2.9 Question: Edit command with the national number - MODIFIER\_COMMANDE - UpdateOrder 12](#_Toc422750740)

[I.2.10 Answer: Edit command with the national number - MODIFIER\_COMMANDE - UpdateOrder 13](#_Toc422750741)

[I.3 Common responses MIP: 13](#_Toc422750742)

[I.4 Objects modeling services, message and crafts objects MIP [V1R3] 19](#_Toc422750743)

[I.5 Example of payment CB 20](#_Toc422750744)

[I.6 Example payment CCL 23](#_Toc422750745)

# Process payment requests: MIP

Description :

MIP is the implementation of the INPI for receiving payment applications to the INPI.

The two possible types of payments are payment by credit card and payment on account (CDC) registered with the INPI.

Type of functional patterns:

**M 3: Synchronous messages Exchange**



sending application: Client application claiming payment

receiving application: MIP

## Header requests MIP

Messages for MIP always have the same header. It is a common part to all messages necessary for authentication of the client application:

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Wording** | **expected values** | **\*** |
|  | header |  |  |
| appliClient | Code assigned by the client application server MIP.  A new code will be created in MIP for the client application. | string | **O** |
| login | Login to the web service. Assigned by MIP for each web service every APPLI\_CLIENTE  A login will be created in MIP forclient application. | string | **O** |
| password | password for the connection (idem)  A password will be created in MIP forclient application. | string | **O** |
| dateCall | Date and time of the call by the client application. Trace the invocation time of the SW MIP possible de-sync server | string  (Standard format: dd / mm / yyyy hh24: mm: ss | **O** |
| idExt | unique identifier sent by the client application to identify an application to a MIP service. Attention can not revive a request to a service if it is already being processed for the same service and same identifier. This application will be rejected. It was not until the return of the first application service to send the second. | string | **O** |

(\*) Mandatory / Optional

Order ID or payment request. It is generated and transmitted by the client application. According to the client application and business service, this identifier refers to a trademark registration (DEM), a purchase in shop, an annuity payment or anything else to pay, an order, etc.

According to the invoked business service,PID control that this id is not treated several times for a client application. It is for the latter to manage the uniqueness of its calls for MIP that control with the identifier that consistency rules related to payment. These rules are detailed description with each business service. The torque data that control the uniqueness of an application for authorization (in the sense of CB portal) is id\_externe + amount payable. If two authorization requests are sent to MIP with the same id\_externe and two different amounts, this will generate two separate requests for authorization CB.

example:

* PID control that an application does not invoke several times PAIEMENT\_CB business service with the same amount id\_ext and even when this is id\_ext status "PAYMENT OK". If payment failed, MIP accept a new request for payment with the same id\_ext and the same or different amount.

A cyclical rule could be introduced if the client applications reuse their IDs.

**Note 1:**

An improvement is expected for the application Cashing returns a specific code (Command already created) with no internal ordering. The business application will handle the return code according to its business rules: stop treatment, continue numnater, etc.

**Note 2:**

MIP retains its data for 12 months.

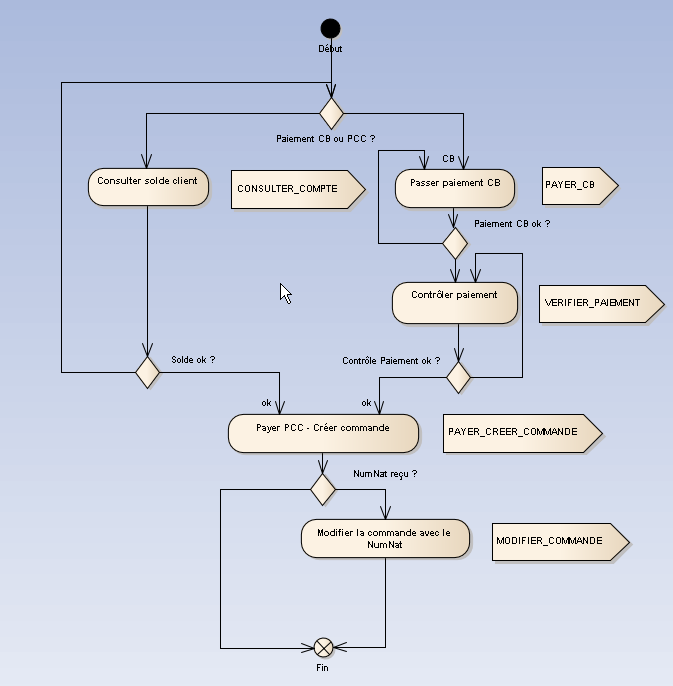
Moreover, the rules applied by Cashing (Creation of the command, CreateOrder) do not recycle the file number. In most cases, the couple ref\_cmde / ref\_dossier must be unique. See the team Cashing these rules specific to each product group.

## The bodies of the messages addressed to MIP

The services offered by MIP:

|  |  |
| --- | --- |
| **BUSINESS SERVICE** | **Description** |
| CONSULTER\_COMPTE consult | **Look instant balance of an account** |
| PAYER\_CB  **paymentCb** | **Pay by credit card** |
| VERIFIER\_PAIEMENT checkPayment | **Check that a payment was made** |
| PAYER\_CREER\_COMMANDE CreateOrder | **Pay by Client Account (CCL) and create his order**  **Create the control of a CB payment in advance** |
| MODIFIER\_COMMANDE UpdateOrder | **Add to order created the National Number recovered following the procedures.** |

The kinematics of calls expected in the ETS:



**Detail:**

- A payment is made by credit card (CB) or account (CDC).

- A payment CB ( "paymentCb") must be followed by creation command ( "CreateOrder").

- A payment CCL ( "CreateOrder") creates a direct order.

- At a creation command ( "CreateOrder"), a payment verification is performed.

- A creation of control can only be achieved if a payment is validated.

- Verification of the payment ( "checkPayment") is optional because the payment is also checked during the creation of the command ( "CreateOrder").

- Consultation of the CCL balance is not necessary to create an order since it is checked during the order creation ( "CreateOrder").

Warning: You can not recall a service being processed. Not until the end of his treatment for recall it.

For example, the "CreateOrder" service is called once with:

* idExt = '123456',
* appliClient = ETS-1 '

If we recall the same service with these parameters when it has not completed treatment, an error is returned and the second application will not be processed.

If payment CB NOK, the flow does not return to that MIP "replayed" an attempt to credit card payment, but to the business application that must decide the further processing (replay, abandonment, judgment of online business process)

Note: certain online procedures and services do not require use Numnat. The call to "PAYER\_CREER\_CMDE CreateOrder" is not followed by a change in the cmde Receipt ( "UpdateOrder"). The business application must ensure in this case to inform correctly and ref\_cmde ref\_dossier.

[V1R3] The [Modeling services, message and objects MIP trades](#_Modélisation_des_services,) identifies as a class diagram types and attributes described in the following chapters:

### Question: Credit card payment - PAYER\_CB (paymentCb)

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Wording** | **Expected value** | **\*** |
|  | Body text |  |  |
| typeCard | 0 = undefined, CB = 1, 2 = VISA, AMEX 3 = 4 = new guy  Possible choice on the CB portal: mandatory (to change the wsdl) | String | **F** |
| dueTotal | Total INPI. fractionated payment management. Not implemented to date. A\_PAYER = TOTAL\_DU  (Expressed in cents) | String | **O** |
| holderMail | mail address of the holder (payment ticket sent to this address with internal control reference to PSAs) | String | **O** |
| timeDisplay | display the payment page in seconds before timeout duration (if not supplied or <60 '', default portal CB: 900 '') | integer  900 '' / default | **F** |
| is3ds | 3D-Secure control Y / N? | Boolean  WE | **F** |
| langDisplay | Display language of the payment page (French by default) | String  FRA, GBR, ESP, ITA, DEU, NLD, SWE, PRT | **~~O~~ F** |
| urlPayOk | back page of CB portal to ETS after payment accepted | String | **O** |
| urlPayKo | Portal page back to ETS after CB refused payment | String | **O** |
| urlCancel | back page of CB portal to ETS canceled after payment. | String | **O** |
| Info | Text information for MIP | String | **F** |

***[V1R3] Back ETS experience:***

* typeCard is systematically evaluated with the value 0 (= undefined).

### Answer: Credit card payment - PAYER\_CB

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Wording** | **Expected value** | **\*** |
|  | Footer |  |  |
| payboxAccess | html to return to the client browser: redirect CB portal | String | **F** |

Warning :

- "PayboxAccess" refers html code interpreted by the client browser.

A redirection to the portal CB entry form page will be performed. The client will then be redirected to one of three url ( "urlPayOk", "urlPayKo", "urlCancel") after the payment or not

- To ensure that payment is accepted, you must invoke the "checkPayment" service ( "VERIFIER\_PAIEMENT") or directly use the "CreateOrder" service ( "PAYER\_CREER\_COMMANDE") that systematically verify that a payment has been validated (identified transaction the couple "idExt" "dueTotal") before creating the command. It is strongly recommended to use redirection CB portal to "urlPayOk" to consider payment as agreed.

- MIP is not informed of the cancellation CB (click on "Cancel": return to "urlCancel") or the browser is closed by the client before making the payment on the CB portal.

### Question: Check credit card payment - VERIFIER\_PAIEMENT - checkPayment

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Wording** | **Expected value** | **\*** |
|  | Body text |  |  |
| dueTotal | Total due to the INPI. fractionated payment management. | String | **O** |

### Answer: Check credit card payment - VERIFIER\_PAIEMENT

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Wording** | **Expected value** | **\*** |
|  | Footer |  |  |
| PaymentStatus | Payment Status | integer  340: payment realizes  341: nonexistent payment | **F** |
| labelPaymentStatus | Label associated with response | String | **F** |
| PaymentMode | Payment (CB / CPC) | String | **F** |
| authorizationNumber | authorization number if payment CB | String | **F** |

### Question: Consult the customer account balance - CONSULTER\_COMPTE - consult

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Wording** | **Expected value** | **\*** |
|  | Body text |  |  |
| customerAccount | Account Code | String | **O** |
| customerAccountPwd | Password associated with the INPI to this account | String | **O** |
| info | Text information for MIP or Cashing | String | **F** |

### Answer: Consult the customer account balance - CONSULTER\_COMPTE

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Wording** | **Expected value** | **\*** |
|  |  |  |  |
| customerAccount | No customer account (one returned by Cashing) | String | **F** |
| customerBalance | Account balance | float  in cents | **F** |
| holder | Account owner | Person | **F** |
| mailingAddressee | Recipient mail | Person | **F** |

**Note** The client account balance control will be performed by the calling application from the balance brought by MIP and the amount.

MIP offers two ways to control CPC

1: Check the balance and create the positive balance if cmde

2: Create the command directly. Cashing makes its own customer balance of control and can not in any way create a command generating a negative balance in the account.

### Question: Payment on Account and order creation in the collection Application - PAYER\_CREER\_COMMANDE - CreateOrder

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Wording** | **Expected value** | **\*** |
|  | Body text |  |  |
| products | *1 to n products* | Product Type | **O** |
| payment | Characteristics of the payment | Payment Type | **O** |
| customerRef | Client reference | String | **F** |
| with application | Features of Applicant / Owner | Type Person | **O** |
| attorney | Features of the Agent | Type Person | **O** |
| addressed | Receiver Features | Type Person | **O** |
| OrderRef | Command reference | String | **O** |
| ORDERDATE | Date of the order | String  Format standard ISO 8601 | **O** |
| reportRef | file's reference | String | **O** |
| isComplete | order completed | Boolean | **O** |

***[V1R3] Back ETS experience:***

* Pattern for generating the OrderRef (for the creation of control) = nomEProc\_idDemande.
  + Ex: 1577 demand for annuity business service: ANN\_1577

Different types:

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Wording** | **Expected value** | **\*** |
|  | Body text |  |  |
| ***Product*** |  |  |  |
| code | Product code known collection | String | **O** |
| quantity | Amount | integer | **O** |
| amount | Rate | integer  in cents | **O** |
| ***Payment*** |  |  |  |
| fashion | Payment Mode CB Rate, or on account CCL Client | String  CCL or CB | **O** |
| shipping | Shipping | integer  In cents | **F** |
| dated | Effective date of payment | String  Date hour minute second payment  Format standard ISO 8601 | **O** |
| amount | Payment amount | integer  In cents | **O** |
| transactionref | No. CB authorization passed by CB portal | String | **F** |
| customeraccount | customer account identifier expected by Cashing | String  Required if Mode = CCL | **O / F** |
| customerAccountPwd | Client password expected by Cashing | String  Required if Mode = CCL | **O / F** |
| **Person** |  |  |  |
| type | Moral or physical person | String  P or M | **O / F** |
| quality | title | String  Mr., Mrs., Ms. | **F** |
| lastname | Name | String | **F** |
| firstname | First name | String | **F** |
| companyName | Social reason | String | **F** |
| legalForm | Legal form | String | **F** |
| E-mail | E-mail | String | **F** |
| address | Address | Address Type | **O** |
|  |  |  |  |
| ***address*** |  |  |  |
| streetOne | Number and track name | String | **O** |
| streetTwo | Additional address | String | **F** |
| city | City | String | **O** |
| ZIP code | Postal code | String | **O** |
| countryCode | ISO Country Code | String | **O** |
| mention | mention distribution | String | **F** |
| department | Service means the service of the company in which a person works. | String | **F** |

Warning :

- The treatment carried out to create a command may take a few seconds (Application Cashing possible pending a portal back CB, ..). We must plan to wait the client (user).

**Person characteristic: They are issued by Cashing. MIP aims to create orders in Cashing. Cashing system rules were implemented by MIP.**

**MIP implements web services via its routine checks on the data transmitted (eg. Mandatory data)**

**Department means the service of the company in which a person works.**

***[V1R3] Back ETS experience:***

* The maximum size of the filled Personn.legalForm given in CreateOrder request by the ETS is fifty (50) characters, the rest is truncated.

**Point of attention for foreign addresses:**

Foreign addresses have the same structure as the standardized French addresses, therefore, it is difficult to map the address lines for foreign addresses. Also cashing application has the same input control rules for foreign addresses, therefore the client application must create a mapping for this particular type of address by feeding MIP:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Client Application** |  | **Transformation?** | **If empty default** | **MAPPING in block MIP address** |
| **Only ADDRESS INTERNATIONAL** |  |  |  |  |
| Line 1 | the name or the recipient's identification |  |  |  |
| line 2 |  | No | - | <StreetOne xsi: type = "xsd: string"> </ streetOne>? |
| line 3 |  | No |  | <StreetTwo xsi: type = "xsd: string"> </ streetTwo>? |
| line 4 |  | No | - | <City xsi: type = "xsd: string"> </ city>? |
| line 5 |  | Should be cut to 5 characters. | - | <ZipCode xsi: type = "xsd: string"> </ zipCode>? |
| line 6 |  | No |  | <Mention xsi: type = "xsd: string"> </ statement>? |
| line 7 | Country | Yes, put the associated ISO code | **mandatory field** | <CountryCode xsi: type = "xsd: string"> </ countryCode>? |

***[V1R3] Back ETS experience:***

* The Address.streetOne Data supplied with "-" when this information is not given in the ETS
* The Address.city Data supplied with "-" when this information is not given in the ETS
* The Address.zipCode Data supplied with "-" when this information is not given in the ETS
* The maximum size of the data Address.zipCode (international) filled in CreateOrder request by the ETS is five (5) characters, the rest is truncated.

### Answer: Payment on Account and order creation in the collection Application - PAYER\_CREER\_COMMANDE - CreateOrder

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Wording** | **Expected value** | **\*** |
|  | Footer |  |  |
| orderNum | the order number created in Cashing | integer | **~~O~~ F** |
| TotalAmount | Total amount of the order | float  in cents | **~~O~~ F** |
| CreateDate | Creation Date of the order | String  ddmmyyyy hh: mm: ss | **F** |
| encaissDate | Creation Date in Cash | String  ddmmyyyy hh: mm: ss | **F** |
| isPaidCb | He has the payment been made by CB? | String  true / false | **~~F~~ O** |
| paymentCbDate | Date of payment CB | String  dd / mm / yyyy | **F** |

Edit "isPaidCb" in boolean in wsdl

### Question: Edit command with the national number - MODIFIER\_COMMANDE - UpdateOrder

If NumNat was recovered for the formality processed, so you need to update the number in the order received.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Wording** | **Expected value** | **\*** |
|  | Body text |  |  |
| orderNum | Field "orderNum" of the response of "CreateOrder" | integer | **~~O~~**  **F** |
| reportNum | File number (NumNat) | String | **O** |

### Answer: Edit command with the national number - MODIFIER\_COMMANDE - UpdateOrder

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Wording** | **Expected value** | **\*** |
|  | Footer |  |  |
| orderNum | Order Number Cashing  Field "orderNum" of the response of "CreateOrder" | Int | **OF** |
| TotalAmount | Total amount of the order | float  in cents | **OF** |
| CreateDate | Creation Date of the order | String  ddmmyyyy hh: mm: ss | **F** |
| encaissDate | Creation Date in Cash | String  ddmmyyyy hh: mm: ss | **F** |

## Common responses MIP:

Common elements in any response MIP, peculiarities are treated at the description of each WS in the paragraphs above.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Wording** | **Expected value** | **\*** |
|  | Response MIP |  |  |
| dated | Date and time of treatment response = end web server. Trace SW invocation times of MIP + de-sync any servers + network response time | String  dd-mm-yyyy hh: mm: ss | **O** |
| code | MIP return code | integer | **O** |
| detail | return code label | String | **F** |

MIP responses to business applications reflect internal events MIP or the events of the application or Cashing CB portal. This simplifies the transcoding list Return Code - business applications do not need to know the details of the internal operations of the MIP MIP or translated internal codes more explicit codification functional.

MIP responses are returned in two different mechanisms.

During the technical processing of the SOAP message, the reply is sent to the client application as a SOAP\_FAULT with a codification of 1-99.

After the technical processing of the SOAP message, the SOAP response is OK (no error in the processing of the message).

| code | detail |
| --- | --- |
| COMMON TO ALL SERVICES: control the header of the message, internal error (MIP database connection problem, ...) (SOAPFault 10 to 99) | |
| 1 | web service processing MIP: OK |
| 10 | SW treatment: technical error |
| 11 | SW treatment: Error parsing SOAP |
| 12 | SW treatment: Internal error / error coding: products, people, payment mode, etc. |
| 20 | SW treatment: Code missing Application |
| 21 | SW treatment: Missing Login |
| 22 | SW treatment: Missing Password |
| 23 | SW Treatment: missing time-Call |
| 24 | SW Treatment: id\_ext missing |
| thirty | SW treatment: MIP does not know this Application |
| 31 | SW treatment: Application unauthorized |
| 32 | Login / password incorrect |
| 34 | Format (date-call) incorrect |
| 40 | The service is running on this id\_ext |
| 41 | Parameter error |
| 42 | return error code |
| 45 | technical error |
| 50 | The system meets a service error! |
| SERVICE "consult" (CONSULTER\_COMPTE) | |
| 0 | OK: return the balance, CDME creation, etc. |
| 100 | No response from Cashing System |
| 101 | Cashing technical error |
| 102 | refCmd must have a size less than or equal to 20 characters |
| 103 | refDossier must have a size less than or equal to 11 characters |
| 104 | Get unknown business code |
| 105 | Cashing technical error |
| 106 | Cashing technical error |
| 107 | Cashing technical error |
| 108 | Cashing technical error |
| 110 | COMPTE\_CLIENT missing |
| 111 | MOT-PASS account missing customer |
| 112 | TOTAL PAY-A-missing |
| 113 | TOTAL PAY-A-zero |
| 114 | CODE\_DEVISE missing |
| 120 | Error message creation Cashing |
| 131 | SW error Cashing |
| 131 | SW error Cashing |
| 131 | SW error Cashing |
| 131 | SW error Cashing |
| 131 | Technical Error = mail sending failed |
| 150 | The code must be entered appOrigin |
| 151 | Code appOrigin nonexistent POS system. |
| 464 | No connection to the client account. |
| 464 | No connection to the client account. |
| 464 | No connection to the client account. |
| 464 | No connection to the client account. |
| 165 | No connection to the client account. |
| SERVICE "paymentCb" (PAYER\_CB) | |
| 201 | TYPE\_CARD missing |
| 202 | AMOUNT\_TO\_PAY missing |
| 203 | DUE\_TOTAL missing |
| 204 | CURRENCY\_CODE missing |
| 205 | ORDER\_RER missing |
| 206 | HOLDER\_MAIL missing |
| 207 | DISPLAYED\_LANG missing |
| 208 | TIME\_DISPLAY missing |
| 209 | 3DS missing |
| 210 | URL\_OK missing |
| 211 | URL\_NOK missing |
| 212 | URL\_ANNULE missing |
| 213 | URL\_ERROR missing |
| 220 | CB unavailable portal |
| 221 | A return URLs is greater than 150 because. |
| 222 | missing mandatory variable |
| 223 | digital variable contains a character |
| 224 | PBX\_TOTAL inf. 3 c. or sup. 10 characters |
| 225 | PBX\_PORTEUR: invalid email size |
| 226 | Value in a variable> maximum size |
| 227 | incorrect customer language code |
| 228 | incorrect customer currency code |
| 230 | CB unavailable portal |
| 231 | technical error SW Portal CB |
| 232 | inconsistent data in the system |
| 240 | CB accepted payment or CDC already realized |
| 241 | current authorization CB portal |
| 242 | CB permission refused by the bank |
| 243 | Cancellation by the customer of his CB authorization application |
| 201 | expired payment page |
| 245 | CCL payment (or other than CB) already for this ID |
| 250 | technical error Paybox |
| 251 | technical error Paybox |
| SERVICE "checkPayment" (VERIFIER\_PAIEMENT) | |
| 301 | Amount to pay missing |
| 340 | payment made |
| 341 | nonexistent payment |
| 350 | Multiple payments for the same id\_ext |
| SERVICE "CreateOrder" (PAYER\_CREER\_COMMANDE) | |
| 401 | CMDE already exists (application identifier) ​​/ Payment / CCL Command already exists for this payment trial CB |
| 410 | Missing cmde Date |
| 411 | Fashion missing payment |
| 412 | incorrect payment method |
| 413 | Incorrect Date Format cmde |
| 414 | incorrect payment Date Format |
| 420 | mandatory country code |
| 421 | wrong country code |
| 422 | wrong civility code |
| 423 | incorrect legal form |
| 424 | Type person = M or P |
| 425 | Type person = M or P |
| 426 | compulsory social reason corporation |
| 427 | Name required for individual |
| 428 | postcode = 5 since. in France |
| 429 | mandatory postal code in France |
| 430 | postcode = 5 since. in France |
| 431 | Route (number and / or street) compulsory |
| 440 | created order |
| 441 | Payment for nonexistent CB cmde |
| 443 | CB payment already exists with this control test CCL |
| 450 | Closed body for this user |
| 451 | Several ligneCommandes with identical recipes codes for same flow |
| 452 | serious technical error type NullPointerException |
| 453 | Date validity expired user |
| 401 | Command with this reference and that date already exists |
| 401 | Command with this record number and date already exists. |
| 456 | Ref. File mandatory for Group B1 Product |
| 457 | Ref. Folder must be empty for different product group B1 |
| 460 | No applicant not found and / or not created. |
| 461 | No agent not found and / or not created. |
| 462 | No cmde can not be 0 |
| 463 | The receipt number must not be 0 |
| 464 | No customer or account authentication failure |
| 465 | The balance of the Customer Account is insufficient |
| 466 | product price differs from the product catalog |
| 467 | cmde total calculation differs from the amount transferred |
| 468 | technical error cmde creation or pre-cmde |
| 468 | technical error cmde creation or pre-cmde |
| 470 | technical error cmde creation or pre-cmde |
| 471 | technical error cmde creation or pre-cmde |
| 472 | Command must have a reference cmde |
| 473 | recipe code {0} empty or nonexistent |
| 474 | Type empty or erroneous payment (CB or CCL) |
| 475 | refCmd must be less / equal 20 since. |
| 476 | refDossier must be less / equal 11 since. |
| 477 | Product Code disabled at the time of Cmd. |
| 478 | Mandatory reference file for a code recipe product group C1. |
| 480 | No pre-Command matching cmde ID: {0} |
| 481 | No Payment for cmde ID: {0} |
| 485 | {0}: Adresse.Service max 38 car. |
| 486 | {0}: Adresse.Complément max 38 car. |
| 487 | {0}: Adresse.nomVoie max 38 car. |
| 488 | {0}: Adresse.mentionDistribution max 38 car. |
| 490 | {0}: Adresse.ville max 32 car. |
| 491 | {0}: Adresse.codePays max 2 car. |
| 492 | BD Creation Failed person |
| 493 | {0}: Adresse.titre max 5 for. |
| 494 | {0}: Adresse.nom max 50 car. |
| 495 | {0}: Adresse.titre max 30 car. |
| 496 | {0}: Adresse.formeJuridique max 5 for. |
| four hundred ninety seven | {0}: Adresse.raisonSociale max 50 car. |
| 498 | A command with the file number and this ref\_cmde already exists. |
| 1001 | List NON-ACCOUNTING DATES could be been fetched |
| 1040 | List DATES NON-ACCOUNTING OK |
| 1101 | Civilities list has been been fetched. |
| 1140 | List civilities OK. |
| 1201 | No products for these criteria = Empty list |
| 1202 | PRODUCTS list has been been fetched |
| 1203 | PRODUCTS list has been been fetched |
| 1204 | PRODUCTS list has been been fetched |
| 1240 | List PRODUCTS OK. |
| 1301 | COUNTRIES list has been been fetched. |
| 1340 | List COUNTRY OK. |
| 1401 | LEGAL FORMS list has been been fetched. |
| 1440 | List LEGAL FORMS OK. |
| 1501 | ACCOUNTING OPERATIONS list has been been fetched. |
| 1502 | database problem |
| 1540 | List ACCOUNTING OPERATIONS OK. |
| 2000 | The case is closed. |
| 2001 | The web service is functional. |
| SERVICE "UpdateOrder" (MODIFIER\_COMMANDE) | |
| 501 | ORDER nonexistent in MIP |
| 502 | already in full CONTROL MIP |
| 520 | technical error modification cmde |
| 520 | technical error modification cmde |
| 503 | No cmde does not match. |
| 503 | No cmde does not match. |
| 510 | No missing cmde |
| 511 | No missing folder |
| 540 | OK: No file added to cmde |
| 550 | technical error SHIFT No. cmde file |

***[V1R3] Back ETS experience:***

Interpretation of return codes MIP by ETS:

|  |  |  |  |
| --- | --- | --- | --- |
| CreateOrder | Return code MIP indicating that everything went well.  The ETS saves the order data base and directs the user to the next screen. | 440 | OK |
| CreateOrder | Return code indicating that MIP order has already been created.  The ETS does not store the order data base and directs the user to the next screen. | 401 | OK |
| CreateOrder | Return code indicating a nonexistent MIP payment (however unlockable by Middle Office, cf. QC 382/393) | 441 | KO |
| CreateOrder | Return code indicating that the MIP DC balance is insufficient to pay | 465 | KO |
|  |  |  |  |
| UpdateOrder | Return code indicating that MIP everything went well | 540 | OK |
|  |  |  |  |
| CheckPayment | Return code indicating that MIP everything went well | 1 | OK |
| CheckPayment | PaymentStatus MIP indicating that the payment is made  Pending Correction bug WS (normally 340 instead of 240).  Accordingly, the ETS function to validate the payment is disabled. | 240 | OK |
|  |  |  |  |
| Consult | Return code indicating that MIP everything went well | 0 | OK |
| Consult | MIP return code indicating that the account is unknown (login or incorrect password) | 464 | KO |
|  |  |  |  |
| PaymentCB | Return code indicating that MIP everything went well | 1 | OK |

Note : Any return code is considered a fatal error by the ETS.

## Modeling services, message and crafts objects MIP [V1R3]



## Example of payment CB [V1R3]

Note : The confidential credentials were masked with XXXX in the examples:

|  |
| --- |
| <? Xml version = "1.0" encoding = "UTF-8"?>  <soapenv: Envelope xmlns: soapenv= "http://schemas.xmlsoap.org/soap/envelope/" xmlns: xsd= "http://www.w3.org/2001/XMLSchema" xmlns: xsi= "http://www.w3.org/2001/XMLSchema-instance">  <soapenv: Body>  <ns1: paymentCb soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xmlns: ns1= "http://int-inpi-mip2.extranet.cdc.fr/soap">  <request href= "# id0"/>  </ns1: paymentCb>  <multiref id= "id0" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "ns2: PaymentCbRequest" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/" xmlns: ns2= "http://int-inpi-mip2.extranet.cdc.fr/soap">  <typeCard xsi: type= "xsd: string">0</typeCard>  <dueTotal xsi: type= "xsd: string">143600</dueTotal>  <holderMail xsi: type= "xsd: string">florent.juilletdesaint-lager@caissedesdepots.fr</holderMail>  <timeDisplay xsi: type= "xsd: int" xsi: nil= "true"/>  <is3ds xsi: type= "xsd: boolean" xsi: nil= "true"/>  <langDisplay xsi: type= "xsd: string">fr</langDisplay>  <urlPayOk xsi: type= "xsd: string">http: // localhost: 8080 / and3-pann-web / web / validerPaybox</urlPayOk>  <urlPayKo xsi: type= "xsd: string">http: // localhost: 8080 / and3-pann-web / web / erreurPaybox</urlPayKo>  <urlCancel xsi: type= "xsd: string">http: // localhost: 8080 / and3-pann-web / web / annulerPaybox</urlCancel>  <info xsi: type= "xsd: string" xsi: nil= "true"/>  <appliClient xsi: type= "xsd: string">XXXX</appliClient>  <login xsi: type= "xsd: string">XXXX</login>  <password xsi: type= "xsd: string">XXXX</password>  <dateCall xsi: type= "xsd: string">06/22/2015 1:56:01 p.m.</dateCall>  <idExt xsi: type= "xsd: string">1577</idExt>  </multiref>  </soapenv: Body>  </soapenv: Envelope> |
| <? Xml version = "1.0" encoding = "UTF-8"?>  <SOAP-ENV: Envelope xmlns: SOAP-ENV= "http://schemas.xmlsoap.org/soap/envelope/" xmlns: ns1= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: xsd= "http://www.w3.org/2001/XMLSchema" xmlns: xsi= "http://www.w3.org/2001/XMLSchema-instance" xmlns: SOAP-ENC= "http://schemas.xmlsoap.org/soap/encoding/" SOAP-ENV: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/">  <SOAP-ENV: Body>  <ns1: paymentCbResponse>  <return xsi: type= "SOAP-ENC: Struct">  <payboxAccess xsi: type= "xsd: string">& Lt; HTML & gt;  & Lt; HEAD & gt;  & Lt; TITLE & gt; Paybox & lt; / TITLE & gt; & lt; / HEAD & gt;  & Lt; BODY onload = "document.PAYBOX.submit ();" & gt; & lt; CENTER & gt;  & Lt; b & gt; & lt; br & gt; current connection & lt; br & gt; the s & amp payment server; eacute; curis & amp; eacute; ... & lt; / b & gt; & lt;! - Paybox MODULE.CGI V502 - & gt;  & Lt; name = Form Paybox Action = 'https: //preprod-tpeweb.paybox.com/cgi/MYchoix\_pagepaiement.cgi' Method = POST & gt;  & Lt; INPUT TYPE = hidden NAME = PBX\_DATA VALUE = 'v' & gt;  & Lt; INPUT TYPE = hidden name = PBX\_SITE value = '2338701' & gt;  & Lt; INPUT TYPE = hidden name = PBX\_RANG value = "01" & gt;  & Lt; INPUT TYPE = hidden name = PBX\_TOTAL value = '143600' & gt;  & Lt; INPUT TYPE = hidden name = PBX\_DEVISE value = '978' & gt;  & Lt; INPUT TYPE = hidden name = PBX\_LANGUE value = 'FRA' & gt;  & Lt; INPUT TYPE = hidden name = PBX\_CMD value = '25419' & gt;  & Lt; INPUT TYPE = hidden name = PBX\_PORTEUR value='florent.juilletdesaint-lager@caissedesdepots.fr'>  & Lt; INPUT TYPE = hidden name = PBX\_ANNULE value = 'http: // localhost: 8080 / and3-pann-web / web / annulerPaybox' & gt;  & Lt; INPUT TYPE = hidden name = PBX\_EFFECTUE value = 'http: // localhost: 8080 / and3-pann-web / web / validerPaybox' & gt;  & Lt; INPUT TYPE = hidden name = PBX\_REFUSE value = 'http: // localhost: 8080 / and3-pann-web / web / erreurPaybox' & gt;  & Lt; INPUT TYPE = hidden name = PBX\_RETOUR value='err:E;refcmd:R;mnt:M;trnsseq:T;auth:A;abn:B;typc:C;trnspbx:S;pays:Y;dtvc:D;ipc:I;bin6:N;emp:H;g3DS:G;enrl:O;a3DS:F;bin2:J;dtpay:W;hrpay:Q;sgn:K'>  & Lt; INPUT TYPE = hidden name = PBX\_PAYBOX value = 'https: //preprod-tpeweb.paybox.com/cgi/MYchoix\_pagepaiement.cgi'>  & Lt; INPUT TYPE = hidden name = PBX\_REPONDRE\_A value = 'http: //mopa2i-test.inpi.fr/cb/paybox'>  & Lt; INPUT TYPE = hidden name = value = PBX\_SOURCE 'HTML' & gt;  & Lt; INPUT TYPE = hidden name = PBX\_VERSION value = '314-OS UNKNOWN' & gt;  & Lt; INPUT TYPE = hidden name = PBX\_TYPECARTE value = '0' & gt;  & Lt; INPUT TYPE = SUBMIT VALUE = 'Paybox' & gt;  & Lt; / FORM & gt;  & Lt; / CENTER & gt; & lt; / BODY & gt;  & Lt; / HTML & gt;  </payboxAccess>  <dated xsi: type= "xsd: string">22-06-2015 1:56:02 p.m.</dated>  <code xsi: type= "xsd: int">1</code>  <detail xsi: type= "xsd: string">web service processing MIP: OK</detail>  </return>  </ns1: paymentCbResponse>  </SOAP-ENV: Body>  </SOAP-ENV: Envelope> |

|  |
| --- |
| <? Xml version = "1.0" encoding = "UTF-8"?>  <soapenv: Envelope xmlns: soapenv= "http://schemas.xmlsoap.org/soap/envelope/" xmlns: xsd= "http://www.w3.org/2001/XMLSchema" xmlns: xsi= "http://www.w3.org/2001/XMLSchema-instance">  <soapenv: Body>  <ns1: CreateOrder soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xmlns: ns1= "http://int-inpi-mip2.extranet.cdc.fr/soap">  <request href= "# id0"/>  </ns1: CreateOrder>  <multiref id= "id0" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "ns2: OrderRequest" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/" xmlns: ns2= "http://int-inpi-mip2.extranet.cdc.fr/soap">  <products soapenc: arrayType= "ns2: Product [1]" xsi: type= "soapenc: Array">  <products href= "# id1"/>  </products>  <payment href= "# id2"/>  <with application href= "# id3"/>  <attorney href= "# id4"/>  <addressee href= "# ID5"/>  <ORDERDATE xsi: type= "xsd: string">2015-06-22T13: 56: 01</ORDERDATE>  <OrderRef xsi: type= "xsd: string">ANN\_1577</OrderRef>  <reportRef xsi: type= "xsd: string" xsi: nil= "true"/>  <isComplete href= "# id6"/>  <appliClient xsi: type= "xsd: string">XXXX</appliClient>  <login xsi: type= "xsd: string">XXXX</login>  <password xsi: type= "xsd: string">XXXX</password>  <dateCall xsi: type= "xsd: string">06/22/2015 1:56:01 p.m.</dateCall>  <idExt xsi: type= "xsd: string">1577</idExt>  </multiref>  <multiref id= "id1" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "ns3: Product" xmlns: ns3= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">  <code xsi: type= "xsd: string">074AE</code>  <quantity href= "# ID7"/>  <amount href= "# ID8"/>  </multiref>  <multiref id= "id4" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "ns4: Person" xmlns: ns4= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">  <address href= "# ID9"/>  <type xsi: type= "xsd: string">P</type>  <quality xsi: type= "xsd: string">Mr.</quality>  <lastname xsi: type= "xsd: string">JULY OF SAINT-LAGER</lastname>  <firstname xsi: type= "xsd: string">Florent</firstname>  <companyName xsi: type= "xsd: string" xsi: nil= "true"/>  <legalForm xsi: type= "xsd: string" xsi: nil= "true"/>  <E-mail xsi: type= "xsd: string">florent.juilletdesaint-lager@caissedesdepots.fr</E-mail>  </multiref>  <multiref id= "id2" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "NS5: Payment" xmlns: NS5= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">  <fashion xsi: type= "xsd: string">CB</fashion>  <shipping href= "# ID10"/>  <dated xsi: type= "xsd: string">2015-06-22</dated>  <amount href= "# ID11"/>  <transactionRef xsi: type= "xsd: string" xsi: nil= "true"/>  <customerAccount xsi: type= "xsd: string"> </customerAccount>  <customerAccountPwd xsi: type= "xsd: string"> </customerAccountPwd>  </multiref>  <multiref id= "ID5" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "NS6: Person" xmlns: NS6= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">  <address href= "# ID12"/>  <type xsi: type= "xsd: string">P</type>  <quality xsi: type= "xsd: string">Mr.</quality>  <lastname xsi: type= "xsd: string">JULY OF SAINT-LAGER</lastname>  <firstname xsi: type= "xsd: string">Florent</firstname>  <companyName xsi: type= "xsd: string" xsi: nil= "true"/>  <legalForm xsi: type= "xsd: string" xsi: nil= "true"/>  <E-mail xsi: type= "xsd: string">florent.juilletdesaint-lager@caissedesdepots.fr</E-mail>  </multiref>  <multiref id= "id6" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "xsd: boolean" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">true</multiref>  <multiref id= "id3" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "ns7: Person" xmlns: ns7= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">  <address href= "# ID13"/>  <type xsi: type= "xsd: string">P</type>  <quality xsi: type= "xsd: string">Mr.</quality>  <lastname xsi: type= "xsd: string">JULY OF SAINT-LAGER</lastname>  <firstname xsi: type= "xsd: string">Florent</firstname>  <companyName xsi: type= "xsd: string" xsi: nil= "true"/>  <legalForm xsi: type= "xsd: string" xsi: nil= "true"/>  <E-mail xsi: type= "xsd: string">florent.juilletdesaint-lager@caissedesdepots.fr</E-mail>  </multiref>  <multiref id= "ID7" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "xsd: int" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">1</multiref>  <multiref id= "ID13" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "NS8: Address" xmlns: NS8= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">  <streetOne xsi: type= "xsd: string">30 AVENUE Leonardo da Vinci</streetOne>  <streetTwo xsi: type= "xsd: string">B31</streetTwo>  <city xsi: type= "xsd: string">Pessac</city>  <ZIP code xsi: type= "xsd: string">33600</ZIP code>  <countryCode xsi: type= "xsd: string">FR</countryCode>  <mention xsi: type= "xsd: string" xsi: nil= "true"/>  <department xsi: type= "xsd: string" xsi: nil= "true"/>  </multiref>  <multiref id= "ID10" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "xsd: int" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">0</multiref>  <multiref id= "ID8" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "xsd: int" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">143600</multiref>  <multiref id= "ID11" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "xsd: int" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">143600</multiref>  <multiref id= "ID9" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "NS9: Address" xmlns: NS9= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">  <streetOne xsi: type= "xsd: string">30 AVENUE Leonardo da Vinci</streetOne>  <streetTwo xsi: type= "xsd: string">B31</streetTwo>  <city xsi: type= "xsd: string">Pessac</city>  <ZIP code xsi: type= "xsd: string">33600</ZIP code>  <countryCode xsi: type= "xsd: string">FR</countryCode>  <mention xsi: type= "xsd: string" xsi: nil= "true"/>  <department xsi: type= "xsd: string" xsi: nil= "true"/>  </multiref>  <multiref id= "ID12" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "NS10: Address" xmlns: NS10= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">  <streetOne xsi: type= "xsd: string">30 AVENUE Leonardo da Vinci</streetOne>  <streetTwo xsi: type= "xsd: string">B31</streetTwo>  <city xsi: type= "xsd: string">Pessac</city>  <ZIP code xsi: type= "xsd: string">33600</ZIP code>  <countryCode xsi: type= "xsd: string">FR</countryCode>  <mention xsi: type= "xsd: string" xsi: nil= "true"/>  <department xsi: type= "xsd: string" xsi: nil= "true"/>  </multiref>  </soapenv: Body>  </soapenv: Envelope> |
| <? Xml version = "1.0" encoding = "UTF-8"?>  <SOAP-ENV: Envelope xmlns: SOAP-ENV= "http://schemas.xmlsoap.org/soap/envelope/" xmlns: ns1= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: xsd= "http://www.w3.org/2001/XMLSchema" xmlns: xsi= "http://www.w3.org/2001/XMLSchema-instance" xmlns: SOAP-ENC= "http://schemas.xmlsoap.org/soap/encoding/" SOAP-ENV: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/">  <SOAP-ENV: Body>  <ns1: createOrderResponse>  <return xsi: type= "SOAP-ENC: Struct">  <orderNum xsi: type= "xsd: int">2053093</orderNum>  <TotalAmount xsi: type= "xsd: float">143600</TotalAmount>  <CreateDate xsi: type= "xsd: string">22062015 0:00:00</CreateDate>  <encaissDate xsi: type= "xsd: string">16062015 0:00:00</encaissDate>  <isPaidCb xsi: type= "xsd: boolean">true</isPaidCb>  <paymentCbDate xsi: type= "xsd: string">06/22/2015</paymentCbDate>  <dated xsi: type= "xsd: string">22-06-2015 2:01:00 p.m.</dated>  <code xsi: type= "xsd: int">440</code>  <detail xsi: type= "xsd: string">Control credit © Ã © e</detail>  </return>  </ns1: createOrderResponse>  </SOAP-ENV: Body>  </SOAP-ENV: Envelope> |

## Example payment CCL [V1R3]

Note : The confidential credentials were masked with XXXX in the examples:

|  |
| --- |
| <? Xml version = "1.0" encoding = "UTF-8"?>  <soapenv: Envelope xmlns: soapenv= "http://schemas.xmlsoap.org/soap/envelope/" xmlns: xsd= "http://www.w3.org/2001/XMLSchema" xmlns: xsi= "http://www.w3.org/2001/XMLSchema-instance">  <soapenv: Body>  <ns1: consult soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xmlns: ns1= "http://int-inpi-mip2.extranet.cdc.fr/soap">  <request href= "# id0"/>  </ns1: consult>  <multiref id= "id0" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "ns2: ConsultRequest" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/" xmlns: ns2= "http://int-inpi-mip2.extranet.cdc.fr/soap">  <customerAccount xsi: type= "xsd: string">12</customerAccount>  <customerAccountPwd xsi: type= "xsd: string">test12</customerAccountPwd>  <info xsi: type= "xsd: string"> </info>  <appliClient xsi: type= "xsd: string">XXXX</appliClient>  <login xsi: type= "xsd: string">XXXX</login>  <password xsi: type= "xsd: string">XXXX</password>  <dateCall xsi: type= "xsd: string">06/22/2015 2:12:14 p.m.</dateCall>  <idExt xsi: type= "xsd: string">1578</idExt>  </multiref>  </soapenv: Body>  </soapenv: Envelope> |
| <? Xml version = "1.0" encoding = "UTF-8"?>  <SOAP-ENV: Envelope xmlns: SOAP-ENV= "http://schemas.xmlsoap.org/soap/envelope/" xmlns: ns1= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: xsd= "http://www.w3.org/2001/XMLSchema" xmlns: xsi= "http://www.w3.org/2001/XMLSchema-instance" xmlns: SOAP-ENC= "http://schemas.xmlsoap.org/soap/encoding/" SOAP-ENV: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/">  <SOAP-ENV: Body>  <ns1: consultResponse>  <return xsi: type= "SOAP-ENC: Struct">  <customerAccount xsi: type= "xsd: int">12</customerAccount>  <customerBalance xsi: type= "xsd: float">1016746224</customerBalance>  <holder xsi: type= "SOAP-ENC: Struct">  <type xsi: type= "xsd: string">M</type>  <quality xsi: nil= "true"/>  <lastname xsi: nil= "true"/>  <firstname xsi: nil= "true"/>  <companyName xsi: type= "xsd: string">CABINET BEAU Lomenie</companyName>  <legalForm xsi: nil= "true"/>  <address xsi: type= "SOAP-ENC: Struct">  <streetOne xsi: type= "xsd: string">158 UNIVERSITY STREET</streetOne>  <streetTwo xsi: nil= "true"/>  <city xsi: type= "xsd: string">PARIS CEDEX 07</city>  <ZIP code xsi: type= "xsd: string">75340</ZIP code>  <countryCode xsi: type= "xsd: string">FR</countryCode>  <mention xsi: nil= "true"/>  <department xsi: nil= "true"/>  </address>  <E-mail xsi: nil= "true"/>  </holder>  <mailingAddressee xsi: type= "SOAP-ENC: Struct">  <type xsi: nil= "true"/>  <quality xsi: nil= "true"/>  <lastname xsi: nil= "true"/>  <firstname xsi: nil= "true"/>  <companyName xsi: type= "xsd: string">CABINET BEAU Lomenie</companyName>  <legalForm xsi: nil= "true"/>  <address xsi: type= "SOAP-ENC: Struct">  <streetOne xsi: type= "xsd: string">158 UNIVERSITY STREET</streetOne>  <streetTwo xsi: nil= "true"/>  <city xsi: type= "xsd: string">PARIS CEDEX 07</city>  <ZIP code xsi: type= "xsd: string">75340</ZIP code>  <countryCode xsi: type= "xsd: string">FR</countryCode>  <mention xsi: nil= "true"/>  <department xsi: nil= "true"/>  </address>  <E-mail xsi: nil= "true"/>  </mailingAddressee>  <dated xsi: type= "xsd: string">22-06-2015 2:12:15 p.m.</dated>  <code xsi: type= "xsd: int">0</code>  <detail xsi: type= "xsd: string">OK: return the balance, credit © ation CDME, etc.</detail>  </return>  </ns1: consultResponse>  </SOAP-ENV: Body>  </SOAP-ENV: Envelope> |

|  |
| --- |
| <? Xml version = "1.0" encoding = "UTF-8"?>  <soapenv: Envelope xmlns: soapenv= "http://schemas.xmlsoap.org/soap/envelope/" xmlns: xsd= "http://www.w3.org/2001/XMLSchema" xmlns: xsi= "http://www.w3.org/2001/XMLSchema-instance">  <soapenv: Body>  <ns1: CreateOrder soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xmlns: ns1= "http://int-inpi-mip2.extranet.cdc.fr/soap">  <request href= "# id0"/>  </ns1: CreateOrder>  <multiref id= "id0" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "ns2: OrderRequest" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/" xmlns: ns2= "http://int-inpi-mip2.extranet.cdc.fr/soap">  <products soapenc: arrayType= "ns2: Product [1]" xsi: type= "soapenc: Array">  <products href= "# id1"/>  </products>  <payment href= "# id2"/>  <with application href= "# id3"/>  <attorney href= "# id4"/>  <addressee href= "# ID5"/>  <ORDERDATE xsi: type= "xsd: string">2015-06-22T14: 12: 14</ORDERDATE>  <OrderRef xsi: type= "xsd: string">ANN\_1578</OrderRef>  <reportRef xsi: type= "xsd: string" xsi: nil= "true"/>  <isComplete href= "# id6"/>  <appliClient xsi: type= "xsd: string">XXXX</appliClient>  <login xsi: type= "xsd: string">XXXX</login>  <password xsi: type= "xsd: string">XXXX</password>  <dateCall xsi: type= "xsd: string">06/22/2015 2:12:14 p.m.</dateCall>  <idExt xsi: type= "xsd: string">1578</idExt>  </multiref>  <multiref id= "id4" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "ns3: Person" xmlns: ns3= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">  <address href= "# ID7"/>  <type xsi: type= "xsd: string">M</type>  <quality xsi: type= "xsd: string" xsi: nil= "true"/>  <lastname xsi: type= "xsd: string" xsi: nil= "true"/>  <firstname xsi: type= "xsd: string" xsi: nil= "true"/>  <companyName xsi: type= "xsd: string">CABINET BEAU Lomenie</companyName>  <legalForm xsi: type= "xsd: string" xsi: nil= "true"/>  <E-mail xsi: type= "xsd: string">florent.juilletdesaint-lager@caissedesdepots.fr</E-mail>  </multiref>  <multiref id= "id3" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "ns4: Person" xmlns: ns4= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">  <address href= "# ID8"/>  <type xsi: type= "xsd: string">M</type>  <quality xsi: type= "xsd: string" xsi: nil= "true"/>  <lastname xsi: type= "xsd: string" xsi: nil= "true"/>  <firstname xsi: type= "xsd: string" xsi: nil= "true"/>  <companyName xsi: type= "xsd: string">CABINET BEAU Lomenie</companyName>  <legalForm xsi: type= "xsd: string" xsi: nil= "true"/>  <E-mail xsi: type= "xsd: string">florent.juilletdesaint-lager@caissedesdepots.fr</E-mail>  </multiref>  <multiref id= "id1" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "NS5: Product" xmlns: NS5= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">  <code xsi: type= "xsd: string">074AE</code>  <quantity href= "# ID9"/>  <amount href= "# ID10"/>  </multiref>  <multiref id= "id2" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "NS6: Payment" xmlns: NS6= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">  <fashion xsi: type= "xsd: string">CCL</fashion>  <shipping href= "# ID11"/>  <dated xsi: type= "xsd: string">2015-06-22</dated>  <amount href= "# ID12"/>  <transactionRef xsi: type= "xsd: string"> </transactionRef>  <customerAccount xsi: type= "xsd: string">12</customerAccount>  <customerAccountPwd xsi: type= "xsd: string">test12</customerAccountPwd>  </multiref>  <multiref id= "ID5" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "ns7: Person" xmlns: ns7= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">  <address href= "# ID13"/>  <type xsi: type= "xsd: string">M</type>  <quality xsi: type= "xsd: string" xsi: nil= "true"/>  <lastname xsi: type= "xsd: string" xsi: nil= "true"/>  <firstname xsi: type= "xsd: string" xsi: nil= "true"/>  <companyName xsi: type= "xsd: string">CABINET BEAU Lomenie</companyName>  <legalForm xsi: type= "xsd: string" xsi: nil= "true"/>  <E-mail xsi: type= "xsd: string">florent.juilletdesaint-lager@caissedesdepots.fr</E-mail>  </multiref>  <multiref id= "id6" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "xsd: boolean" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">true</multiref>  <multiref id= "ID12" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "xsd: int" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">167800</multiref>  <multiref id= "ID13" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "NS8: Address" xmlns: NS8= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">  <streetOne xsi: type= "xsd: string">158 UNIVERSITY STREET</streetOne>  <streetTwo xsi: type= "xsd: string" xsi: nil= "true"/>  <city xsi: type= "xsd: string">PARIS CEDEX 07</city>  <ZIP code xsi: type= "xsd: string">75340</ZIP code>  <countryCode xsi: type= "xsd: string">FR</countryCode>  <mention xsi: type= "xsd: string" xsi: nil= "true"/>  <department xsi: type= "xsd: string" xsi: nil= "true"/>  </multiref>  <multiref id= "ID8" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "NS9: Address" xmlns: NS9= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">  <streetOne xsi: type= "xsd: string">158 UNIVERSITY STREET</streetOne>  <streetTwo xsi: type= "xsd: string" xsi: nil= "true"/>  <city xsi: type= "xsd: string">PARIS CEDEX 07</city>  <ZIP code xsi: type= "xsd: string">75340</ZIP code>  <countryCode xsi: type= "xsd: string">FR</countryCode>  <mention xsi: type= "xsd: string" xsi: nil= "true"/>  <department xsi: type= "xsd: string" xsi: nil= "true"/>  </multiref>  <multiref id= "ID11" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "xsd: int" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">0</multiref>  <multiref id= "ID7" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "NS10: Address" xmlns: NS10= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">  <streetOne xsi: type= "xsd: string">158 UNIVERSITY STREET</streetOne>  <streetTwo xsi: type= "xsd: string" xsi: nil= "true"/>  <city xsi: type= "xsd: string">PARIS CEDEX 07</city>  <ZIP code xsi: type= "xsd: string">75340</ZIP code>  <countryCode xsi: type= "xsd: string">FR</countryCode>  <mention xsi: type= "xsd: string" xsi: nil= "true"/>  <department xsi: type= "xsd: string" xsi: nil= "true"/>  </multiref>  <multiref id= "ID9" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "xsd: int" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">1</multiref>  <multiref id= "ID10" soapenc: root= "0" soapenv: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/" xsi: type= "xsd: int" xmlns: soapenc= "http://schemas.xmlsoap.org/soap/encoding/">167800</multiref>  </soapenv: Body>  </soapenv: Envelope> |
| <? Xml version = "1.0" encoding = "UTF-8"?>  <SOAP-ENV: Envelope xmlns: SOAP-ENV= "http://schemas.xmlsoap.org/soap/envelope/" xmlns: ns1= "http://int-inpi-mip2.extranet.cdc.fr/soap" xmlns: xsd= "http://www.w3.org/2001/XMLSchema" xmlns: xsi= "http://www.w3.org/2001/XMLSchema-instance" xmlns: SOAP-ENC= "http://schemas.xmlsoap.org/soap/encoding/" SOAP-ENV: encodingStyle= "http://schemas.xmlsoap.org/soap/encoding/">  <SOAP-ENV: Body>  <ns1: createOrderResponse>  <return xsi: type= "SOAP-ENC: Struct">  <orderNum xsi: type= "xsd: int">2053095</orderNum>  <TotalAmount xsi: type= "xsd: float">167800</TotalAmount>  <CreateDate xsi: type= "xsd: string">22062015 0:00:00</CreateDate>  <encaissDate xsi: type= "xsd: string">16062015 0:00:00</encaissDate>  <isPaidCb xsi: type= "xsd: boolean">false</isPaidCb>  <paymentCbDate xsi: nil= "true"/>  <dated xsi: type= "xsd: string">22-06-2015 2:13:28 p.m.</dated>  <code xsi: type= "xsd: int">440</code>  <detail xsi: type= "xsd: string">Control credit © Ã © e</detail>  </return>  </ns1: createOrderResponse>  </SOAP-ENV: Body>  </SOAP-ENV: Envelope> |